# Beyond Euclidean Distance Matrices

**Laboratoire de Communications Audiovisuelles**
**Ecole polytechnique fédérale de Lausanne**

**Master Thesis**
**Department of Electrical and Electronics Engineering**

June 21st, 2019, EPFL, Lausanne

**Cem Ates Musluoglu**
**Supervisor: Dr. Adam James Scholefield**

# Acknowledgments

I would like to thank my supervisor Dr. Adam James Scholefield for this project. Thank you for this research topic, all your help and the meetings we had, I have left every one of them with more ideas and inspiration.

I would also like to thank Prof. Ivan Dokmanic for accepting to be the expert at my Master's project exam.

I would like to thank my family and friends for their support.

# Contents

# Abstract

In this text, we study the extension of Euclidean Distance Matrices (EDM) to other types of linear varieties. Constructing these matrices so as to contain the minimum distance between a set of linear varieties, we try to create a general framework for what we call Extended Euclidean Distance Matrices (EEDM) and look at theoretical similarities with EDMs. Then, we propose algorithms to reconstruct the set of linear varieties that generated the observed EEDM, where we concentrate on two subtypes: Minimum distance between a set of points with a set of linear varieties of any dimension; and minimum distance between a set of lines. Finally, we show that EDMs can also be easily extended from points in a Euclidean space to functions in finite dimensional Hilbert spaces.

# Notation

## Symbols

| | |
|---|---|
| $n, m$ | Number of measurements |
| $d$ | Dimension of the ambient space |
| $k, l$ | Dimension of the linear variety |
| $\mathbf{1}_n$ | Column vector of dimension $n$ containing all 1's |
| $\text{diag}(G)$ | Vector containing the diagonal of a square matrix $G$ |
| $\mathbf{x}_i$ | Point $i$ in space |
| $\mathbf{p}_j$ | Intercept vector of linear variety $j$ |
| $\mathbf{v}_j^\ell$ | $\ell$-th direction vector of linear variety $j$ |
| $\Phi$ | **For Chapters 2 and 3:** Matrix containing direction vectors for a given linear variety |
| $N$ | Matrix containing normal vectors for a given linear variety in its rows |
| $X, P$ | Matrices for which the columns contain points and intercepts respectively |
| $V_j$ | Matrix gathering the $\ell$-th direction vector for $m$ distinct linear varieties |
| $\mathcal{F}$ | Linear variety (set containing all the points) |
| $\{\mathcal{F}_i\}_{1\leq i\leq n}$ | Collection of $n$ distinct linear varieties |
| $D$ | Euclidean, or generalized, distance matrix |
| $\text{edm}(X)$ | Euclidean distance matrix generated by $X$ |
| $\text{edm}(X, P)$ | Euclidean distance matrix containing the distances between columns of $X$ and columns of $P$ |
| $d(\mathcal{F}, \mathcal{G})$ | Minimum distance between linear varieties $\mathcal{F}$ and $\mathcal{G}$ |
| $\text{eedm}\Big(\{\mathcal{F}_i\}\Big)$ | Distance matrix containing the distances between the linear varieties in the collection $\{\mathcal{F}_i\}$ |
| $\text{eedm}\Big(\{\mathcal{F}_i\}, \{\mathcal{G}_j\}\Big)$ | Distance matrix containing the distances between the linear varieties in the first and second collection |
| $\text{pldm}\Big(\{\mathbf{x}_i\}, \{\mathcal{F}_j\}\Big)$ | Distance matrix containing the distances between the points $\{\mathbf{x}_i\}$ and linear varieties $\{\mathcal{F}_j\}$ |
| $\text{affdim}(X)$ | Affine dimension of the columns of $X$ |
| $\text{affdim}(\mathcal{S})$ | Affine dimension of the set $\mathcal{S}$ |
| $\mathcal{N}$ | Null space |
| $\mathcal{R}$ | Range space |
| $\mathcal{S}^\perp$ | Orthogonal complement of a set $\mathcal{S}$ |
| $A^T$ | Transpose |

| | |
|---|---|
| $A^*$ | Hermitian Transpose |
| $A \circ B$ | Hadamard product of $A$ and $B$ |
| $A \oslash B$ | Hadamard division of $A$ by $B$ |
| $A^{\circ 2}$ | Hadamard square of $A$ |
| $H$ | Hilbert space of dimension $d$ |
| $\langle x, y \rangle$ | Inner product on $H$ |
| $\|\| \cdot \|\|^2$ | Both $\ell_2$ and $\mathcal{L}^2$ norms |
| $\mathrm{hdm}(X)$ | Hilbert distance matrix generated by $X$ |
| $\Phi$ | **For Chapter 4:** Orthonormal basis of $H$ |
| $\mathbb{1}\{\cdot\}$ | Indicator function |

## Acronyms and Abbreviations

| | |
|---|---|
| EDM | Euclidean Distance Matrix |
| EEDM | Extended Euclidean Distance Matrix |
| PLDM | Point to Linear variety Distance Matrix |
| LDM | Line Distance Matrix |
| HDM | Hilbert Distance Matrix |
| LV | Linear Variety |
| OA | Orthonormality Assumption |
| MDS | Multidimensional Sclaing |
| MDU | Multidimensional Unfolding |
| SDR | Semidefinite Relaxation |
| GD | Gradient Descent |
| PSD | Positive Semi Definite |
| EVD | Eigenvalue Decomposition |
| SVD | Sinular Value Decomposition |

# List of Figures

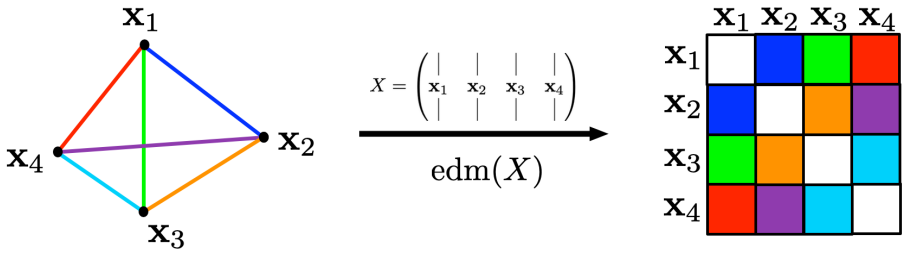
# Chapter 1

# Introduction

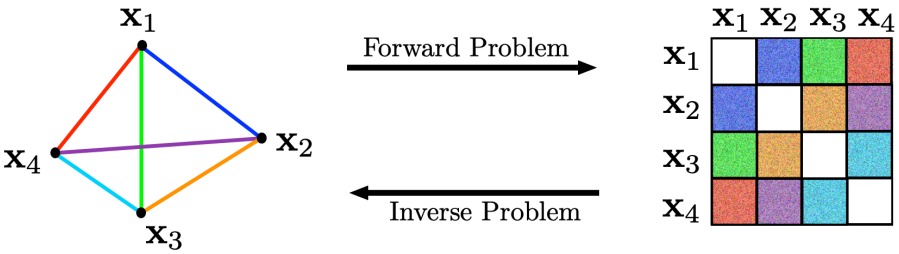
Euclidean distance matrices (EDM) come from a simple idea; given a configuration of points in space, the distances (squared) between them are stored in a structured manner in a symmetric matrix, where the components are given by:

$$d_{ij} = ||\mathbf{x}_i - \mathbf{x}_j||^2.$$

As we will discuss in Section 1.1, there has been many studies on the theory and practical aspects of EDMs, and shown that they implicitly contain information on the geometry of the points that generated the matrix. As depicted in Figure 1.1, going from points to EDMs is usually referred to as the forward problem, whereas the inverse problem consists of reconstructing the configuration by measuring the EDM. In practice the measurements may be noisy or some of them missing.




(a) Representation of the construction of EDMs for 4 points in 2D space.




(b) Same configuration as in 1.1a but we observe noisy measurements of the distances. Reconstructing the configuration of points is often named the "Inverse Problem".

Figure 1.1: Visual representation of EDMs.

Based on this existing theory, we ask ourselves whether there exists such matrices containing the distances between other type of objects than points. We restrain ourselves to linear varieties,

| | Points | Lines | Planes | ... |
|---|---|---|---|---|
| Points | EDM | ? | ? | ... |
| Lines | ? | ? | ? | ... |
| Planes | ? | ? | ? | ... |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋱ |

Figure 1.2: This figure represents the table we are trying to fill. EDMs have been studied in detail, and many properties are already known. The other cases remain less explored. Although points and lines in 2D and points and planes in 3D have been studied before by Krekovic et al. [1] and [2] respectively, we will adopt a slightly different approach, as discussed in Chapter 2, which is why we indicated them as unexplored too in the table.

i.e. lines, planes etc. in Euclidean space and look at methods of reconstructing the original configuration from noisy measurements. Due to the similar approach we will have to EDMs, we review in detail these matrices in Section 1.2, after having introduced previous works in Section 1.1. The final section of this chapter defines the notation and discusses some properties of linear varieties, to avoid ambiguities later on. Chapter 2 is dedicated to the extension of linear varieties, where we define these matrices and explore theoretical properties. Chapter 3 discusses inverse problems in the context of extended EDMs, i.e. methods to retrieve the original configurations from potentially noisy and missing measurements. Overall, our aim is to describe a general framework for these extended EDMs and explore how close these objects are to EDMs. An overview is depicted in Figure 1.2. Finally, in Chapter 4, we show that EDMs can be naturally extended to finite dimensional Hilbert spaces. In particular, we see that we obtain distance matrices with very similar properties between $d$-dimensional vectors and functions of a Hilbert space for which the basis contains $d$ elements.

## 1.1 A bit of history and context

Interestingly, EDMs have started to be used quite late considering their simple definition. Around the 1930's, Menger [3] and Schoenberg [4] discuss the sufficient and necessary conditions required for a set of values to be measures of the distances between a certain points. The matrix form as we know it appears in the paper from Young and Householder [5], where the important rank property of EDMs is proved. Further properties involving the rank have been studied by Gower [6] and [7]. Moreover, the set containing EDMs have been proven to be a convex cone by Hayden et al. [8] and analyzed in detail by Dattorro [9]. An early discussion of the problem of finding the closest EDM to a given matrix, and therefore projecting on the EDM cone, can be found in the papers by Mathar [10] and Glunt et al. [11]. On the other hand, knowing the distances between a set of points, the problem of reconstructing the set is called Multidimensional Scaling (MDS) and discussed by Torgerson [12]. A simple eigendecomposition algorithm, often called classical MDS is presented in the analysis by Gower [6]. The case where we consider two different sets of points is called Multidimensional Unfolding (MDU) and is presented by Schönemann [13] and also Crocco et al. [14]. However, the mapping from the set of points to distance matrices is not injective,

indeed, EDMs are invariant to rigid motion of the set (rotation, reflection, translation). Therefore, reconstructing the exact configuration requires a knowledge of some anchor points and can be done using the Orthogonal Procrustes Algorithm, proposed by Schoenemann [15]. It is worth mentioning that a large portion of these results were developed by the Psychometric (Quantitative Psychology) Society, where the distances were considered as dissimilarity measures and the points would correspond to stimuli, as described by Mead [16].

Many of the practical applications of EDMs correspond to inverse problems, i.e., reconstructing the configuration of points given the distance measurements. Moreover, in practice, these measurements may be inaccurate, which we can be modeled as a noisy observation of the original matrix, but also some values may be missing. An early measure, called stress, of goodness of fit is introduced in Kruskal [17], for example to answer how plausible it is to obtain the noisy measurements we observe for a certain set of points. Therefore, algorithms for estimating the true distances have been studied. Alternating Rank-Based EDM completion by Dokmanic and Vetterli [18] is based on alternatingly forcing the rank property and keeping the observed measures. Keshavan et al. [19] propose OptSpace, a matrix completion algorithm filling missing measurements and denoising existing ones. It is based on gradient descent on a non-convex Frobenius norm objective, but convergence to the global optimum is proven to be a high probability event. This algorithm can also be applied to other completion problems, such as collaborative filtering. A detailed analysis of the objective function can be found in the text by Keshavan et al. [20]. The Alternating Descent algorithm proposed by Parhizkar [21] uses the stress function as the objective, which can be expressed as a sum of fourth order polynomials. The algorithm then consists of minimizing independently these polynomials. This objective function has initially been studied by Takane et al. [22]. On the other hand, a Semidefinite Programming method has been studied by Alfakih et al. [23], where the rank constraint making the problem non-convex is dropped, but as mentioned by Krislock and Wolkowicz [24] and can easily be observed by experiments, the dimension parameter of the set of points is often lost. A possible solution involves including the trace norm, as discussed by Weinberger and Saul [25] and Biswas et al. [26]. A very compact and complete discussion about the theory of EDMs and comparison of these methods can be found in the paper by Dokmanic et al. [27].

These theoretical results have led to the use of EDMs in several applications, initially in psychometrics and then ranging from molecule and protein shape reconstruction (Holm and Sander [28] and Havel and Wüthrich [29]) to image manifold learning (Weinberger and Saul [25]), including also musical rythm analysis, "Euclidean rythms", by Demaine et al. [30]. These measures have also been used to determine the shape of a room, as described by Dokmanić et al. [31], Parhizkar [21], Dokmanic and Vetterli [18] and Dokmanić et al. [32]. In this problem, the distances are sometimes unlabeled, i.e. we measure the distances but we do not know which distance corresponds to which pair of points. Dokmanić et al. [31] propose the Echo Sorting Algorithm to tackle this problem. Phase retrieval discussed by Ranieri et al. [33] also deals with Euclidean distance geometry. Again, the distances are unlabeled and a solution is proposed by solving the turnpike problem. A different approach to the unassigned (unlabeled) distance geometry problem is presented by Huang and Dokmanić [34] where the observed distances are considered as following a discrete distribution, where the support is the set of the distances, and the problem comes down to finding a set of points with the same distance distribution.

Recently, the notion of distance matrices has been broadened to a larger class of objects than only static points. Tabaghi et al. [35] introduce the notion of kinetic EDMs (KEDM), where the points have a trajectory in time. Polynomial and bandlimited trajectories have been studied and shown that they can be recovered from KEDMs in the same paper. This technique has applications in, for example, localization of robot swarms (Cornejo and Nagpal [36]). In Simultaneous Localization And Mapping (SLAM, Durrant-Whyte and Bailey [37]), we study whether it is possible for a robot to both map its surrounding and localize itself within it. In this context, Krekovic et al. [1] propose a method to reconstruct the room by directly taking into account the higher dimensional

(compared to a point) aspect of the walls (lines, in 2D). Kreković et al. [2] extend this analysis to 3D room reconstruction, where walls are modeled as planes. A similar formulation allows to localize a microphone array and retrieve the direction of arrival of acoustic events in [38]. A detailed study of the uniqueness of such distance matrices is presented by Krekovic et al. [39]. We see that, gradually, the studies gained dimensions; MDS led to MDU, then points and lines and finally points and planes, which comes to our analysis; we look whether the matrices obtained from computing the distances of higher dimensional objects between each other have similar properties to EDMs and are there ways to reconstruct the original configurations.

## 1.2 Review of EDM and inverse problems

In this section, we discuss definitions and theoretical properties of EDMs. This will be helpful in the following chapters because we will study matrices similar to them and our analysis has foundations from what has previously been discovered for EDMs.

We place ourselves in a space of $d$ dimensions, which we will refer to as the dimension of the ambient space in the following text, and consider $n$ distinct points $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^n \subset \mathbb{R}^d$, we construct the matrix $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$ containing the points in its columns. Then, the EDM $D \in \mathbb{R}^{n\times n}$ such that $[D]_{i,j} = d_{ij} = ||\mathbf{x}_i - \mathbf{x}_j||^2$, is given by the function edm : $\mathbb{R}^{d\times n} \to \mathbb{R}^{n\times n}$, defined as:

$$D \triangleq \text{edm}(X) \triangleq \mathbf{1}_n \text{diag}(X^TX)^T - 2X^TX + \text{diag}(X^TX)\mathbf{1}_n^T, \tag{1.1}$$

where $\mathbf{1}_n$ is the $n$-dimensional column vector containing 1's in each of its entries.

**Remark.** *To see why this equation holds, which will also be useful later, we note that for conveniently sized matrices $A$ and $B$, the entry $(i, j)$ of $A^TB$ is given by $\mathbf{a}_i^T\mathbf{b}_j$, where $\mathbf{a}_i$ is the $i$-th column of $A$, and $\mathbf{b}_j$ is the $j$-th column of $B$. Similarly, a matrix containing $||\mathbf{x}_i||^2$ in row $i$ and every column $j$, can be written as $diag(X^TX)\mathbf{1}_n^T$, where the columns of $X$ contain the vectors $\mathbf{x}_i$. Since $d_{ij} = ||\mathbf{x}_i||^2 + ||\mathbf{x}_j||^2 - 2\mathbf{x}_i^T\mathbf{x}_j$, we obtain the relationship in 1.1.*

Some properties we can directly observe are:

1. **Symmetry**: $d_{ij} = d_{ji}$,
2. **Positivity**: $d_{ij} \geq 0$, $d_{ij} = 0 \iff i = j$,
3. **Triangle inequality**: $d_{ij} \leq d_{im} + d_{mj}$.

In most applications, $n$ is much larger than $d$, therefore, by computing the EDM, we usually use more values to describe the points than initially $(n \times d)$. By symmetry, we have $\binom{n}{2} = \frac{1}{2}n(n-1)$ distinct (in general) values, but we still lose some information. Looking for example at Figure 1.1, we would have obtained the same EDM by shifting every point to the left by the same amount, or rotating every point around the center of the points, which leads to the following result:

**Lemma 1.1** (Invariance to rigid motion)**.** *Suppose $R \in \mathbb{R}^{d\times d}$ a rotation/reflection matrix and $\mathbf{b} \in \mathbb{R}^d$ a translation vector, then*

$$edm(RX + R\mathbf{b}\mathbf{1}_n^T) = edm(X).$$

*Proof.* A matrix $R$ is a rotation/reflection matrix if and only if $R$ is orthogonal, i.e $R^TR = I$, hence, $(RX + R\mathbf{b}\mathbf{1}_n^T)^T(RX + R\mathbf{b}\mathbf{1}_n^T) = X^TX + \mathbf{1}_n\mathbf{b}^T\mathbf{b}\mathbf{1}_n^T + X^T\mathbf{b}\mathbf{1}_n^T + \mathbf{1}_n\mathbf{b}^TX$. By linearity of the diag operator, and observing that $X^T\mathbf{b}\mathbf{1}_n^T = \text{diag}(X^T\mathbf{b}\mathbf{1}_n^T)\mathbf{1}_n^T$ and $\text{diag}(X^T\mathbf{b}\mathbf{1}_n^T) = \text{diag}(\mathbf{1}_n\mathbf{b}^TX)$, we obtain the desired result. □

This result shows that EDMs only keep relative information between the points and that infinitely many point sets can result in the same EDM, as long as the configurations are differing up to a rigid motion. This is important to know since many algorithms used in inverse problems find one of these solutions, that we denote as $\widehat{\mathcal{X}}$, without a guarantee of returning the true one, $\mathcal{X}$. This requires additional knowledge, such as the true coordinates of a subset of $\mathcal{X}$. Schoenemann [15] describes a way to find the required orthogonal matrix and translation vector to compute the best aligning configuration, called the Orthogonal Procrustes problem. The procedure is presented in Algorithm 1, where the matrices $X_a, \widehat{X}_a \in \mathbb{R}^{d\times n'}$ contain a subset of size $n' \leq n$ of $\mathcal{X}$ and $\widehat{\mathcal{X}}$ respectively, in their columns.

**Algorithm 1** Orthogonal Procrustes, Schoenemann [15]

**function** OrthogonalProcrustes($\widehat{X}, \widehat{X}_a, X_a$)
  $\bar{\mathbf{x}}_a \leftarrow \frac{1}{n'} X_a \mathbf{1}_{n'}$  $\bar{X}_a \leftarrow X_a - \bar{\mathbf{x}}_a \mathbf{1}_{n'}^T$
  $\widetilde{\mathbf{x}}_a \leftarrow \frac{1}{n'} \widehat{X}_a \mathbf{1}_{n'}$  $\widetilde{X}_a \leftarrow \widehat{X}_a - \widetilde{\mathbf{x}}_a \mathbf{1}_{n'}^T$
  $U\Sigma V^T \leftarrow \widetilde{X}_a \bar{X}_a^T$ ▷ SVD
  $R \leftarrow VU^T$
  **return** $R(\widehat{X} - \widetilde{\mathbf{x}}_a \mathbf{1}_n^T) + \bar{\mathbf{x}}_a \mathbf{1}_n^T$
**end function**

The previously cited characteristics of EDMs, namely symmetry, positivity and the triangle inequality are only necessary conditions, but not sufficient. Indeed, there exists matrices that satisfy these conditions but are not EDMs, i.e. their components do not describe a configuration of points in Euclidean space. Gower [6] described a way to fully characterize these matrices, as given in the theorem below.

**Theorem 1.2** (Gower [6])**.** *A matrix $D$ is an EDM if and only if $-\frac{1}{2}(I - \mathbf{1}_n \mathbf{s}^T) D (I - \mathbf{s}\mathbf{1}_n^T) \succcurlyeq 0$ for any $\mathbf{s} \in \mathbb{R}^n$ such that $\mathbf{s}^T \mathbf{1}_n = 1$ and $\mathbf{s}^T D \neq \mathbf{0}$.*

*Proof.* It can be shown that $-\frac{1}{2}(I - \mathbf{1}_n \mathbf{s}^T) D (I - \mathbf{s}\mathbf{1}_n^T) = Y^T Y$, i.e. a Gram matrix of a certain configuration of points $Y$, hence it must be positive semi-definite (PSD). Moreover, since $\mathrm{tr}(D) = 0$, we have that $D$ is not PSD $\Rightarrow \mathbf{w}^T D \mathbf{w} < 0$ for some $\mathbf{w} \in \mathbb{R}^n$. Therefore, we can take $\mathbf{w} = (I - \mathbf{s}\mathbf{1}_n^T)\mathbf{u}$ and the relationship holds as long as $I - \mathbf{s}\mathbf{1}_n^T$ is invertible. However, we want $\mathbf{u}^T Y^T Y \mathbf{u} \geq 0$, which implies $\det(I - \mathbf{s}\mathbf{1}_n^T) = 0$ and $\mathbf{s}^T \mathbf{1}_n = 1$. Moreover, we need to avoid $D\mathbf{s} = \mathbf{0}$ because otherwise $-\frac{1}{2}(I - \mathbf{1}_n \mathbf{s}^T) D (I - \mathbf{s}\mathbf{1}_n^T) = -\frac{1}{2} D$ which is not PSD. Therefore if there exists $\mathbf{s} : \mathbf{s}^T \mathbf{1}_n = 1, D\mathbf{s} \neq \mathbf{0}$ for which the expression in the theorem is PSD, then $D$ is Euclidean.

For the converse part, consider another vector $\mathbf{t} : \mathbf{t}^T \mathbf{1}_n = 1$. Then $(I - \mathbf{1}_n \mathbf{t}^T)(I - \mathbf{1}_n \mathbf{s}^T) = (I - \mathbf{1}_n \mathbf{t}^T)$, consequently, $-\frac{1}{2}(I - \mathbf{1}_n \mathbf{t}^T)(I - \mathbf{1}_n \mathbf{s}^T) D (I - \mathbf{s}\mathbf{1}_n^T)(I - \mathbf{t}\mathbf{1}_n^T) = (I - \mathbf{1}_n \mathbf{t}^T) D (I - \mathbf{t}\mathbf{1}_n^T)$. Therefore, for a vector $\mathbf{s} : \mathbf{s}^T \mathbf{1}_n = 1, D\mathbf{s} \neq \mathbf{0}$, if $-\frac{1}{2}(I - \mathbf{1}_n \mathbf{s}^T) D (I - \mathbf{s}\mathbf{1}_n^T)$ is PSD, then it is PSD for all such $\mathbf{s}$. □

**Corollary 1.3** (Schoenberg [4])**.** *In particular, the previous theorem holds for $\mathbf{s} = \frac{1}{n}\mathbf{1}$. In this case, $J = I - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is the geometric centering matrix, where $XJ$ is centered at the origin, and we have $-\frac{1}{2} JDJ \succcurlyeq 0$.*

The classical MDS algorithm mentioned earlier is based on these results. Taking the geometric centering matrix as defined in the previous corollary, we remark that $-\frac{1}{2} JDJ = (XJ)^T (XJ)$ which is the Gram matrix corresponding to the centered version of $X$ around the origin. Using the eigenvalue decomposition (EVD), we have $(XJ)^T (XJ) = U \Lambda U^T \Rightarrow \hat{X} = \Lambda^{1/2} U^T$, where $\hat{X}$ corresponds to the reconstruction of the point set centered at the origin. We note that $\Lambda^{1/2} U^T \in \mathbb{R}^{n\times n}$, but since $\mathrm{rank}(X^T X) = d$ (supposing $d < n$), then, the last $n - d$ eigenvalues are 0, considering we order them in decreasing absolute value. Hence, to obtain a reconstruction in

$\mathbb{R}^{d\times n}$, we should use $[\text{diag}(\sqrt{\lambda_1}\dots\sqrt{\lambda_d}), \mathbf{0}_{d\times(n-d)}]U^T$. As pointed out by Dokmanic et al. [27], the truncation of the $d$ largest eigenvalues has also the advantage of being relatively robust to noisy measurements. Algorithm 2 describes this procedure concisely.

**Remark.** *In Algorithm 2, we took* $\mathbf{s} = \frac{1}{n}\mathbf{1}$*, which leads to the reconstruction centered around the origin. Other options could also be considered. For example taking* $\mathbf{s} = \mathbf{e}_i$ *reconstructs a configuration such that point* $i$ *is at the origin.*

**Algorithm 2** Classical MDS, Gower [6]

```
function ClassicalMDS(D)
    J ← I − (1/n) 1 1^T
    U Λ U^T ← −(1/2) J D J                                   ▷ EVD
    return [diag(√λ_1 ... √λ_d), 0_{d×(n−d)}] U^T
end function
```

In the derivation of the previous algorithm, we used the rank of the Gram matrix to obtain a result on the number of non-zero eigenvalues which led to the reconstruction of the configuration of points. Indeed, in usual applications, the dimension of the ambient space is much smaller than the number of measurements. Therefore, we obtain a relatively big matrix for the EDM, however, the next theorem shows that the *embedding (or affine) dimension* of the point set is still implicitly present.

**Theorem 1.4** (Gower [7])**.** *The rank of an EDM is at most* $r+2$*, where* $r = affdim(X) \leq d$*. Morover, if the points lie on the relative boundary of an* $r$*-dimensional hypersphere,* $rank(D) = r+1$*.*

**Remark.** *The affine dimension is related to* $D$ *with :* $r = affdim(X) = rank(JDJ)$*.*

**Remark.** *By* $r$*-dimensional hypersphere, we mean the hypersphere requiring at least* $r$ *dimensions for it to exist in an Euclidean space. For example,* $r = 2$ *for a circle.*

*Proof.* For points placed in $d$ dimensions without a particular configuration, Equation 1.1 allows us to see that $\text{rank}(\text{edm}(X)) \leq d+2$, where we use the fact that the rank of an outer product is 1, $\text{rank}(X^TX) \leq \min(\text{rank}(X), \text{rank}(X)) = d$, and $\text{rank}(A+B) \leq \text{rank}(A) + \text{rank}(B)$.

If the points lie on an affine subspace of dimension $r$, then this means that there exists an orthogonal matrix $R \in \mathbb{R}^{d\times d}$ and a vector $\mathbf{b} \in \mathbb{R}^d$ such that the last $d-r$ entries of the columns of $\tilde{X} = RX + \mathbf{b}\mathbf{1}^T$ are 0. Therefore $\text{rank}(\tilde{X}^T\tilde{X}) = r$ and $\text{rank}(\text{edm}(\tilde{X})) \leq r+2$. Since $\text{edm}(\tilde{X}) = \text{edm}(X)$ by Lemma 1.1, $\text{rank}(\text{edm}(X)) \leq r+2$.

For the last result, we use the same argument as the one by Gower [7]. Suppose that $\text{rank}(D) = r'$ and, without loss of generality, that the points are already centered around the origin ($XJ = X$). Then, there exists $n-r'$ linearly independent vectors $\mathbf{u}_i$ such that $D\mathbf{u}_i = \mathbf{0}, \forall 1 \leq i \leq n-r'$. Using Theorem 2 by Gower [7], we can also find a generalized inverse $D^-$ of $D$ such that $\mathbf{1}_n^T D^- D = \mathbf{1}_n^T$. This implies that $\mathbf{1}_n^T\mathbf{u}_i = 0$. But then, $\mathbf{u}_i$'s also belong to the null space of $X^TX = -\frac{1}{2}JDJ$. Moreover, $\frac{1}{n}\mathbf{1}_n$ is also in the null space of the Gram matrix, and also linearly independent from $\mathbf{u}_i$'s. Theorem 1 by Gower [7] states that if the points $X$ lie on the relative boundary of a hypersphere, $\mathbf{1}_n^T D^- \mathbf{1}_n \neq 0$. This condition implies (detailed in the paper) that other vectors in the null space of the Gramian are linear combinations of $\{\mathbf{u}_i\}_{1\leq i\leq n-r'}$ and $\frac{1}{n}\mathbf{1}_n$. By the rank-nullity theorem, $\text{rank}(X^TX) = r' - 1 = \text{rank}(X)$. □

**Example 1.1.** *Randomly distributed points in* $\mathbb{R}^3$ *will most probably result in a rank equal to 5. Points on a plane in* $\mathbb{R}^3$ *will result in a rank equal to 4. Points on the boundary of a circle in* $\mathbb{R}^4$ *will result in a rank of 3.*

This is an important property of EDMs, and we will see in Chapter 2 that some extensions of EDMs have also a rank constraint. Dokmanic and Vetterli [18] (Algorithm 3) and Keshavan et al. [19] (OptSpace) propose algorithms robust to noisy and missing measurements of EDM components to estimate the original matrix. These measurements, contained in the matrix $\widetilde{D} \in \mathbb{R}^{n\times n}$ can be expressed using the model:

$$\widetilde{D} = D + Z, \tag{1.2}$$

where we consider the additive noise model and usually consider $z_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$. Additionally, some measurements may not be available to us, therefore the complete observed matrix can be written as $M \circ \widetilde{D}$. Here, "$\circ$" denotes the component-wise, or Hadamard, product and $M$ is a binary mask, putting the missing measurements to 0. Algorithm 3 proposes a method that enforces the rank constraint and the observed measurements in an alternating manner; however, this process may converge to a matrix that is not an EDM or may even not converge at all. The notation $D_M$ indicates the entries of $D$ restricted to non-zero entries of $M$.

**Algorithm 3** Alternating Rank-Based EDM Completion, Dokmanic and Vetterli [18]

**function** RankCompleteEDM($\widetilde{D}, M$)
  $D_M \leftarrow \widetilde{D}_M$
  $D_{\mathbf{1}\mathbf{1}^T - M} \leftarrow \mu$ ▷ Initialize unobserved entries
  **repeat**
    $D \leftarrow$ EVThreshold($D, r$)
    $D_M \leftarrow \widetilde{D}_M$ ▷ Force known entries
    $\mathrm{diag}(D) \leftarrow \mathbf{0}$ ▷ Diagonal gets $\mathbf{0}$
    $D \leftarrow (D)_+$ ▷ Zero the negative entries
  **until** MaxIter or Convergence
  **return** $D$
**end function**

**function** EVThreshold($D, r$)
  $U\Lambda U^T \leftarrow D$ ▷ EVD
  $\Sigma \leftarrow \mathrm{diag}(\lambda_1 \dots \lambda_r, 0 \dots 0)$
  **return** $U\Sigma U^T$
**end function**

Other algorithms where the rank property is not directly exploited rely on the minimization of the Frobenius matrix norm:

$$\min_{X\in\mathbb{R}^{d\times n}} ||M \circ (\mathrm{edm}(X) - \widetilde{D})||_F^2. \tag{1.3}$$

This objective function is often called the s-stress function, initially introduced by Takane et al. [22]. Parhizkar [21] notes that for each component $X_{ij}$ of $X$, the s-stress function can be written as a fourth-order polynomial, the overall function being a sum of these polynomials. The same paper proposes a greedy algorithm, Alternating Descent, where at each stage, these fourth-order polynomials are minimized, independently of the rest. The process is repeated until convergence or the maximum number of iterations is attained.

On the other hand, this problem can also be written as the following semidefinite program:

$$\begin{aligned} \underset{H}{\text{maximize}} \quad & \mathrm{tr}(H) - \lambda||M \circ (\widetilde{D} - \mathcal{K}(QHQ^T))||_F \\ \text{subject to} \quad & H \in \mathbb{S}_+^{n-1}, \end{aligned} \tag{P}$$

where the function $\mathcal{K} : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ is the equivalent of the function edm, but takes the Gram matrix as argument, i.e. $\mathcal{K}(X^TX) = \mathrm{edm}(X)$. The matrix $Q \in \mathbb{R}^{n\times(n-1)}$ must satisfy $Q^T\mathbf{1} = \mathbf{0}$, $Q^TQ = I$ and $QQ^T = J$. The choice proposed by Alfakih et al. [23] is $Q = [y\mathbf{1}, x\mathbf{1}\mathbf{1}^T + I]^T$, where $x = -1/(n + \sqrt{n})$ and $y = -1/\sqrt{n}$. We omit the derivation of the problem (P), but full explanations and analysis can be found in the papers by Alfakih et al. [23], Weinberger and Saul

[25], Biswas et al. [26], Dokmanic et al. [27]. In the remaining of this text, we will refer to this semidefinite program as SDR (for semidefinite relaxation).

For the sake of completeness, we also present below the properties of the matrix manifold containing every $n \times n$ EDM.

**Definition 1.1.** *A subset $K$ of a vector space $V$ is a cone if $\forall x \in C$, $\theta \in \mathbb{R}^+$, $\theta x \in K$. Moreover, $K$ is a convex cone if $\forall x, y \in K$ and $\alpha, \beta \in \mathbb{R}^+$, $\alpha x + \beta y \in K$.*

**Definition 1.2.** *We denote by $\mathbb{EDM}^n$ the set containing every EDM of dimension $n \times n$. The set of hollow and symmetric matrices are given by $\mathbb{S}_h^n = \{A \in \mathbb{S}^n : diag(A) = \mathbf{0}\}$. The set of positive semidefinite matrices is given $\mathbb{S}_+^n = \{A \in \mathbb{S}^n : A \succcurlyeq 0\}$. Finally, we also define the orthogonal complement of the geometric center subspace $\mathbb{S}_c^{n\perp} = \{\mathbf{u}\mathbf{1}^T + \mathbf{1}\mathbf{u}^T : \mathbf{u} \in \mathbb{R}^n\}$.*

Following the results from [3], [4], [6] and Hayden et al. [8], Dattorro [9] gives a compact equation for the EDM cone; $\mathbb{EDM}^n = \mathbb{S}_h^n \cap (\mathbb{S}_c^{n\perp} - \mathbb{S}_+^n)$, as written in one of the earliest pages of this last text. Also, using the above definitions and properties, we can see that $\mathbb{EDM}^n$ is a closed convex cone.

**Lemma 1.5** (Dattorro [9], Parhizkar [21])**.** *The relative interior of $\mathbb{EDM}^n$ is given by*

$$relint\ \mathbb{EDM}^n = \{D \in \mathbb{EDM}^n : rank(JDJ) = n - 1\},$$

*which is a convex cone. The relative boundary is*

$$rel\ \partial\ \mathbb{EDM}^n = \bigcup_{r=0}^{n-2} \{D \in \mathbb{EDM}^n : rank(JDJ) = r\}.$$

This lemma shows that every EDM with $\text{rank}(JDJ) < n - 1$ lie in the boundary of $\mathbb{EDM}^n$. As pointed out by Parhizkar [21], we do not have a guarantee of convexity for the relative border of the manifold, which justifies why projecting onto the EDM cone is difficult.

## 1.3 Linear Varieties

In our analysis of extension of EDMs, we will study matrices containing the distance between one linear variety and another in their entries. Therefore, this section provides definitions of the latter so that we avoid ambiguities later on. Most of the definitions follow the book by Vetterli et al. [40], where more detailed analysis can be found.

**Definition 1.3.** *A vector space $V$ is said to have dimension $d$ when it contains a linearly independent set of $d$ elements and every set with $d + 1$ or more elements is linearly dependent.*

**Definition 1.4** (Linear varieties, LV)**.** *A nonempty subset $S$ of a vector space $V$ is a subspace when it is closed under vector addition and scalar multiplication. A subset $T$ of $V$ is an affine subspace if there exists an $x \in V$ such that any $t \in T$ can be written as $x + s$ for some $s \in S$. If $V = \mathbb{R}^d$, these affine subspaces are called linear varieties or flats.*

Throughout this text, we will consider $V = \mathbb{R}^d$, therefore of dimension $d$. Based on Definition 1.4, LVs are a set of points which can be written as the set $\mathcal{F} = \{\mathbf{x} \in \mathbb{R}^d : N\mathbf{x} = \mathbf{c}\}$, where $N \in \mathbb{R}^{(d-k)\times d}$, $k \leq d$. In addition, if $\text{rank}(N) = d - k$, the matrix describes an LV of dimension $k$. In the following parts, we will consider that $N$ is full rank and, without loss of generality, that its rows are unit norm. Special cases include $k = 0$, in which case we have a point, $k = 1$, which results in a line, $k = 2$, which describes a plane. By the rank-nullity theorem, there exists $k$ linearly

independent vectors $\mathbf{v}^\ell \in \mathbb{R}^d$ such that $N\mathbf{v}^\ell = \mathbf{0}$ and any $\mathbf{x}$ belonging to this set can be written as a parametric equation of the form

$$\mathbf{f}(\mathbf{t}) = \mathbf{p} + \Phi\mathbf{t} = \mathbf{p} + \sum_{\ell=1}^{k} \mathbf{v}^\ell t_\ell, \tag{1.4}$$

where $\Phi = [\mathbf{v}^1 \dots \mathbf{v}^k] \in \mathbb{R}^{d\times k}$, $\mathbf{t} \in \mathbb{R}^k$ and $\mathbf{p} \in \mathbb{R}^d$ such that $N\mathbf{p} = \mathbf{c}$, or equivalently, $\mathbf{p} = N^T\mathbf{c}$, since $NN^T = I$. We also observe that the rows of $N$ describe an orthonormal basis for the vectors normal to the LV and any row $\mathbf{n}^j$ of N belongs to $\mathcal{N}(\Phi^T)$, and therefore $\mathbf{p} \in \mathcal{N}(\Phi^T)$ too. We call such vectors $\mathbf{p}$, $\mathbf{v}^\ell$, $\mathbf{n}^j$ the intercept, direction and normal vectors respectively. Again, without loss of generality, we may assume that the $\mathbf{v}^\ell$'s are unit norm. Overall, any LV can be fully described by the pair $(N, \mathbf{c})$ or $(\mathbf{p}, \Phi)$ using the above assumptions. This is summarized below.

**Assumption** (Orthonormality Assumption, OA)**.** *For an LV given by $(\mathbf{p}, \mathbf{c}, \Phi, N)$, we assume that $\{\mathbf{v}^\ell\}_{1\le\ell\le k} \cup \{\mathbf{n}^j\}_{1\le j\le d-k}$ forms an orthonormal basis for $\mathbb{R}^d$ and $\mathbf{p} = N^T\mathbf{c}$ such that $\mathbf{p}^T\mathbf{v}^\ell = 0, \forall\ 1 \le \ell \le k$. This assumption also implies that the intercept is the point in the LV closest to the origin in terms of $\ell_2$ norm.*

**Example 1.2.** *In $\mathbb{R}^3$, $N = [-1, 7, -4]/\sqrt{66}$ and $c = 132/\sqrt{66}$ describe a plane. Taking $\mathbf{v}^1 = [3, 1, 1]^T/\sqrt{11}, \mathbf{v}^2 = [1, -1, -2]^T/\sqrt{6}$ and $\mathbf{p} = [-2, 14, -8]^T$, any point on the plane can be written as $\mathbf{p} + \mathbf{v}^1 t_1 + \mathbf{v}^2 t_2$ since $N\mathbf{p} = c$ and $\mathbf{v}^1, \mathbf{v}^2, N^T$ are orthogonal to each other.*

Additionally, we will try to get a general framework for the extension of EDMs, and in some of the derivations, it will be helpful to assume that the LVs we consider do not intersect.

**Lemma 1.6.** *In a d-dimensional affine space, a necessary condition for two LVs of dimension $k$ and $l$ respectively to not intersect is $k + l < d$.*

An important theorem to justify this claim is given below:

**Theorem 1.7** (Rouché-Capelli)**.** *The system $A\mathbf{x} = \mathbf{b}$ admits solutions, i.e. is consistent, if and only if rank$(A)$ = rank$([A, \mathbf{b}])$. Moreover, the solution is unique if and only if the rank of $A$ is equal to the number of columns of $A$.*

Consider two LVs $\mathbf{f}_1(\mathbf{t}) = \mathbf{p} + \Phi\mathbf{t}$ and $\mathbf{f}_2(\mathbf{s}) = \mathbf{q} + \Psi\mathbf{s}$ of dimension $k$ and $l$ respectively and an ambient space of dimension $d$. Without loss of generality, we assume $k \ge l$. The system to determine whether they intersect is of the form $\mathbf{q} - \mathbf{p} = [\Phi, -\Psi][\mathbf{t}^T, \mathbf{s}^T]^T$. By Theorem 1.7, a necessary condition for the LVs not to intersect is that this is either an overdetermined system, in which case $k + l < d$, or the columns of $\Psi$ are spanned by the columns of $\Phi$, in which case the LVs are parallel. This digression allows us to see that the dimension constraint $k + l < d$ is consistent with the OA, in fact this dimension constraint ensures that we have enough degrees of freedom for the OA, and a meaningful distance measure for the LVs as we will see in the next section. In the remaining of this text, we will assume $k + l < d$.

# Chapter 2

# Extension of EDMs to linear varieties: Definition and Properties

The review of the EDMs in the previous chapter is of crucial importance to extend the distance matrices into higher dimensional affine objects, which we will call extended EDMs (EEDM). This chapter builds a general framework for EEDMs and studies their properties, to allow us to understand the objects we are looking into, before asking ourselves whether inverse problems are feasible. In the case of EDMs, the entries of the matrix contains the squared $\ell_2$ norm between each pair of points, which is chosen as the metric. In EEDMs, the ambiguity is that the objects we consider are generally a set of infinite points and the metric must be more precise. To measure the distance between two sets, a common choice is the minimum $\ell_2$ norm, or least squares distance, which is the option we study in this text. We also choose to keep the distances squared, as in the EDM case. This is one of our main deviations from the model by Krekovic et al. [1] and Kreković et al. [2]. In these studies, the main objective is to reconstruct the shape of a room, in 2 or 3D. The distance matrix contains (non-squared) $\ell_2$ distances between points, which correspond to microphones and planes, which represent the walls, or lines, for the 2D case. In the practical applications, these microphones are always within the room walls, therefore the distance from the origin of the points will always be less than the minimum distance from the origin to the lines (in 2D) or planes (in 3D). Mathematically, the model is $d_{ij} = (\mathbf{p}_j - \mathbf{r}_i)^T \mathbf{n}_j$, where $\mathbf{r}_i$ describes the position of microphone $i$ and $\mathbf{n}_j$ is the normal unit norm vector of line/plane $j$ and $\mathbf{p}_j$ any point in that line/plane. This represents the minimum, non-squared, distance between a point and a line in 2D, a point and a plane in 3D, and more generally, a point and an LV of dimension $d-1$ in an ambient space of dimension $d$. However, to generalize this to any pair of LV, in general configurations and any given dimension, we need to use a different model.

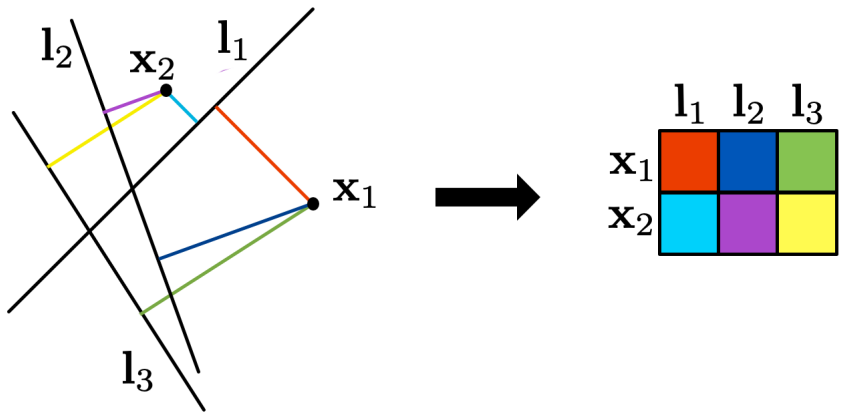
**Definition 2.1** (EEDM)**.** *Let $\{\mathcal{F}_i\}_{1\leq i\leq n}$ be a collection of $n$ LVs of dimension $k$, embedded in dimension $d$. Similarly, $\{\mathcal{G}_j\}_{1\leq j\leq m}$ contains $m$ LVs of dimension $l$. Then, we define the function eedm, such that $D = eedm\Big(\{\mathcal{F}_i\},\{\mathcal{G}_j\}\Big) \in \mathbb{R}^{n\times m}$ is the EEDM of the two collections of LVs. The entry $(i,j)$ of $D$ is given by:*

$$d_{ij} \triangleq d(\mathcal{F}_i,\mathcal{G}_j) \triangleq \min_{\mathbf{x}\in\mathcal{F}_i,\mathbf{y}\in\mathcal{G}_j} ||\mathbf{x}-\mathbf{y}||^2. \tag{2.1}$$

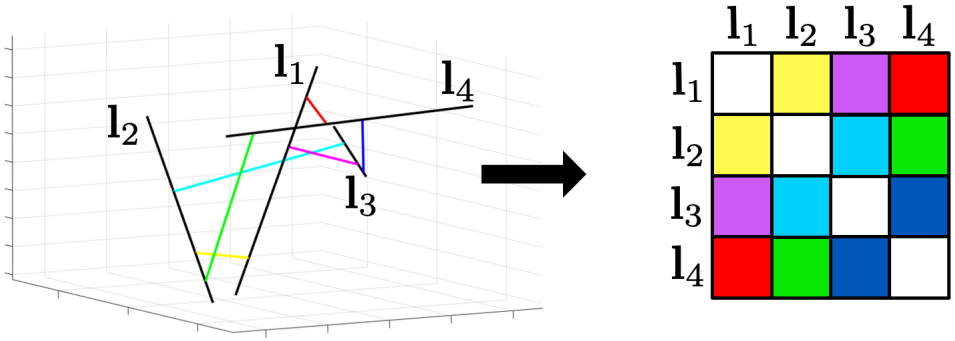
*On the other hand, if we want only the distances of a collection of LVs of the same dimension, we write $D = eedm(\{\mathcal{F}_i\}) \in \mathbb{R}^{n\times n}$, and:*

$$d_{ij} \triangleq d(\mathcal{F}_i,\mathcal{F}_j) \triangleq \min_{\mathbf{x}\in\mathcal{F}_i,\mathbf{y}\in\mathcal{F}_j} ||\mathbf{x}-\mathbf{y}||^2. \tag{2.2}$$

**Example 2.1.** *$d = 3, k = 0, l = 1, n = 3, m = 5$ corresponds to a setting where we have 3 points*




(a) Distance between a configuration of points $\mathbf{x}_i$ and lines $\mathbf{l}_j$ in 2D. Only the distances between objects of different configurations are kept, as in Equation 2.1. The matrix is computed using equation 2.10.




(b) Matrix of distances in between a configuration of lines in 3D, as described in Equation 2.2. The matrix is obtained using Equation 2.7.

Figure 2.1: Example of EEDMs.

*and 5 lines in 3 dimensions.* $d = 5, k = 1, l = 2, n = 7, m = 4$ *corresponds to a setting where we have 7 lines and 4 planes in 5 dimensions.*

**Observations:** In the special case of $k = 0$, $\{\mathcal{F}_i\}$ reduces to a set of points ascribed to the columns of the matrix $X \in \mathbb{R}^{d\times n}$ and the last equation gives $\text{eedm}(\{\mathcal{F}_i\}) = \text{edm}(X)$. Moreover, if $l$ is also 0, the first equation comes down to computing the distance between the points $X$ and the ones in $\{\mathcal{G}_j\}$, ascribed in the columns of $P \in \mathbb{R}^{d\times m}$, defined as:

$$\text{edm}(X, P) \triangleq \mathbf{1}_n \text{diag}(P^T P)^T + \text{diag}(X^T X)\mathbf{1}_m^T - 2X^T P \in \mathbb{R}^{n\times m}. \tag{2.3}$$

We also note that $\text{eedm}\Big(\{\mathcal{G}_j\}, \{\mathcal{F}_i\}\Big) = \text{eedm}\Big(\{\mathcal{F}_i\}, \{\mathcal{G}_j\}\Big)^T$.

**Notation.** *When considering multiple LVs of dimension* $k$*;* $\mathbf{f}_i(\mathbf{t}) = \mathbf{p}_i + \sum_{\ell=1}^{k} \mathbf{v}_i^\ell t_\ell$*, we will denote as* $P$ *the* $d \times n$ *matrix containing the intercepts of each LV in its columns, and* $V_\ell$ *the* $d \times n$

*matrix containing the $\ell$-th direction vector of each LV in its columns, i.e.* $P = [\mathbf{p}_1, \dots, \mathbf{p}_n]$ *and* $V_\ell = [\mathbf{v}_1^\ell, \dots, \mathbf{v}_n^\ell]$.

In the general case, the **Positivity** property is guaranteed since we consider non-intersecting LVs. However, **Symmetry** is obtained only in the case of Equation 2.2 and there is no **Triangle Inequality** property anymore. As a side note, we do not require objects for which we do not compute the distance to be non-intersecting, for example in Figure 2.1a, the lines cross each other.

## 2.1 Minimum distance and special cases

Intuitively, these distances should be well defined, since there should exist a pair of points where the LVs are closest. Formally, this problem has been studied by DuPré and Kass [41], Ben-Israel and Greville [42] and Gross and Trenkler [43]. The former two make the OA assumption, whereas the latter generalizes the solution.

**Theorem 2.1** (Minimum distance, DuPré and Kass [41])**.** *Under the OA, consider an LV $\mathcal{F}$ of dimension $k$ described by $(\mathbf{p}, \Phi)$ and a distinct LV $\mathcal{G}$ of dimension $l$ given by $(\mathbf{q}, \Psi)$, such that $\mathcal{F} \cap \mathcal{G} = \emptyset$. Then:*

$$d(\mathcal{F}, \mathcal{G}) = ||C(\mathbf{p} - \mathbf{q})||^2 = ||C(\mathbf{x} - \mathbf{y})||^2 \tag{2.4}$$

$\forall \mathbf{x} \in \mathcal{F}, \forall \mathbf{y} \in \mathcal{G}$*, where $C \neq 0$ and its rows form an orthonormal basis of $\mathcal{N}(\Phi^T) \cap \mathcal{N}(\Psi^T)$.*

**Remark.** *If the context requires us to work with the implicit equation of LVs; $\mathcal{F} = \{\mathbf{x} : N\mathbf{x} = \mathbf{c}\}$ and $\mathcal{G} = \{\mathbf{y} : M\mathbf{y} = \mathbf{a}\}$, we have $\mathcal{N}(\Phi^T) = \mathcal{R}(\Phi)^\perp = \mathcal{N}(N)^\perp = \mathcal{R}(N^T)$, therefore we require the rows of $C$ to form an orthonormal basis of $\mathcal{R}(N^T) \cap \mathcal{R}(M^T)$. Then, we may write $d(\mathcal{F}, \mathcal{G}) = ||C(N^T\mathbf{c} - M^T\mathbf{a})||^2$.*

*Proof.* The parametric equations of $\mathcal{F}$ and $\mathcal{G}$ are $\mathbf{f}(\mathbf{t}) = \mathbf{p} + \Phi\mathbf{t}$ and $\mathbf{g}(\mathbf{s}) = \mathbf{q} + \Psi\mathbf{s}$ respectively. Then:

$$\begin{aligned} d(\mathcal{F}, \mathcal{G}) = \min_{\mathbf{t},\mathbf{s}} ||\mathbf{p} + \Phi\mathbf{t} - \mathbf{q} - \Psi\mathbf{s}||^2 &= \min_{\mathbf{t},\mathbf{s}} ||\mathbf{p} - \mathbf{q}||^2 + ||\Phi\mathbf{t} - \Psi\mathbf{s}||^2 + 2(\mathbf{p} - \mathbf{q})^T(\Phi\mathbf{t} - \Psi\mathbf{s}) \\ &= \min_{\mathbf{u}} ||\mathbf{p} - \mathbf{q}||^2 + ||A\mathbf{u}||^2 + 2(\mathbf{p} - \mathbf{q})^T(A\mathbf{u}), \end{aligned}$$

where $A = [\Phi, -\Psi] \in \mathbb{R}^{d\times(k+l)}$ and $\mathbf{u} = [\mathbf{t}^T, \mathbf{s}^T]^T \in \mathbb{R}^{k+l}$. This is a quadratic polynomial in $\mathbf{u}$ and convex since $A^TA \succeq 0$. Taking the gradient with respect to $\mathbf{u}$ and forcing it to $\mathbf{0}$ gives:

$$A^TA\mathbf{u}^* = A^T(\mathbf{q} - \mathbf{p}).$$

Let $P_{\mathcal{I}}$ denote the orthogonal projection onto the set $\mathcal{I}$. Then,

$$\mathbf{q} - \mathbf{p} = P_{\mathcal{R}(A)}(\mathbf{q} - \mathbf{p}) + P_{\mathcal{N}(A^T)}(\mathbf{q} - \mathbf{p}),$$

hence, $A^T(\mathbf{q} - \mathbf{p}) = A^TP_{\mathcal{R}(A)}(\mathbf{q} - \mathbf{p})$. Since $A\mathbf{u}^*$ and $P_{\mathcal{R}(A)}(\mathbf{q} - \mathbf{p})$ both belong to $\mathcal{R}(A)$, we have the equality $A\mathbf{u}^* = P_{\mathcal{R}(A)}(\mathbf{q} - \mathbf{p})$. Replacing this in the norm, we have:

$$d(\mathcal{F}, \mathcal{G}) = ||(I - P_{\mathcal{R}(A)})(\mathbf{p} - \mathbf{q})||^2.$$

Moreover, $I - P_{\mathcal{R}(A)} = P_{\mathcal{R}(A)^\perp}$ and $\mathcal{R}(A)^\perp = \mathcal{N}(A^T) = \mathcal{N}(\Phi^T) \cap \mathcal{N}(\Psi^T)$. Since $P_{\mathcal{R}(A)^\perp}\Phi = P_{\mathcal{R}(A)^\perp}\Psi = 0$,

$$d(\mathcal{F}, \mathcal{G}) = ||P_{\mathcal{R}(A)^\perp}(\mathbf{p} - \mathbf{q})||^2 = ||P_{\mathcal{R}(A)^\perp}(\mathbf{x} - \mathbf{y})||^2$$

$\forall \mathbf{x} \in \mathcal{F}, \forall \mathbf{x} \in \mathcal{G}$. Taking $C : C^TC = P_{\mathcal{R}(A)^\perp}$, we obtain the desired result since $CC^T = I$.

□

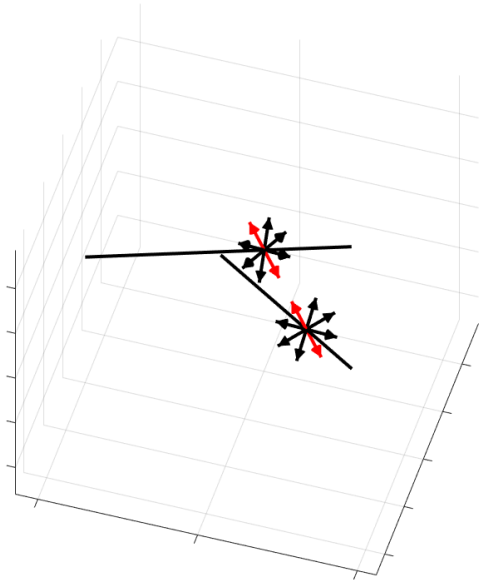


Figure 2.2: The two lines in this figure are the objects for which we compute the minimum distance. The arrows are examples of vectors orthogonal to the respective lines, i.e belonging to $\mathcal{N}(\Phi^T)$ for line 1 and $\mathcal{N}(\Psi^T)$ for line 2. The red arrows are the vectors which additionally also belong to $\mathcal{N}(\Phi^T) \cap \mathcal{N}(\Psi^T)$. The vector connecting the minimum distance points will be a linear combination of these red vectors (for lines a basis describing such vectors contains only one vector in 3D).

Geometrically, this theorem tells us that the vector connecting the LVs at the minimum distance is orthogonal to both of these objects; an example is provided in Figure 2.2. Even though it is helpful to have a closed-form solution, finding $C$ is not trivial in the general case. We therefore develop below some of the interesting cases that arise, where the solution is given by a more explicit expression.

**Parallel LVs:** Assuming $k \geq l$, if the LVs are parallel, $\mathcal{N}(\Phi^T) \cap \mathcal{N}(\Psi^T) = \mathcal{N}(\Psi^T)$. The rows of the matrix $M$ constitute an orthonormal basis for $\mathcal{N}(\Psi^T)$, where $M : \mathcal{G} = \{\mathbf{x} : M\mathbf{x} = \mathbf{a}\}$, and $\mathbf{a}$ is chosen conveniently. This leads to $d(\mathcal{F}, \mathcal{G}) = ||M(\mathbf{p} - \mathbf{q})||^2$.

**Linearly independent LVs:** In this case, we may write $\dim(\mathcal{R}(\Phi) + \mathcal{R}(\Psi)) = k + l$, which implies that $A$ is full column rank. The optimal parameters can then be written as $\mathbf{u}^* = (A^T A)^{-1} A^T (\mathbf{q} - \mathbf{p})$, therefore $d(\mathcal{F}, \mathcal{G}) = ||(I - A(A^T A)^{-1} A^T)(\mathbf{p} - \mathbf{q})||^2$. We usually encounter this case when $\mathcal{F}$ and $\mathcal{G}$ are non parallel lines, i.e. $\Phi = \boldsymbol{\phi}$ and $\Psi = \boldsymbol{\psi}$ are linearly independent vectors. Then, $A^T A$ reduces to a $2 \times 2$ matrix:

$$A^T A = \begin{pmatrix} 1 & -\boldsymbol{\phi}^T \boldsymbol{\psi} \\ -\boldsymbol{\phi}^T \boldsymbol{\psi} & 1 \end{pmatrix} \Rightarrow (A^T A)^{-1} = \frac{1}{1 - (\boldsymbol{\phi}^T \boldsymbol{\psi})^2} \begin{pmatrix} 1 & \boldsymbol{\phi}^T \boldsymbol{\psi} \\ \boldsymbol{\phi}^T \boldsymbol{\psi} & 1 \end{pmatrix}$$

and:

$$(\mathbf{p} - \mathbf{q})^T A (A^T A)^{-1} A^T (\mathbf{p} - \mathbf{q}) = \frac{1}{1 - (\boldsymbol{\phi}^T \boldsymbol{\psi})^2} \begin{pmatrix} -\mathbf{q}^T \boldsymbol{\phi} & -\mathbf{p}^T \boldsymbol{\psi} \end{pmatrix} \begin{pmatrix} 1 & \boldsymbol{\phi}^T \boldsymbol{\psi} \\ \boldsymbol{\phi}^T \boldsymbol{\psi} & 1 \end{pmatrix} \begin{pmatrix} -\boldsymbol{\phi}^T \mathbf{q} \\ -\boldsymbol{\psi}^T \mathbf{p} \end{pmatrix}.$$

Observing that $d(\mathcal{F}, \mathcal{G})$ can be written as $||\mathbf{p} - \mathbf{q}||^2 - (\mathbf{p} - \mathbf{q})^T A (A^T A)^{-1} A^T (\mathbf{p} - \mathbf{q})$ and developing the expression above, **the distance between two non parallel and non intersecting lines** is:

$$d(\mathcal{F}, \mathcal{G}) = ||\mathbf{p} - \mathbf{q}||^2 - \frac{1}{1 - (\boldsymbol{\phi}^T \boldsymbol{\psi})^2} \left( 2(\boldsymbol{\phi}^T \boldsymbol{\psi})(\boldsymbol{\psi}^T \mathbf{p})(\boldsymbol{\phi}^T \mathbf{q}) + (\boldsymbol{\psi}^T \mathbf{p})^2 + (\boldsymbol{\phi}^T \mathbf{q})^2 \right) \tag{2.5}$$

$$= ||\mathbf{p} - \mathbf{q}||^2 - t^* \boldsymbol{\phi}^T \mathbf{q} - s^* \boldsymbol{\psi}^T \mathbf{p}, \tag{2.6}$$

where

$$t^* = \frac{\boldsymbol{\phi}^T\mathbf{q} + (\boldsymbol{\phi}^T\boldsymbol{\psi})(\boldsymbol{\psi}^T\mathbf{p})}{1-(\boldsymbol{\phi}^T\boldsymbol{\psi})^2},\ s^* = \frac{\boldsymbol{\psi}^T\mathbf{p} + (\boldsymbol{\phi}^T\boldsymbol{\psi})(\boldsymbol{\phi}^T\mathbf{q})}{1-(\boldsymbol{\phi}^T\boldsymbol{\psi})^2}$$

are the parameters such that the point $\mathbf{p}+\boldsymbol{\phi}t^*$ on line $\mathcal{F}$ and the point $\mathbf{q}+\boldsymbol{\psi}s^*$ on line $\mathcal{G}$ are the pair minimizing the distance between the lines.

By inspection, when considering a set of $n$ lines $\{\mathcal{F}_i\}_{1\leq i\leq n}$, the EEDM is therefore:

$$\text{eedm}(\{\mathcal{F}_i\}) = \text{edm}(P) - T\circ(V^TP) - T^T\circ(P^TV) \in \mathbb{R}^{n\times n}, \tag{2.7}$$

where $T = [(P^TV)\circ(V^TV) + V^TP] \oslash [\mathbf{1}_n\mathbf{1}_n^T - (V^TV)^{\circ 2}] \in \mathbb{R}^{n\times n}$ and $\text{diag}(T) = \mathbf{0}$. This matrix contains the parameters $t_{ij}$ for which the point $\mathbf{p}^i+\mathbf{v}^it_{ij}$ of line $i$ is the closest one among any other point in line $i$ to line $j$. In this last expression, "$\oslash$" and "$(\cdot)^{\circ 2}$" correspond to the component-wise division and square respectively.

Equation 2.7 is the equivalent of EDMs for lines, but we see that the expression is much more complicated in the sense that it contains many non-linear, component-wise products. We also note that without a particular structure of the lines, the matrix in 2.7 is full rank. Extending this to planes is even harder because it may not be unusual for two planes to have a direction in common, i.e. $A = [\Phi, -\Psi]$ is not full column rank, therefore it is not trivial to find the matrix $C$ without assumptions or prior knowledge about the positioning of the planes.

On the other hand, another option would be for $A$ to have orthogonal rows, which leads to $A^TA = I$, in which case a similar development results in $d(\mathcal{F},\mathcal{G}) = ||\mathbf{p}-\mathbf{q}||^2 - ||\Phi^T\mathbf{q}||^2 - ||\Psi^T\mathbf{p}||^2$. In the general setting, this is a rare case, except for;

**One LV is a point:** Suppose $\mathcal{G} = \{\mathbf{q}\}$, then $A = \Phi$ and:

$$d(\{\mathbf{q}\},\mathcal{F}) = ||\mathbf{q}-\mathbf{p}||^2 - ||\Phi^T\mathbf{q}||^2 = ||\mathbf{q}-\mathbf{p}||^2 - \sum_{\ell=1}^{k}((\mathbf{v}^\ell)^T\mathbf{q})^2. \tag{2.8}$$

The parameter $\mathbf{t}\in\mathbb{R}^k$ of $\mathcal{F}$ for which the distance is minimum is $\mathbf{t}^* = \Phi^T\mathbf{q}$. This is justified by taking the gradient of $||\mathbf{p}+\Phi\mathbf{t}-\mathbf{q}||^2$ with respect to $\mathbf{t}$ and putting it to $\mathbf{0}$ since it is a quadratic convex function on $\mathbf{t}$ ($\Phi^T\Phi = I \succeq 0$). Then:

$$\nabla_{\mathbf{t}}||\mathbf{p}+\Phi\mathbf{t}-\mathbf{q}||^2 = 2\Phi^T\Phi\mathbf{t} + 2\Phi^T(\mathbf{p}-\mathbf{q}).$$

By the OA, this results in $\mathbf{t}^* = \Phi^T\mathbf{q}$ and the point $\widetilde{\mathbf{p}}$ on $\mathcal{F}$ minimizing the distance is:

$$\widetilde{\mathbf{p}} = \mathbf{p} + \Phi\Phi^T\mathbf{q}. \tag{2.9}$$

Generalizing the result of Equation 2.8 to $n$ points $\mathbf{x}_i$ ascribed in the columns of $X$ and $m$ LVs $\mathcal{F}_j = \{\mathbf{p}_j + \sum_{\ell=1}^{k}\mathbf{v}_j^\ell t_\ell\}$ of dimension $k$, the EEDM can be expressed as:

$$\text{eedm}\Big(\{\mathbf{x}_i\},\{\mathcal{F}_j\}\Big) = \text{edm}(X,P) - \sum_{\ell=1}^{k}(X^TV_\ell)^{\circ 2} \in \mathbb{R}^{n\times m}, \tag{2.10}$$

where $P = [\mathbf{p}_1,\ldots,\mathbf{p}_m]$ and $V_\ell = [\mathbf{v}_1^\ell,\ldots,\mathbf{v}_m^\ell]$, i.e. the $\ell$-th direction vectors of each LV.

**Intersecting LVs:** Even though we assumed non intersecting LVs, we also discuss this possibility for the sake of completeness. DuPré and Kass [41] prove that $\mathcal{F}$ and $\mathcal{G}$ intersect if and only if $C = 0$ or $\mathbf{p}-\mathbf{q}\in\mathcal{N}(C)$. Geometrically, these conditions imply that either the LVs intersect in the usual way, or, considering $k\geq l$, that $\mathcal{G}\subseteq\mathcal{F}$.

## 2.2 Invariance

As in the case of points, linear varieties also have different configurations that result in the same distance matrix, and again the information on the absolute position is lost and only the relative information is kept. Intuitively, two configurations of LVs differing by a rigid motion should give the same EEDM, because the distances are not affected by "the position from where we look" to compute them.

**Lemma 2.2.** *Consider the two following configurations: $\{\mathcal{F}_i\}$ containing $n$ LVs of dimension $k$ and $\{\mathcal{G}_j\}$ having $m$ LVs of dimension $l$, embedded in an ambient space of dimension $d$. Then:*

$$eedm\Big(\{\mathcal{F}_i\},\{\mathcal{G}_j\}\Big) = eedm\Big(\{\widetilde{\mathcal{F}}_i\},\{\widetilde{\mathcal{G}}_j\}\Big), \tag{2.11}$$

*where $\{\widetilde{\mathcal{F}}_i\}$ and $\{\widetilde{\mathcal{G}}_j\}$ are obtained by applying **the same** rigid motion to both initial configurations. This result also applies for only one configuration: $eedm(\{\mathcal{F}_i\}) = eedm(\{\widetilde{\mathcal{F}}_i\})$.*

*Proof.* By Theorem 2.1, each entry of the EEDMs is of the form $||C(\mathbf{x}-\mathbf{y})||^2$, where $\mathbf{x}$ and $\mathbf{y}$ are points on different LVs. Consider the orthogonal matrix $R \in \mathbb{R}^{d\times d}$ representing rotation/reflection, and a vector $\mathbf{b} \in \mathbb{R}^d$ for translation. Assuming, without loss of generality, that the translation is applied first, any point $\mathbf{x}$ on any LV of the configurations becomes $R\mathbf{x}+R\mathbf{b}$ after the transformation. Therefore:

$$||C(R\mathbf{x}+R\mathbf{b}-R\mathbf{y}-R\mathbf{b})||^2 = ||CR(\mathbf{x}-\mathbf{y})||^2 = ||C(\mathbf{x}-\mathbf{y})||^2.$$

Since $R^TR = I$ and $C$ has orthonormal rows, $CR$ just changes the basis of the rows of $C$, and keeping $\mathcal{R}(C^T) = \mathcal{R}(R^TC^T)$. Since any basis would do, the two last terms above are indeed equal, which completes the proof. □

In terms of the parametric equation, we can represent rigid transformations of an LV given by the pair $(\mathbf{p},\Phi)$ as having transformed intercept and direction vectors $(R\mathbf{p}+R\mathbf{b}, R\Phi)$. Here, $R$ is an orthogonal matrix and $\mathbf{b}$ is a vector representing rotation/reflection and translation respectively. Note that the translation is only applied to the intercept vectors whereas **the same** rotation/reflection is applied to both the intercepts and the direction vectors. This is an important result for reconstructing the original configuration from EEDM measurements. In the context of matching the original configuration and the one we have reconstructed from EEDM measurements, the Orthogonal Procrustes algorithm (Algorithm 1) can be used on the intercepts given by $P$. Then, the rotation/reflection matrix obtained from this process can be applied to the direction vectors to have an estimation of the original LVs. This procedure comes of course after estimating the configuration from EEDM measurements, which will be the main topic of Chapter 3.

**Remark.** *We can also see this result by replacing $X$ and $P$ by $RX+\mathbf{b}\mathbf{1}_n^T$ and $RP+\mathbf{b}\mathbf{1}_n^T$ respectively and $V$ by $RV$ in Equation 2.10 or 2.7 for example. However, we observe that we need to have $\mathbf{b} \in \mathcal{N}(\Phi^T)$. This is often too constraining and sometimes not even possible. At first, this might seem contradictory to what we discussed previously, but it is not the case; it is simply a consequence of the OA. Indeed, since we cannot take any intercept $\mathbf{p}$ given $\Phi$ because of the OA, but need to satisfy $\Phi^T\mathbf{p} = \mathbf{0}$, we also need to satisfy this property for the translation vectors. Therefore, this constraint is only present because of the OA and is reflected on the closed form solutions in these equations mentioned above. Mathematically, translating the overall configuration by a vector $\mathbf{b}$ can be represented by adding a different translation vector $\widetilde{\mathbf{b}}_j$ to each LV $j$, where $\widetilde{\mathbf{b}}_j = \frac{\mathbf{b}^T\mathbf{p}_j}{||\mathbf{p}_j||^2}\mathbf{p}_j$ is the orthogonal projection of $\mathbf{b}$ onto the intercept of LV $j$.*

We end this discussion by noting that there exists other, non-rigid, invariances for EEDMs, as presented in detail by Krekovic et al. [39] (Theorem 1). For instance, in 2D, for distances between

points $\mathbf{x}_i$ and lines given by the pairs $(\mathbf{p}_j, \Phi_j)$, such that $||\mathbf{x}_i||^2 < ||\mathbf{p}_j||^2$ (supposing the configuration is centered at the origin), the distance matrix is the same if and only if the points are collinear, the lines enclose a parallelogram, or every line is parallel. In 3D, for distances between points $\mathbf{x}_i$ and planes given by the pairs $(\mathbf{p}_j, \Phi_j)$, such that $||\mathbf{x}_i||^2 < ||\mathbf{p}_j||^2$, if there are $m < 6$ planes, there are infinitely many configurations that result in the same distance matrix. If $m \geq 6$, then the points must be coplanar or the configuration must belong to one of the specific classes presented in Figure 1 in [39] for this to happen. We consider general configurations and generally many measurements to only consider invariance to rigid motion, and leave the generalization of invariance other than rigid motion, in any dimension, and for any type of LV for a future study.

## 2.3 Rank

In this section we look at the rank properties of EEDMs, motivated by the rank constraints of EDMs. Unfortunately, the minimum distance between LVs given in Equation 2.4 depends on the matrix $C$ which differs in general for each pair of LV. This is seen in the following conjecture made by Ivan Dokmanic in a private correspondence.

**Conjecture 2.3** (General rank, Ivan Dokmanic)**.** *For $n$ LVs of dimension $k$ in the configuration $\{\mathcal{F}_i\}$ embedded in an ambient space of dimension $d$, if $k = 0$, rank$\Big($eedm$\Big(\{\mathcal{F}_i\}\Big)\Big) \leq d + 2$. Otherwise, eedm$\Big(\{\mathcal{F}_i\}\Big)$ is full rank in the general setting. Additionally, considering $m$ LVs of dimension $l > 0$ $\{\mathcal{G}_j\}$, eedm$\Big(\{\mathcal{F}_i\}, \{\mathcal{G}_j\}\Big)$ is full rank in the general setting if $k > 0$.*

*Sketch of Proof.* Let us denote by $\mathbf{x}_i = \mathbf{p}_i + \Phi_i t_i$ a point belonging to $\mathcal{F}_i$ and $\mathbf{y}_j = \mathbf{q}_j + \Psi_j t_j$ a point belonging to $\mathcal{G}_j$, for $1 \leq i \leq n$ and $1 \leq j \leq m$. We also define $C_{ij}$, a matrix such that its rows are an orthonormal basis for $\mathcal{N}(\Phi_i^T) \cap \mathcal{N}(\Psi_j^T)$. Then, $d(\mathcal{F}_i, \mathcal{G}_j) = ||C_{ij}(\mathbf{x}_i - \mathbf{y}_j)||^2$ and the EEDM $D \in \mathbb{R}^{n \times m}$ can be written as:

$$D = \begin{pmatrix} ||C_{11}\mathbf{x}_1||^2 + ||C_{11}\mathbf{y}_1||^2 - 2\mathbf{x}_1^T C_{11}^T C_{11}\mathbf{y}_1 & \cdots & ||C_{1m}\mathbf{x}_1||^2 + ||C_{1m}\mathbf{y}_m||^2 - 2\mathbf{x}_1^T C_{1m}^T C_{1m}\mathbf{y}_m \\ ||C_{21}\mathbf{x}_2||^2 + ||C_{21}\mathbf{y}_1||^2 - 2\mathbf{x}_2^T C_{21}^T C_{21}\mathbf{y}_1 & \cdots & ||C_{2m}\mathbf{x}_2||^2 + ||C_{2m}\mathbf{y}_m||^2 - 2\mathbf{x}_2^T C_{2m}^T C_{2m}\mathbf{y}_m \\ \vdots & \ddots & \vdots \\ ||C_{n1}\mathbf{x}_n||^2 + ||C_{n1}\mathbf{y}_1||^2 - 2\mathbf{x}_n^T C_{n1}^T C_{n1}\mathbf{y}_1 & \cdots & ||C_{nm}\mathbf{x}_n||^2 + ||C_{nm}\mathbf{y}_m||^2 - 2\mathbf{x}_n^T C_{nm}^T C_{nm}\mathbf{y}_m \end{pmatrix}$$

This expression is similar to EDMs except that each inner product is weighted by a different matrix $C_{ij}$, depending on the row and column. This high dependency on the pair of LVs we choose to compute the minimum distance makes the EEDM full rank without very specific and too constraining assumptions on the configurations. However, formulating the precise assumptions to guarantee that $D$, given above, is full rank is not trivial and so this result remains a conjecture—at least for now.

The results of this conjecture can also be verified in practice. For example, the EEDM for lines given by Equation 2.7 has rank $n$ for configurations generated randomly. This seems to be an early bad news for rank properties. However, we omitted one case in this theorem, the one for which one of the LV configurations is a set of points, thus $k = 0$ and/or $l = 0$. Indeed, we will show that this type of EEDM is rank deficient, and to show that we need to come back to Equation 2.10. The first term, edm$(X, P)$ has rank less than $d+2$, because it is a sum of two outer products and a matrix of rank $d$ (Equation 2.3). The other term is less trivial but it can be shown that it is rank deficient too, as detailed in the following lemma.

**Lemma 2.4.** *Let $A \in \mathbb{R}^{n \times m}$ be a matrix of rank $d$. Then rank$(A^{\circ 2}) \leq \binom{d+1}{2}$.*

*Proof.* As shown in Styan [44] using SVD, the rank of a Hadamard product is less than the product of the rank of the two matrices. We use a similar approach here for the component-wise square of a matrix. Since $A$ has rank $d$, it can be written as $\sum_{i=1}^{d} \sigma_i \mathbf{u}_i \mathbf{w}_i^T$, where the $\mathbf{u}_i$'s and $\mathbf{w}_i$'s are the left and right singular vectors of $A$ respectively and $\sigma_i$'s the singular values. Then:

$$A^{\circ 2} = \left(\sum_{i=1}^{d} \sigma_i \mathbf{u}_i \mathbf{w}_i^T\right) \circ \left(\sum_{j=1}^{d} \sigma_j \mathbf{u}_j \mathbf{w}_j^T\right) = \sum_{i=1}^{d}\sum_{j=1}^{d} \sigma_i \sigma_j (\mathbf{u}_i \mathbf{w}_i^T) \circ (\mathbf{u}_j \mathbf{w}_j^T).$$

Observing that $(\mathbf{u}_i \mathbf{w}_i^T) \circ (\mathbf{u}_j \mathbf{w}_j^T) = (\mathbf{u}_i \circ \mathbf{u}_j)(\mathbf{w}_i \circ \mathbf{w}_j)^T$, we write:

$$A^{\circ 2} = \sum_{i=1}^{d}\sum_{j=1}^{d} \sigma_i \sigma_j (\mathbf{u}_i \circ \mathbf{u}_j)(\mathbf{w}_i \circ \mathbf{w}_j)^T.$$

Since $\mathbf{u}_i \circ \mathbf{u}_j = \mathbf{u}_j \circ \mathbf{u}_i$, there are $\binom{d}{2}$ terms in this double sum that can be combined with the corresponding $\binom{d}{2}$ terms. At the end, the double sum contains at most $d^2 - \binom{d}{2} = \binom{d+1}{2}$ distinct outer products, therefore $\operatorname{rank}(A^{\circ 2}) \leq \binom{d+1}{2}$. □

Looking at Equation 2.10, this result is sufficient to say that the EEDM containing distances between points $\{\mathbf{x}_i\}$ and LVs $\{\mathcal{F}_j\}$ of dimension $k$ are rank deficient. Using previous results, $\operatorname{rank}\Big(\operatorname{eedm}\Big(\{\mathbf{x}_i\}, \{\mathcal{F}_j\}\Big)\Big) \leq d + 2 + k\binom{d+1}{2}$. However, the expression on the right might quickly become large, but fortunately, this bound is not tight. Indeed, simple tests confirm that the rank is much less than this expression. Surprisingly, the rank is in fact independent of $k$.

**Theorem 2.5.** *Consider $n$ points $\{\mathbf{x}_i\}$ and $m$ LVs $\{\mathcal{F}_j\}$ of dimension $k$. Then,*

$$\operatorname{rank}\Big(\operatorname{eedm}\Big(\{\mathbf{x}_i\}, \{\mathcal{F}_j\}\Big)\Big) \leq d + 1 + \binom{d+1}{2}. \tag{2.12}$$

*Proof.* Let $X = \sum_{\alpha=1}^{d} \sigma_\alpha(X) \mathbf{u}_\alpha(X) \mathbf{w}_\alpha(X)^T$ and $V_\ell = \sum_{\beta=1}^{d} \sigma_\beta(V_\ell) \mathbf{u}_\beta(V_\ell) \mathbf{w}_\beta(V_\ell)^T$, corresponding to the SVD decomposition. Then, $X^T V_\ell$ can be written as $\sum_{\alpha=1}^{d}\sum_{\beta=1}^{d} \kappa(\alpha, \beta, \ell) \mathbf{w}_\alpha(X) \mathbf{w}_\beta(V_\ell)^T$, where the coefficients $\kappa(\alpha, \beta, \ell) = \sigma_\alpha(X)\sigma_\beta(V_\ell)(\mathbf{u}_\alpha(X)^T \mathbf{u}_\beta(V_\ell))$. Similarly to the expression in the proof of the previous lemma, we have:

$$(X^T V_\ell)^{\circ 2} = \sum_{\alpha=1}^{d}\sum_{\beta=1}^{d}\sum_{\gamma=1}^{d}\sum_{\delta=1}^{d} \kappa(\alpha, \beta, \ell)\kappa(\gamma, \delta, \ell)(\mathbf{w}_\alpha(X) \circ \mathbf{w}_\gamma(X))(\mathbf{w}_\beta(V_\ell) \circ \mathbf{w}_\delta(V_\ell))^T.$$

Hence

$$\begin{aligned}\sum_{\ell=1}^{k} (X^T V_\ell)^{\circ 2} &= \sum_{\ell=1}^{k} \sum_{\alpha,\beta,\gamma,\delta} \kappa(\alpha, \beta, \ell)\kappa(\gamma, \delta, \ell)(\mathbf{w}_\alpha(X) \circ \mathbf{w}_\gamma(X))(\mathbf{w}_\beta(V_\ell) \circ \mathbf{w}_\delta(V_\ell))^T \\ &= \sum_{\alpha,\gamma} (\mathbf{w}_\alpha(X) \circ \mathbf{w}_\gamma(X)) \left(\sum_{\ell=1}^{k}\sum_{\beta,\delta} \kappa(\alpha, \beta, \ell)\kappa(\gamma, \delta, \ell)(\mathbf{w}_\beta(V_\ell) \circ \mathbf{w}_\delta(V_\ell))\right)^T.\end{aligned}$$

On the other hand, using a similar computation, $X^T X = \sum_{\alpha=1}^{d}\sum_{\gamma=1}^{d} \kappa(\alpha, \gamma) \mathbf{w}_\alpha(X) \mathbf{w}_\gamma(X)^T$, where this time $\kappa(\alpha, \gamma) = \sigma_\alpha(X)\sigma_\gamma(X)(\mathbf{u}_\alpha(X)^T \mathbf{u}_\gamma(X))$. Observing that for two vectors of the same length $\mathbf{w}_1, \mathbf{w}_2$, $\operatorname{diag}(\mathbf{w}_1 \mathbf{w}_2^T) = \mathbf{w}_1 \circ \mathbf{w}_2$, we may write:

$$\operatorname{diag}(X^T X)\mathbf{1}_m^T = \sum_{\alpha=1}^{d}\sum_{\gamma=1}^{d} \kappa(\alpha, \gamma)(\mathbf{w}_\alpha(X) \circ \mathbf{w}_\gamma(X))\mathbf{1}_m^T.$$

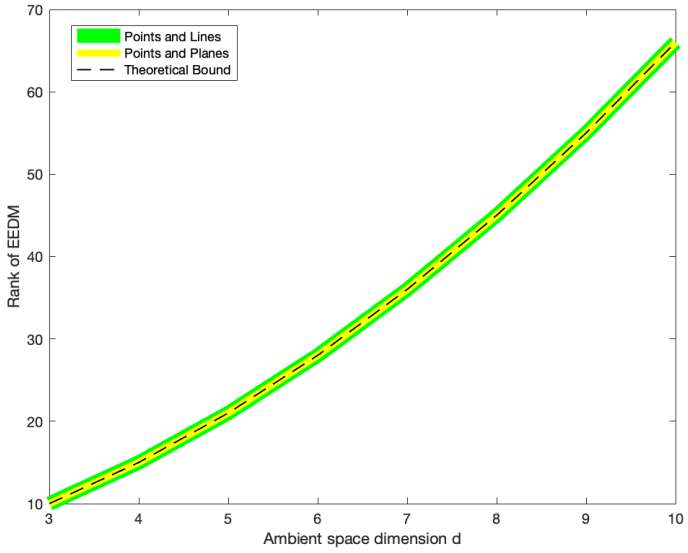



Figure 2.3: We see a comparison between the rank upper bound we obtained from Theorem 2.5 with respect to the dimension of the ambient space $d$. The green curve shows the EEDM rank obtained from computing the distance between points and lines. The yellow curve is computed similarly but lines are replaced by planes. Each value obtained for a dimension is averaged over 100 tests, and the number $n$ of points and $m$ of lines/planes is a random integer between 100 and 120. Other numbers of measurements still give the same result, as long as we satisfy $\min(n, m) \geq d+1+\binom{d+1}{2}$. Every linear variety used for the computations is generated randomly.

Combining these last two results, we have an expression of the form:

$$\operatorname{diag}(X^T X)\mathbf{1}_m^T - \sum_{\ell=1}^{k}(X^T V_\ell)^{\circ 2} = \sum_{\alpha=1}^{d}\sum_{\gamma=1}^{d}(\mathbf{w}_\alpha(X) \circ \mathbf{w}_\gamma(X))\mathbf{w}(\alpha,\gamma)^T,$$

where $\mathbf{w}(\alpha, \gamma)$ is the vector resulting from the combination of the expressions. As in the case of the previous lemma, there is at most $\binom{d+1}{2}$ distinct vectors $\mathbf{w}_\alpha(X) \circ \mathbf{w}_\gamma(X)$ in the double sum, which finally leads to $\operatorname{rank}\left(\operatorname{diag}(X^T X)\mathbf{1}_m^T - \sum_{\ell=1}^{k}(X^T V_\ell)^{\circ 2}\right) \leq \binom{d+1}{2}$. Since the sum of remaining two terms in the computation of the EEDM has at most rank $d+1$, we obtain the desired result. □

In fact, in the general setting, this upper bound is met with equality, as shown in Figure 2.3. For this to hold, we need of course $d+1+\binom{d+1}{2} \leq \min(n, m)$ since the EEDM has size $n \times m$. This means that to benefit from this upper bound, we need a large number of measurements and/or low ambient space dimension. Analogously to the EDM rank bound presented in Theorem 1.4, some interesting special cases arise when the points and LVs are arranged in a special configuration. Looking back at the derivation of the upper bound in Theorem 2.5, we assumed that $\operatorname{rank}(X) = \operatorname{rank}(P) = d$. Therefore, the upper bound on the rank can further be reduced if $\operatorname{affdim}(X)$ or $\operatorname{affdim}(P)$ is less than $d$, where $X$ and $P$ are defined as in Equation 2.10. This occurs when the points or the intercept vectors of the LVs are positioned on an affine subspace of dimension less than $d$.

**Corollary 2.6.** *Consider $n$ points $\{\mathbf{x}_i\}$ and $m$ LVs $\{\mathcal{F}_j\}$ of dimension $k$, where the intercept vectors of the LVs are given by $\{\mathbf{p}_j\}$. Supposing affdim$(\{\mathbf{x}_i\}) = r$ and affdim$(\{\mathbf{p}_j\}) = r'$, then:*

$$\operatorname{rank}\Big(\operatorname{eedm}\big(\{\mathbf{x}_i\}, \{\mathcal{F}_j\}\big)\Big) \leq \min(r, r') + 1 + \binom{r+1}{2}. \tag{2.13}$$

*Furthermore, if the points* $\{\mathbf{x}_i\}$ *are on the relative boundary of an* $r$*-dimensional hypersphere, and the intercepts* $\{\mathbf{p}_j\}$ *are positioned on the boundary of an* $r'$*-dimensional hypersphere, we have:*

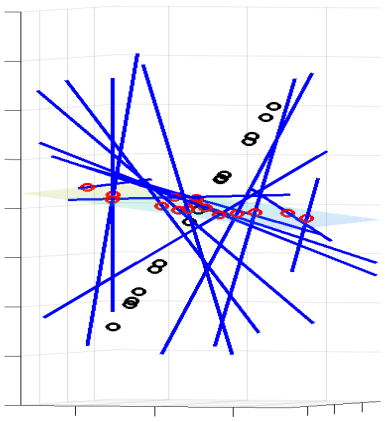
$$rank\Big(eedm\Big(\{\mathbf{x}_i\},\{\mathcal{F}_j\}\Big)\Big) \leq \min(r,r') + \binom{r+1}{2}. \tag{2.14}$$

*Moreover, this last inequality still holds if the intercepts* $\{\mathbf{p}_j\}$ *are such that affdim*$(\{\mathbf{p}_j\}) = r'$.

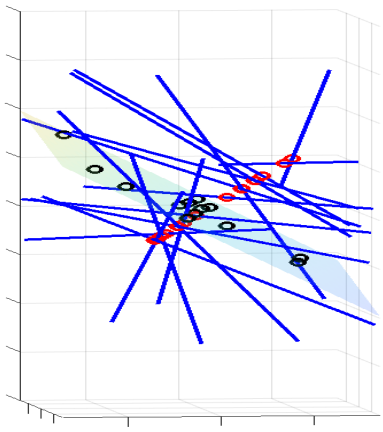
*Proof.* Again, referring to Equation 2.10, we have $\text{rank}(X^TP) \leq \min(r,r')$. Without loss of generality, we may write $X = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{w}_i^T$, i.e. the sum is over $r$ terms (If it is not the case, we could always find a rotation/reflection and translation so that we can, and by Lemma 2.2, we obtain the same EEDM). Replacing $d$ in the computations by $r$, we obtain the desired result.

For the hypersphere configurations, we start by first observing that the additional rank in Equation 2.13 compared to Equation 2.14 is due to the term $\mathbf{1}_n\text{diag}(P^TP)^T$ from Equation 2.10. We will show that $\text{rank}(\mathbf{1}_n\text{diag}(P^TP)^T + \text{diag}(X^TX)\mathbf{1}_m^T - (X^TV_\ell)^{\circ 2}) = \text{rank}((X^TV_\ell)^{\circ 2})$ in the context of Equation 2.14, for a fixed direction $\ell$. This is without loss of generality since the dimension of $\mathcal{F}_j'$'s does not affect the rank. First, we remember that $\text{rank}(\text{diag}(X^TX)\mathbf{1}_m^T - (X^TV_\ell)^{\circ 2}) = \text{rank}((X^TV_\ell)^{\circ 2})$. Then, we note that if the points $X$ lie on the relative boundary of an $r$-dimensional hypersphere, supposing of radius $\rho$, we have $\text{diag}(X^TX) = \rho^2\mathbf{1}_n$. Then, we may write $\mathbf{1}_n\text{diag}(P^TP)^T + \text{diag}(X^TX)\mathbf{1}_m^T = \text{diag}(X^TX)(\mathbf{1}_m + \frac{1}{\rho^2}\text{diag}(P^TP))^T$. As shown in the proof of the previous theorem, $\text{rank}(\text{diag}(X^TX)\mathbf{1}_m^T - (X^TV_\ell)^{\circ 2}) = \text{rank}((X^TV_\ell)^{\circ 2})$ because of the term $\text{diag}(X^TX)$, and independently of the term $\mathbf{1}_m$. Hence, replacing it by $\mathbf{1}_m + \frac{1}{\rho^2}\text{diag}(P^TP)$, the sum does not actually get additional terms. Therefore, $\text{rank}(\mathbf{1}_n\text{diag}(P^TP)^T + \text{diag}(X^TX)\mathbf{1}_m^T - (X^TV_\ell)^{\circ 2}) = \text{rank}((X^TV_\ell)^{\circ 2})$, which explains why the rank is one unit lower in the latter equation, since we cannot use this argument in the former equation in the general case. As a final note, this proof is equivalent to showing that $\mathbf{1}_m$ can be expressed as a linear combination of $\{\mathbf{w}_\alpha(X) \circ \mathbf{w}_\gamma(X)\}_{\alpha,\gamma}$ (as defined in the proof of the previous theorem) when the points $\{\mathbf{x}_i\}$ lie on the relative boundary of an $r$-dimensional hypersphere. □

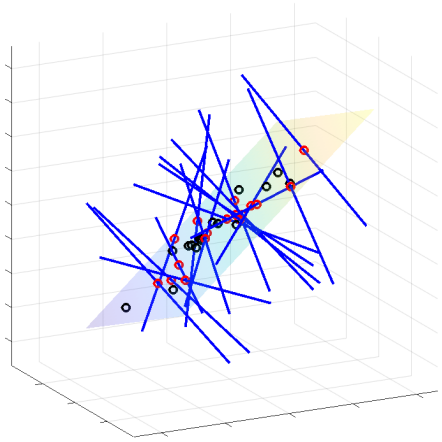
Some examples of the results stated in this corollary are presented in Figure 2.4. We see that these special configurations do not require that both the points and the intercepts lie on the same surface, as in Figure 2.4d. Looking at Figure 2.4c and 2.4d, the points and the intercepts are actually both on the surface of a plane but in the latter case, the additional special configuration, i.e. positioning on the relative boundary of a circle reduces the rank by 1. This can be interpreted as removing a degree of freedom, since $r$-dimensional hyperspheres are in fact objects of dimension $r-1$.



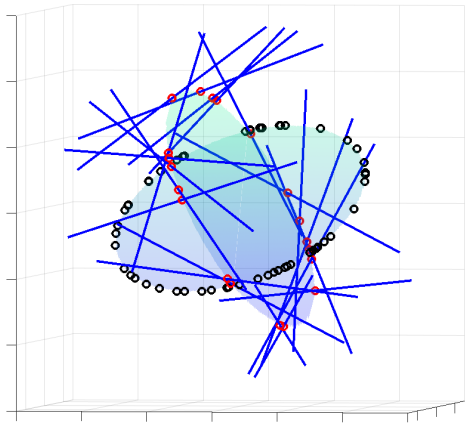
(a) The points are on a line and the intercepts on a plane. This results in an EEDM of rank 3.



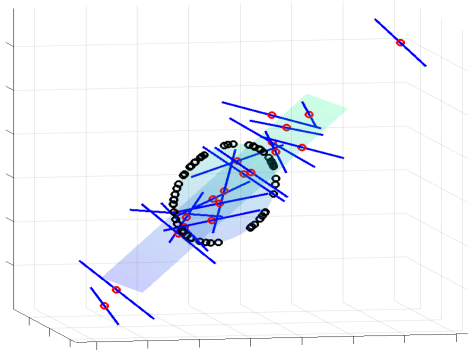
(b) The points are on a plane and the lines on a line. This results in an EEDM of rank 5.



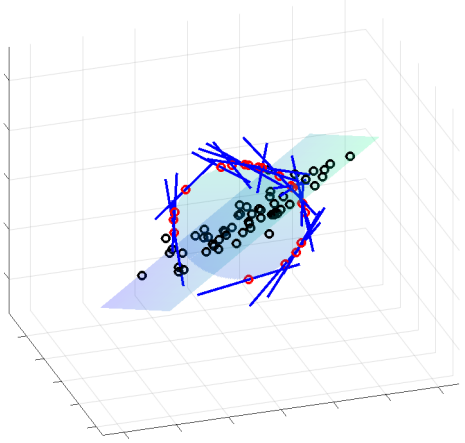
(c) Both points and intercepts are on the same plane. This results in an EEDM of rank 6.



(d) Both points and intercepts lie on the relative boundary of a circle but not the same one. This results in an EEDM of rank 5.



(e) Points lie on the relative boundary of a circle and intercepts on a plane. This results in an EEDM of rank 5.



(f) Intercepts lie on the relative boundary of a circle and points on a plane. This results in an EEDM of rank 6, therefore, no changes compared to Figure 2.4c

Figure 2.4: Some examples of special configurations that reduce the general upper bound. In these images the ambient space dimension is 3, the points $\{\mathbf{x}_i\}$ are represented in black, and the lines in blue. The red dots correspond to the intercepts $\{\mathbf{p}_j\}$ of the lines. The surface in which the points lie has been colored for better visualization.

# Chapter 3

# Inverse problems

After defining EEDMs and exploring their theoretical properties, this chapter discusses methods and algorithms to retrieve the configurations of LVs that generated the distances we measured. Referring to the definitions is Section 1.2, we will denote the true EEDM as $D$, i.e. the matrix containing the exact distances generated by the configuration of LVs. In practice, it is common to measure noisy values, therefore we express by $\widetilde{D} = D + Z$ the noisy measurement matrix, where we assume $z_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$. Finally, we define the mask $M$, a binary matrix modelling missing measurements. In the general case, we observe $M \circ \widetilde{D}$. We present in the following parts two different approaches; first we look at general methods for reconstructing the configurations in Section 3.1. This means that the algorithms we present are not specific to EEDMs. Especially, we will look into **line distance matrices (LDM)**. Then, in Section 3.2, we restrain ourselves to EEDMs where one of the configuration is a set of points (distances between a set of points and a set of LVs). For easier referencing to this case, we call them in this chapter **point to linear variety distance matrices (PLDM)**. As we saw in Section 2.3, PLDMs have an upper bound on their rank, therefore we will look at low-rank approximations of the measured matrix to denoise them. Then, we present a method inspired by the MDU problem in the case of EDMs to reconstruct the original configurations. In both approaches, we convert the problem into an optimization problem with the aim to find the objects most likely to have generated our observations. Below, we define LDMs and PLDMs, as many examples in this chapter will involve them, and present the notations we will use throughout this chapter.

**Definition 3.1** (LDM)**.** *Consider $n$ non-parallel, non-intersecting lines $\{\mathcal{F}_i\}_{1\leq i\leq n}$ in $d$ dimensions given by their parametric equation $\mathbf{p}_i + \mathbf{v}_i t$ and the matrices $P, V \in \mathbb{R}^{d\times n}$ such that $P = [\mathbf{p}_1, \dots, \mathbf{p}_n]$ and $V = [\mathbf{v}_1, \dots, \mathbf{v}_n]$. We call the matrix containing the minimum distance between each and every line in its entries LDM (as depicted in Figure 2.1b) and define:*

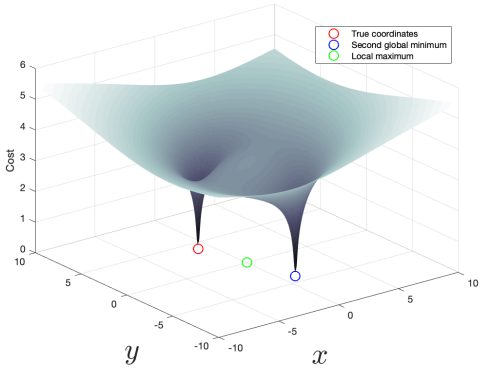
$$ldm(P, V) \triangleq eedm\Big(\{\mathcal{F}_i\}\Big) = edm(P) - T \circ (V^T P) - T^T \circ (P^T V) \in \mathbb{R}^{n\times n}, \tag{3.1}$$

*where the last expression is the one we derived in Equation 2.7.*

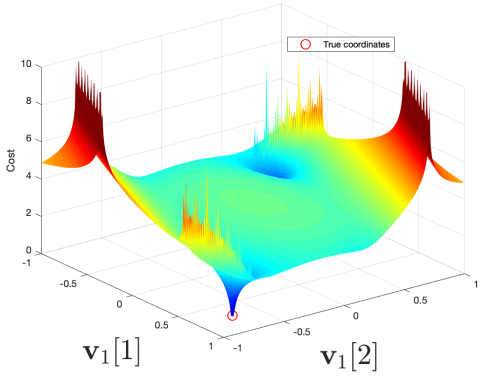
**Definition 3.2** (PLDM)**.** *Consider $n$ points $\{\mathbf{x}_i\}_{1\leq i\leq n}$ and $m$ LVs $\{\mathcal{F}_j\}_{1\leq j\leq m}$ of dimension $k$ embedded in dimension $d$. The parametric equation of the $j$-th LV is $\mathbf{p}_j + \sum_{\ell=1}^{k} \mathbf{v}_j^\ell t_\ell$. We define $X = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, the matrix containing the points, $P = [\mathbf{p}_1, \dots, \mathbf{p}_m] \in \mathbb{R}^{d\times m}$ the matrix containing the intercepts, and $V_\ell = [\mathbf{v}_1^\ell, \dots, \mathbf{v}_m^\ell] \in \mathbb{R}^{d\times m}, \forall\, 1 \leq \ell \leq k$, the matrices containing the $\ell$-th direction vector of each LV. Then, we call the distance matrix containing the distances between the points and the LVs PLDM (as depicted in Figure 2.1a) and define:*

$$pldm(X, P, \{V_\ell\}_{1\leq\ell\leq k}) \triangleq eedm\Big(\{\mathbf{x}_i\}, \{\mathcal{F}_j\}\Big) = edm(X, P) - \sum_{\ell=1}^{k} (X^T V_\ell)^{\circ 2} \in \mathbb{R}^{n\times m}, \tag{3.2}$$

*where the last expression is the one we derived in Equation 2.10.*




(a) Similarly to the example by Parhizkar [21], we consider an EDM of $n = 3$ points in $d = 1$ dimension. The plot represents $f(x, y) = ||\text{edm}(X) - D||_F^2$ where $X = [0, x, y]$, and the true configuration is $[0, -1, 4]$. A global minimum is observed for the original configuration but also for $x = 1, y = -4$. We also note a local maximum for $x = y = 0$.




(b) For this example, we consider $n = 4$ lines in $d = 3$ dimensions, each line $i$ having intercept $\mathbf{p}_i$ and direction $\mathbf{v}_i$. The plot shows the s-stress as a function of the first and second component of the direction vector of the first line, $\mathbf{v}_1[1]$ and $\mathbf{v}_1[2]$, the ground truth being $\mathbf{v}_1 = [1/\sqrt{2}, -1/\sqrt{2}, 0]^T$.

Figure 3.1: Examples of the stress function surface plots for two different cases. The value of the objective function is in logarithms in both cases for better visualization of the variations.

## 3.1 General Methods

In this section, we will look first at a gradient descent based approach and then to interior point methods. These methods do not exploit any potential hidden structure of EEDMs as they can be used for many optimization problems. Based on previous studies ([22], [27], [2]), we choose the objective $f$ to minimize to be:

$$f\Big(\{\mathcal{F}_i\}, \{\mathcal{G}_j\}\Big) = \left|\left| M \circ \left( \text{eedm}\Big(\{\mathcal{F}_i\}, \{\mathcal{G}_j\}\Big) - \widetilde{D} \right) \right|\right|_F^2, \tag{3.3}$$

where the minimization is over the two collection of LVs $\{\mathcal{F}_i\}_{1 \le i \le n}$ and $\{\mathcal{G}_j\}_{1 \le j \le m}$, and $||\cdot||_F$ denotes the Frobenius norm.

For EDMs, this function, called the s-stress function, has been studied by Takane et al. [22] and Parhizkar [21]. As we stated in Lemma 1.5, when the number of measurements is one less than the dimension of the ambient space ($d = n - 1$), the manifold containing these EDMs is given by relint $\mathbb{EDM}^n$ which is a convex set. For this particular setting, Gaffke and Mathar [45] propose an algorithm to find the global minimum of the s-stress function. Otherwise, as shown in the analysis by Parhizkar [21], the objective contains local maxima, saddle points, and global minima. An example, based on the one provided by Parhizkar [21], illustrates the surface plot of the s-stress function in Figure 3.1a.

The situation is more complicated for higher dimensional objects. Consider a set of lines in 3D. Since the minimization is over all possible lines, the problem can be translated to finding the best intercepts and direction vectors for each line. Figure 3.1b shows the surface plot of the s-stress, where the two variables are the first and second component of the direction vector for line 1. As it can be seen, the function is highly non-convex and experiments show that this undesired

property is in particular present over direction vectors. This gives us an idea about the challenge of reconstructing the LVs.

### 3.1.1 Gradient Descent

A natural first approach to tackle our problem is by applying gradient descent on our objective $f$. To be able to give concrete examples, we restrain ourselves to EDMs and LDMs. Even though theoretically more satisfying algorithms already exist for EDMs, we still study them for comparison purposes. We will see that adding the direction vectors into the equation will make the minimization harder, taking the example of LDMs. As seen in Figure 3.1b, $f$ becomes highly non-convex when the objects are lines, therefore we should not expect that gradient descent would give impressive results because of the presence of many local minima, but we rather analyze how good we can approximate the configurations, starting from a close estimation of the actual points/lines, and decided to add this analysis to report our results using this method.

A full overview of the algorithm can be found in the last Part of the book by Boyd and Vandenberghe [46]. Here, we give a short reminder of the gradient descent algorithm for a differentiable function $f : \mathbf{dom}\ f \to \mathbb{R}$ taking the argument $x \in \mathbf{dom}\ f$. The intuitive idea is to move in the opposite direction of the gradient of $f$ on the surface of its graph, since the gradient points to the direction of maximum ascent, the aim being finding the pair $(x^*, f(x^*))$, the optimal point/value. After choosing a starting point $x^{(0)}$, the iterates are given by:

$$x^{k+1} = x^k - \gamma_k \nabla f(x^k).$$

Here, $\gamma_k$ is the step-size, and there exist various methods for choosing this value, such as the backtracking line-search algorithm (Boyd and Vandenberghe [46]).

In our analysis, the objective $f$ is a function taking a matrix $A$ as input. An option for gradient descent in this case is to take the gradient, which would be a matrix, to contain the partial derivative $\frac{\partial f}{\partial A_{ij}}$ in its entry $(i, j)$. We assume that the objective is locally differentiable to allow us to compute the gradient. For a theoretical background on optimization over matrix manifolds, the text by Absil et al. [47] gives a detailed analysis of the subject. Then, in the EDM case, the gradient of the function $\widetilde{f}(X) = ||\mathrm{edm}(X) - \widetilde{D}||_F^2$ is

$$\nabla_X \widetilde{f}(X) = 4\big[X \circ (\mathbf{1}_d \mathbf{1}_n^T (2D - \widetilde{D} - \widetilde{D}^T)) - X(2D - \widetilde{D} - \widetilde{D}^T)\big]. \tag{3.4}$$

For LDMs, we write $\widetilde{f}(P, V) = ||\mathrm{ldm}(P, V) - \widetilde{D}||^2$, and the gradient with respect to $P$ and $V$ respectively of $f$ is

$$\begin{aligned} \nabla_P f(P, V) = 4\big[&P \circ (\mathbf{1}_d \mathbf{1}_n^T (2D - \widetilde{D} - \widetilde{D}^T)) - P(2D - \widetilde{D} - \widetilde{D}^T) \\ &- V(T \circ (2D - \widetilde{D} - \widetilde{D}^T))\big] \end{aligned} \tag{3.5}$$

$$\nabla_V f(P, V) = -4\big[P(T^T \circ (2D - \widetilde{D} - \widetilde{D}^T)) + V(T^T \circ T \circ (2D - \widetilde{D} - \widetilde{D}^T))\big]. \tag{3.6}$$

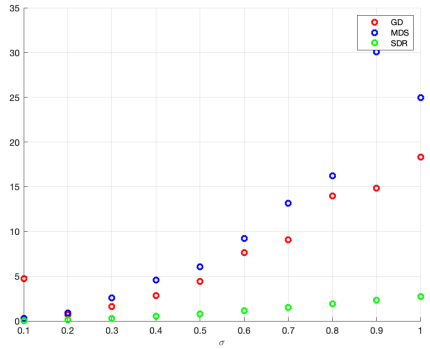
The derivation of these results are given in Appendix A. The iterates are therefore:

$$X^{(k+1)} = X^{(k)} - \gamma_k \nabla_X \widetilde{f}(X^{(k)}),$$

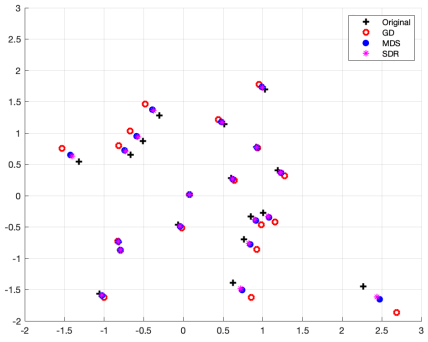
for EDMs, and for the LDM case, we can update each matrix separately:

$$\begin{aligned} P^{(k+1)} &= P^{(k)} - \gamma_k \nabla_P f(V^{(k+1)}, P^{(k)}) \\ V^{(k+1)} &= V^{(k)} - \gamma_k \nabla_V f(V^{(k)}, P^{(k)}). \end{aligned}$$

Figure 3.2 depicts comparisons of the gradient descent with other methods, namely classical MDS and SDR, for estimating original points from noisy EDM measurements. Overall, we note that the




(a) In this example we fix $n = 20$, and run various algorithms to estimate the original points from noisy observations $D + Z$ such that $z_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$. We compare the loss function over different values for the standard deviation $\sigma$ of the additive noise. Each point is averaged over 20 tests, where the original points are generated randomly each time.




(b) Example of reconstructed points after applying the Orthogonal Procrustes algorithm using different methods. The observed matrix $D + Z$ is a noisy version of the EDM $D$, such that $z_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, 1)$.

Figure 3.2: Example of comparison results between gradient descent, classical MDS and SDR for EDMs. The initial point $X^{(0)}$ for gradient descent was chosen randomly for each result.

gradient descent method is slightly better than classical MDS but not as good as SDR. In general, the initial point $X^{(0)}$ to start the gradient descent algorithm can be taken randomly, avoiding us to need an initial knowledge about the configuration of points that generated the observed EDM.

However, for the LDM case, the results were not satisfying using gradient descent. Experimentally, we observe that the main difficulty is to estimate correctly the direction vectors, which make the objective function highly non-convex as discussed previously. The algorithm gets stuck in local minima unless we have an unrealistically good initial guess of the direction vectors $V$.

This method is therefore not a good option for inverse problems involving EEDMs, in addition to a closed form equation of the EEDM itself, it also requires the computation of its gradient, or an approximation of it.

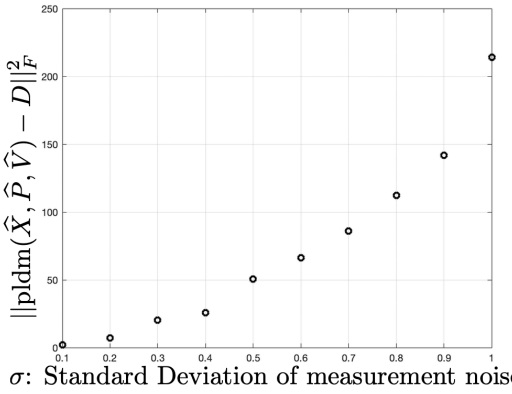
### 3.1.2 Interior Point Method

To deal with the non-convexity of the objective, we turn our attention to global optimization tools. For a PLDM between a set of $n$ points and a set of $m$ $k$-dimensional LVs, we can write our optimization problem as follows:

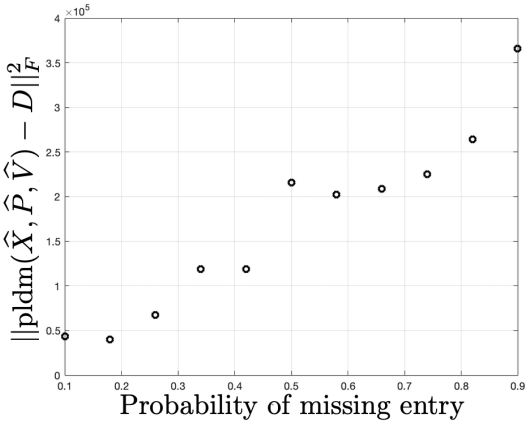
$$
\begin{aligned}
\underset{\substack{X\in\mathbb{R}^{d\times n}\\ P\in\mathbb{R}^{d\times m}, V_\ell\in\mathbb{R}^{d\times m}}}{\text{minimize}} \quad & ||M \circ (\text{pldm}(X, P, \{V_\ell\}) - \widetilde{D})||_F^2 \\
\text{subject to} \quad & ||\mathbf{v}_j^\ell|| = 1,\ \forall 1 \le j \le m,\ \forall 1 \le \ell \le k \\
& \text{diag}(P^T V_\ell) = \mathbf{0},\ \forall 1 \le \ell \le k \\
& \text{diag}(V_{\ell_1}^T V_{\ell_2}) = \mathbf{0},\ \forall \ell_1 \ne \ell_2 \\
& -1 \le \mathbf{v}_j^\ell \le 1,\ \forall 1 \le j \le m,\ \forall 1 \le \ell \le k,
\end{aligned}
\tag{$Q_f$}
$$

where the inequalities in the last constraint are component-wise. The formulation is similar for LDMs. The option we adopted to solve this problem uses interior point methods for nonlinear

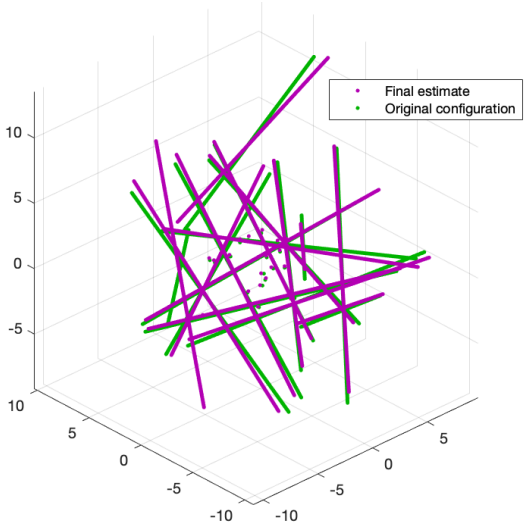
programming, presented by Ugray et al. [48], and using the Matlab implementation. This algorithm requires an initial estimate for the optimal values; or starting points. To have short computation times, these estimations should be relatively close to the original values, especially for the direction vectors. Figure 3.6 gives examples about the performance of the algorithm. The first two figures compare the estimated PLDM, denoted by $\text{pldm}(\hat{X}, \hat{P}, \hat{V})$ to the true one, $D$, for noisy and missing measurements, by measuring the squared Frobenius norm of their difference. In these examples, we chose $X^{(0)} = X + \mathcal{N}(0,1)$, $P^{(0)} = P + \mathcal{N}(0,1)$, $V^{(0)} = V + \mathcal{N}(0, 0.01)$, where $X, P$ and $V$ are the ground truth values, and the addition represents a random starting point around the true values. Each point on the plots are obtained by averaging 10 independent tests, each test creating the data randomly.



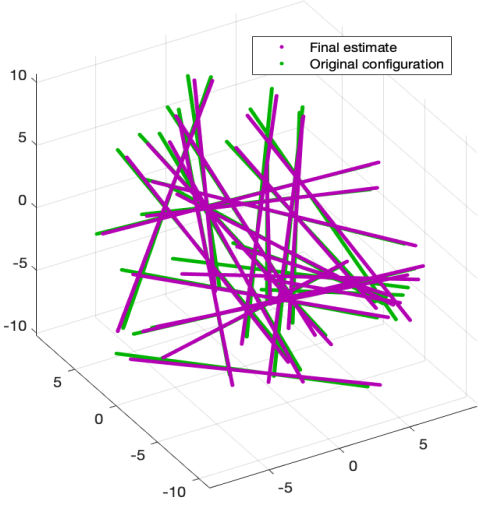

(a) This plot compares results of the objective for growing additive noise variance over the observed measurements. The probability of an observed distance to be missing was fixed to 0.1.




(b) This plot compares results of the objective for growing probability of missing entries in the observed distance matrix. The noise on the measurements are additive Gaussian with a fixed variance of 0.01




(c) Example of reconstructed points and lines compared to the original configuration, in the case of PLDMs. The entries of the measured distance matrix have independent additive noise following $\mathcal{N}(0, 0.25)$.




(d) Example of reconstructed lines compared to the original configuration, in the case of LDMs. The entries of the measured distance matrix have independent additive noise following $\mathcal{N}(0, 0.25)$.

Figure 3.3: Example of performance results by solving the optimization problem $(Q_f)$.

Over various tests, we observed that using the starting points described above, the algorithm is

robust to noisy observations of the EEDMs. However, for missing measurements, we start to get bad estimations around the probability 0.2 of each single measurement being missing.

## 3.2 Specific Algorithms for PLDMs

Starting from our analysis in Section 2.3 and inspired by the algorithms in Section 1.2, we present here approaches specific to the context of PLDMs, i.e. the distance matrix contains the minimum distance between a point and an LV in each of its entries. We suppose that the rank $r$ of the PLDM is known in practice or can be estimated well, which is not unrealistic since we usually know the dimension of the ambient space and can get an upper bound on the rank as shown in Theorem 2.5 or if we have a prior knowledge of particular structures in the configurations as in Corollary 2.6. The methods discussed here need only the measured matrix and the estimation of the rank of the original distance matrix, and, as opposed to previous algorithms, they do not require an initial estimation of the configurations which was a drawback since we often needed a good approximation of the true values, especially for direction vectors.

### 3.2.1 Denoising and Completion

First, consider we observe a noisy PLDM $\widetilde{D} \in \mathbb{R}^{n\times m}$ as defined above. Before reconstructing the configurations, we aim to estimate the original PLDM $D \in \mathbb{R}^{n\times m}$, and call the estimation $\widehat{D} \in \mathbb{R}^{n\times m}$. This can be expressed as the well-known low-rank approximation problems, expressed as:

$$\begin{aligned} \underset{\widehat{D}\in\mathbb{R}^{n\times m}}{\text{minimize}} \quad & g(\widehat{D},\widetilde{D}) \\ \text{subject to} \quad & \operatorname{rank}(\widehat{D}) \leq r. \end{aligned} \tag{Q}$$

There exists various options for the objective function $g$, but to be coherent with the s-stress function, we choose $g(\widehat{D},\widetilde{D}) \triangleq ||\widehat{D}-\widetilde{D}||_F^2$. The low-rank approximation problem with the Frobenius norm as a cost function has in fact been studied by Eckart and Young [49], where it is shown that the solution is a truncated sum of the singular value decomposition of the observed matrix, as detailed in the theorem below.

**Theorem 3.1** (Eckart and Young [49])**.** *Let $\widetilde{D}$ be a rank $r'$ matrix and its singular value decomposition given by $\sum_{i=1}^{r'} \sigma_i \mathbf{u}_i \mathbf{w}_i^T$, with the singular values $\sigma_i$ of $\widetilde{D}$ arranged in decreasing order. Then, for $r < r'$, the truncated sum*

$$\widehat{D} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{w}_i^T$$

*is a minimizer of $||\widetilde{D}-A||_F^2$ over the matrices $A$ with rank less than or equal to $r$, and the minimum is $||\widetilde{D}-\widehat{D}||_F^2 = \sum_{i=r+1}^{r'} \sigma_i^2$.*

As a side note, we could have also taken $g(\widehat{D},\widetilde{D}) = ||\widehat{D}-\widetilde{D}||_2^2$, where $||A||_2 = \sigma_{max}(A)$ is the spectral norm of a matrix, and this would have resulted in the same low-rank matrix $\widehat{D}$, as proven by Mirsky [50], the difference being that in the spectral norm case, the minimum is attained at $||\widetilde{D}-\widehat{D}||_2^2 = \sigma_{r+1}^2$.

Continuing from Theorem 3.1, writing $\widetilde{D} = U\Sigma W^T$ by SVD and denoting by $\widehat{\Sigma}$ the matrix containing the $r$ largest singular values of $\widetilde{D}$ in its diagonal, we estimate $D$ by $\widehat{D} = \widehat{U}\widehat{\Sigma}\widehat{W}^T$, where $\widehat{U}$ and $\widehat{W}$ contain the left and right singular vectors corresponding to the $r$ largest singular values respectively.

However, if we additionally have missing measurements, modeled by $M \circ \widetilde{D}$, doing only this computation is not sufficient anymore, since the low-rank approximation matrix can contain negative values at the positions where there were missing entries. Adapting Algorithm 3, we propose Algorithm 4 presented below, where we alternatingly force the rank property and zero the negative entries. Line 6 in the algorithm, where we force observed entries, is optional, in the sense that it will be helpful if our measurements are noiseless or the noise is low, but in practice it is usually helpful since the noisy observations are close to the true value, closer than what the SVD could potentially give. We note that compared to Algorithm 3 we replace the EVD by SVD since we do not have a guarantee of the matrix being symmetric or even square.

**Algorithm 4** Alternating Rank-Based PLDM Completion

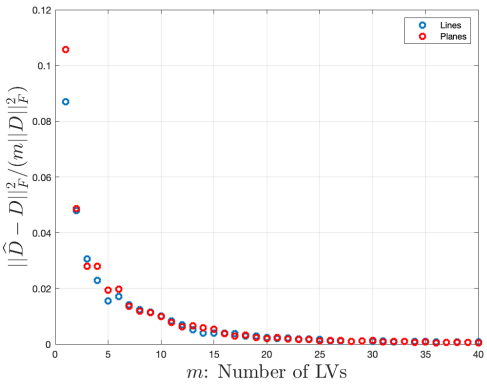
```
1: function RankCompletePLDM(D̃, M)
2:     D_M ← D̃_M
3:     D_{11^T − M} ← μ        ▷ Initialize unobserved entries
4:     repeat
5:         D ← SVThreshold(D, r)
6:         D_M ← D̃_M           ▷ Force known entries
7:         D ← (D)_+           ▷ Zero the negative entries
8:     until MaxIter or Convergence
9:     return D
10: end function
11: function SVThreshold(D, r)
12:     UΣW^T ← D               ▷ SVD
13:     Σ ← diag(σ_1 ... σ_r, 0 ... 0)
14:     return UΣW^T
15: end function
```

Similarly to Algorithm 3, we do not have a guarantee of convergence, but experimentally, the results are satisfying and the missing entries are completed well, especially for large numbers of measurements $n$ and $m$. Moreover, Figure 3.4 shows performance results in $d = 3$ dimensions of Algorithm 4, namely the value of $||\widehat{D} - D||_F^2/(m||D||_F^2)$, where $\widehat{D}$ is the estimated matrix and $D$ the true PLDM, when observing $M \circ (D + Z)$. Each point in the figures is obtained by averaging 50 independent tests, and for each test, the points and LVs are generated randomly. Moreover, the number of iterations of Algorithm 4 is fixed to 100. We present PLDM examples where the LVs are either lines or planes.

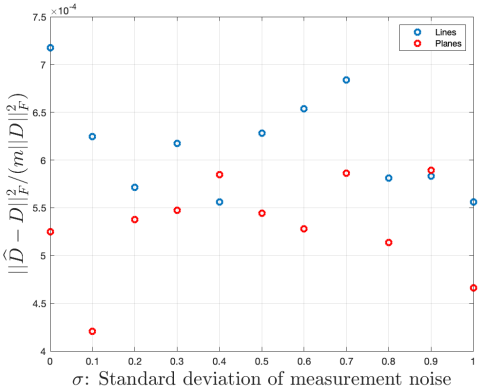
We observe that there is not much difference between what we obtain when the PLDM is generated by points/lines or points/planes. The algorithm is more effective when the number of measurements is large, as stated previously. This can be explained by the fact that each measured distance implicitly adds a geometrical constraint, hence, the larger the number of distances we get, the fewer the number of degrees of freedom there is for guessing the rest. The noise does not affect much the approximation error, since in realistic cases, it is very small compared to the values of the entries of the distance matrices. As expected, the error grows as the number of missing measures grow. However, this small value for the error, for example when the probability of each single measurement being missing is around 0.9 should not be interpreted as a good estimation of $D$. Experimentally, the estimations stop becoming accurate around a probability of missing measurements of 0.3.

### 3.2.2 Reconstruction

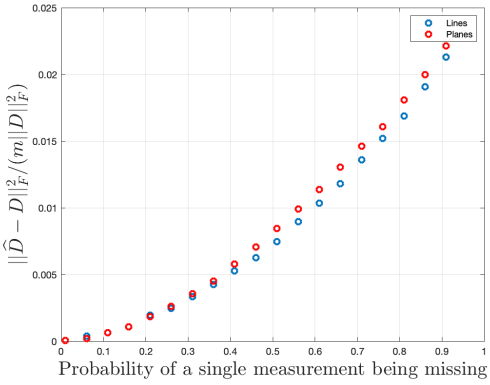
We now turn our attention to reconstructing the configurations of points and LVs that generated the observed PLDM, i.e. retrieving $X, P, \{V_\ell\}_{1 \leq \ell \leq k}$ as defined in Equation 3.2, and for this we will translate the problem as a special case of MDU. We remind that MDU is the setting where we




(a) Plot of the approximation error $||\widehat{D} - D||_F^2/(m||D||_F^2)$ for growing number of LVs $m$. The number of points $n$ is fixed to 20. The additive noise $Z$ is fixed, such that $z_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, 0.01)$ and the probability of a measurement being missing is fixed to 0.1.




(b) Plot of the approximation error $||\widehat{D} - D||_F^2/(m||D||_F^2)$ for growing additive noise variance in the observed PLDM. The number of points and LVs are fixed to $n = 20$ and $m = 40$. The probability of a measurement being missing is fixed to 0.1.




(c) Plot of the approximation error $||\widehat{D} - D||_F^2/(m||D||_F^2)$ for growing probability of measurements being missing. The number of points and LVs are fixed to $n = 20$ and $m = 40$. The additive noise $Z$ is fixed, such that $z_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, 0.01)$.

Figure 3.4: Performance evaluation of Algorithm 4.

partition a set of points into two subsets and we measure the distances of the points belonging to different subsets and not the same subset. In our context, each LV $j$ contains $n$ points which are the closest points on the LV to one of the points in the set $\{\mathbf{x}_i\}_{1\leq i\leq n}$. Let us denote by $\widetilde{\mathbf{p}}_{ij} \in \mathbb{R}^d$ the point on LV $j$ closest to the point $\mathbf{x}_i$. These can be interpreted as "ghost" points $\{\widetilde{\mathbf{p}}_{ij}\}_{1\leq j\leq m}$ appearing in the configuration only when dealing with point $\mathbf{x}_i$. As proposed by Dokmanic and Vetterli [18], MDU can be expressed as EDM completion in the following way and adapted to our case. Consider the collection of points $\mathbf{x}_i \cup \{\widetilde{\mathbf{p}}_{ij}\}_{1\leq j\leq m}$ and the matrix $\widetilde{P}_i = [\widetilde{\mathbf{p}}_{i1}, \ldots, \widetilde{\mathbf{p}}_{im}] \in \mathbb{R}^{d\times m}$. Then, the EDM $D_i$ of this collection can be written as:

$$D_i \triangleq \operatorname{edm}\Big(\mathbf{x}_i \cup \{\widetilde{\mathbf{p}}_{ij}\}_j\Big) = \begin{pmatrix} 0 & \operatorname{edm}(\mathbf{x}_i, \widetilde{P}_i) \\ \operatorname{edm}(\mathbf{x}_i, \widetilde{P}_i)^T & \operatorname{edm}(\widetilde{P}_i) \end{pmatrix} \in \mathbb{R}^{(m+1)\times(m+1)}. \tag{3.7}$$

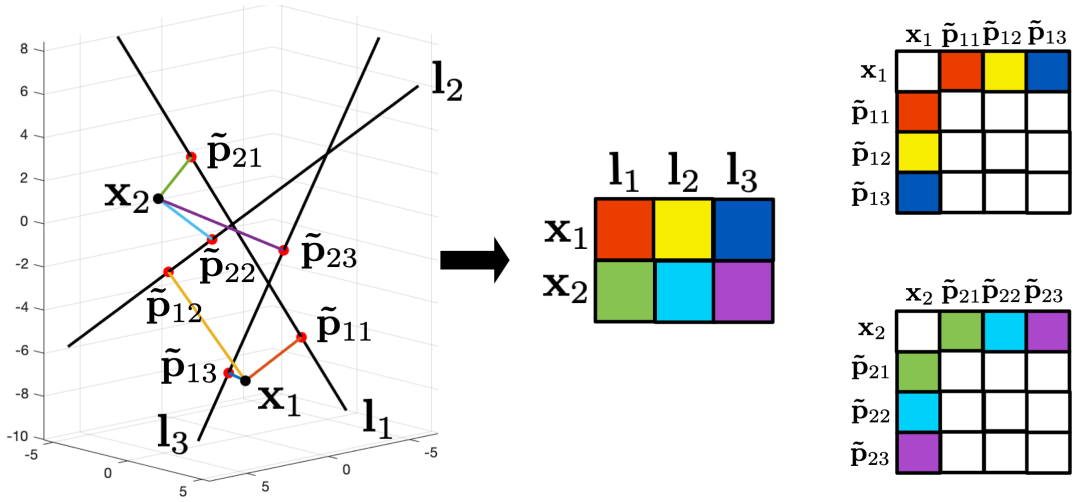



Figure 3.5: In this example, we consider the distance between the two points $\mathbf{x}_i$ (black dots) and and the three lines $\mathbf{l}_j$ (black lines). The red points correspond to the points $\widetilde{\mathbf{p}}_{ij}$ on line $j$ closest to the point $\mathbf{x}_i$. The resulting PLDM can be decoupled into two incomplete EDM, one for each point $i$, as shown on the right.

However, we do not know the distances between these "ghost" points, i.e. $\mathrm{edm}(\widetilde{P}_i)$, but this can be represented as observing $M_{MDU} \circ D_i$, where

$$M_{MDU} = \begin{pmatrix} 0 & \mathbf{1}_m^T \\ \mathbf{1}_m & 0_{m\times m} \end{pmatrix}, \tag{3.8}$$

therefore we note that this problem becomes similar to EDM completion for each point $\mathbf{x}_i$ separately, which we choose to address by solving the semidefinite relaxation presented in Problem (P) in Section 1.2. A visual representation is provided in Figure 3.5. The reconstruction procedure is described in Algorithm 5.

**Algorithm 5** PLDM Reconstruction

**function** ReconstructPLDM($D, d$)
  **for** $i \in \{1, \dots, n\}$ **do**
    $D_i, \widehat{XP}_i \leftarrow \mathrm{SDR}(D_i, M_{MDU}, d)$
  **end for**
  $\{\widehat{\mathbf{x}}_i\}, \{\widehat{\mathbf{p}}_{ij}\} \leftarrow$ MatchXP($\widehat{XP}_1, \dots, \widehat{XP}_n$)
  **for** $j \in \{1, \dots, m\}$ **do**
    $\widehat{\mathbf{p}}_j, \{\widehat{\mathbf{v}}_j^\ell\}_\ell \leftarrow$ FitLV($\{\widehat{\mathbf{p}}_{ij}\}_i, \{\widehat{\mathbf{x}}_i\}$)
  **end for**
  **return** $\{\widehat{\mathbf{x}}_i\}, \{\widehat{\mathbf{p}}_j\}, \{\widehat{\mathbf{v}}_j^\ell\}$
**end function**

**function** SDR($D_i, M_{MDU}, d$)
  $x \leftarrow -1/(m+1+\sqrt{m+1})$
  $y \leftarrow -1/\sqrt{m+1}$
  $Q \leftarrow [y\mathbf{1}_m, x\mathbf{1}_m\mathbf{1}_m^T + yI]^T$
  $H \leftarrow$ SolveP($D_i, Q, M_{MDU}, \lambda$)
  $D_i \leftarrow \mathrm{edm}(QHQ^T)$
  $W\Lambda W^T \leftarrow D_i$ ▷ EVD
  $\widehat{XP} \leftarrow \mathrm{diag}(\sqrt{\lambda_1}, \dots, \sqrt{\lambda_d})[\mathbf{w}_1, \dots, \mathbf{w}_d]^T$
  **return** $D_i, \widehat{XP}$
**end function**

Overall, the strategy is to reconstruct each "ghost" point, along with the points $\mathbf{x}_i$, and then fit an LV for each set $\{\widetilde{\mathbf{p}}_{ij}\}_i$, since for a fixed $j$, each $\widetilde{\mathbf{p}}_{ij}$ is on the surface of the same LV $j$. We note that the number $n$ of points $\{\mathbf{x}_i\}$ should be greater or equal to $k$, the dimension of the linear varieties, to be able to determine each linear variety $j$ with $\{\widetilde{\mathbf{p}}_{ij}\}_i$. We now detail and discuss some practical subtleties of Algorithm 5.

**SDR**: The function SDR is used for completing the EDM $D_i$ and estimating the set of points that generated it. It is largely based on the one presented by Dokmanic et al. [27], the only difference

being that we used an eigenvalue decomposition instead of the singular value decomposition, since the $D_i$ is symmetric. We only keep the $d$ largest eigenvalue-eigenvector pairs to force the resulting points to be $d$-dimensional. Of course, since the points are $d$-dimensional in reality, the remaining $m+1-d$ smallest eigenvalues will be very small compared to the first $d$ of them, this can also be verified in practice, where quick simulations show that these smallest eigenvalues are nearly 0, but we choose to truncate them.

**SolveP**: The function SolveP is the one solving the semidefinite program defined in Problem (P). There exists various programming libraries to solve (P), in our case, we chose to use the Matlab/CVX implementation (Grant et al. [51], Grant and Boyd [52]). The regularization term $\lambda$ proposed by Biswas et al. [26] is chosen heuristically, generally depending on the number of measurements we have.

**Note:** Instead of the function SDR, which solves the optimization Problem (P), we could have taken other EDM completion algorithms such as Rank Alternation (Algorithm 3), Alternating Descent or OptSpace, as presented in Section 1.2. However, OptSpace may not be the better option in this case, because it assumes the missing entries are scattered randomly, and does not perform well when they are structured in blocks, as in $M_{MDU}$ ([27]).

Therefore, at this point of Algorithm 5, for a fixed $i$, we have a way of retrieving the points that generated $D_i$. We call the matrix containing the estimates of the points in its columns $\widehat{XP}_i = [\widehat{\mathbf{x}}_i, \widehat{\mathbf{p}}_{i1}, \dots, \widehat{\mathbf{p}}_{im}] \in \mathbb{R}^{d\times(m+1)}$. But these estimates we have for $\mathbf{x}_i$ and $\{\widetilde{\mathbf{p}}_{ij}\}_j$ are entangled over the different values of $i$, in the sense that we do not have a guarantee for them to be aligned correctly relative to each other, which is a direct consequence of the invariance to rigid motion of EDMs.

**MatchXP**: This function addresses the latter problem and its goal is to align correctly the estimates ascribed in the columns of $\widehat{XP}_1, \dots, \widehat{XP}_n$. In theory, we can use Orthogonal Procrustes to match the columns of $\widehat{XP}_i$ to $\mathbf{x}_i \cup \{\widetilde{\mathbf{p}}_{ij}\}_j$, and doing this for each point $i$, we would solve the problem, but this is possible only when we have prior knowledge about the points, for example some anchors. This is however the version presented in Algorithm 5. On the other hand, if we additionally have access to the distances between $\mathbf{x}_i$'s, i.e. $\mathrm{edm}(X)$, we can construct an alternative matrix $D_i$ to be completed:

$$D_i = \mathrm{edm}\Big(\{\mathbf{x}_i\} \cup \{\widetilde{\mathbf{p}}_{ij}\}_j\Big) = \begin{pmatrix} \mathrm{edm}(X) & \mathrm{edm}(X, \widetilde{P}_i) \\ \mathrm{edm}(X, \widetilde{P}_i)^T & \mathrm{edm}(\widetilde{P}_i) \end{pmatrix} \in \mathbb{R}^{(n+m)\times(n+m)}, \tag{3.9}$$

with the corresponding mask:

$$M_{MDU} = \begin{pmatrix} \mathbf{1}_n\mathbf{1}_n^T & [\mathbf{1}_m, 0_{m\times(n-1)}]^T \\ [\mathbf{1}_m, 0_{m\times(n-1)}] & 0_{m\times m} \end{pmatrix}. \tag{3.10}$$

In this case, the matrix $\widehat{XP}_i$ would additionally contain an estimate for every point $\mathbf{x}_i$: $\widehat{XP}_i = [\widehat{\mathbf{x}}_1, \dots, \widehat{\mathbf{x}}_n, \widehat{\mathbf{p}}_{i1}, \dots, \widehat{\mathbf{p}}_{im}]$. Then, the coordinates of the points $\mathbf{x}_i$ can be estimated from $\mathrm{edm}(X)$, using for example the classical MDS algorithm, and taken as anchor points. It follows to match $\{\widehat{\mathbf{x}}_i\}$ to these anchors, to find an alinement for the overall set, i.e. rotation/reflection matrices $R_i$ and translation vectors $\mathbf{b}_i$. This version does not require any changes in the algorithm or our discussion except that each "$m+1$" should be replaced by "$m+n$".

**Remark.** *In this latter situation, we get a new estimation of the full set of points $\{\mathbf{x}_i\}$ at each iteration. Therefore, these estimations could be averaged at the end as the final estimation.*

**FitLV**: At this point, we have all the estimates $\widehat{\mathbf{x}}_i$ and $\widehat{\mathbf{p}}_{ij}$ of the points $\mathbf{x}_i$ and $\widetilde{\mathbf{p}}_{ij}$ respectively, $\forall 1 \le i \le n$ and $\forall 1 \le j \le m$. This last function is used to fit the corresponding LV of dimension $k$ passing through the points $\{\widehat{\mathbf{p}}_{ij}\}_i$ and this for every $j$. Since the estimations may contain small

errors, we will solve the problem statistically, i.e. by minimizing the mean squared error (MSE). For a fixed LV $j$, this can be written as:

$$\begin{aligned} \underset{\substack{N_j \in \mathbb{R}^{(d-k)\times d} \\ \mathbf{c}_j \in \mathbb{R}^d}}{\text{minimize}} \quad & E_i[||N_j(\widehat{\mathbf{p}}_{ij} - \mathbf{c}_j)||^2] = \mathcal{L}(N_j, \mathbf{c}_j) \\ \text{subject to} \quad & N_j N_j^T = I. \end{aligned} \tag{$Q_{\text{fit}}$}$$

**Algorithm 6** FitLV

**function** FitLV($\{\widehat{\mathbf{p}}_{ij}\}_i, \{\widehat{\mathbf{x}}_i\}$)
  $\mathbf{p}_j^{\text{avg}} \leftarrow \frac{1}{n}[\widehat{\mathbf{p}}_{1j}, \dots, \widehat{\mathbf{p}}_{nj}]\mathbf{1}_n$
  $K_j \leftarrow \frac{1}{n}\sum_{i=1}^n (\widehat{\mathbf{p}}_{ij} - \mathbf{p}_j^{\text{avg}})(\widehat{\mathbf{p}}_{ij} - \mathbf{p}_j^{\text{avg}})^T$
  $W \Lambda W^T \leftarrow K_j$ ▷ EVD, such that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$
  $\widehat{\Phi}_j \leftarrow [\mathbf{w}_1, \dots, \mathbf{w}_k]$
  $\widehat{\mathbf{p}}_j \leftarrow \frac{1}{n}[\widehat{\mathbf{p}}_{1j}, \dots, \widehat{\mathbf{p}}_{nj}]\mathbf{1}_n - \frac{1}{n}\widehat{\Phi}_j\widehat{\Phi}_j^T[\widehat{\mathbf{x}}_1, \dots, \widehat{\mathbf{x}}_n]\mathbf{1}_n$
  **return** $\widehat{\mathbf{p}}_j, \widehat{\Phi}_j$
**end function**

This problem is translated as follows; we are trying to find the normal vectors ascribed in the rows of $N_j$ that best fits, in the least squares sense, the points $\{\widehat{\mathbf{p}}_{ij}\}_i$. Since the normal vectors are orthonormal by assumption, we have $N_j N_j^T = I$. Finally, $\mathbf{c}_j$ is a point on the LV (or one fixed to be). This is so that we deal only with the normal vectors for now, since $N_j(\widehat{\mathbf{p}}_{ij} - \mathbf{c}_j)$ describes a linear subspace (i.e. $\mathbf{0}$ belongs to it). We can solve this greedily by first starting with $\mathbf{c}_j$. We have:

$$\mathcal{L}(N_j, \mathbf{c}_j) = E_i[\widehat{\mathbf{p}}_{ij}^T N_j^T N_j \widehat{\mathbf{p}}_{ij}] + \mathbf{c}_j^T N_j^T N_j \mathbf{c}_j - \mathbf{c}_j^T N_j^T N_j E_i[\widehat{\mathbf{p}}_{ij}] - E_i[\widehat{\mathbf{p}}_{ij}]^T N_j^T N_j \mathbf{c}_j.$$

Defining the statistical average of the points to be $\mathbf{p}_j^{\text{avg}} = E_i[\widehat{\mathbf{p}}_{ij}] = \frac{1}{n}[\widehat{\mathbf{p}}_{1j}, \dots, \widehat{\mathbf{p}}_{nj}]\mathbf{1}_n$ and taking the gradient of $\mathcal{L}$ with respect to $\mathbf{c}_j$, we can put the expression to $\mathbf{0}$ to obtain the minimum since $\mathcal{L}$ is convex and quadratic on $\mathbf{c}_j$ ($N_j^T N_j \succeq 0$):

$$\nabla_{\mathbf{c}_j} \mathcal{L} = 2N_j^T N_j \mathbf{c}_j^* - 2N_j^T N_j \mathbf{p}_j^{\text{avg}} = \mathbf{0}$$

which implies that we can take $\mathbf{c}_j^* = \mathbf{p}_j^{\text{avg}}$. Since for a scalar $c$, we have $\text{tr}(c) = c$, we may write:

$$\begin{aligned} \mathcal{L}(N_j) &= \text{tr}\Big(E_i[(\widehat{\mathbf{p}}_{ij} - \mathbf{p}_j^{\text{avg}})^T N_j^T N_j (\widehat{\mathbf{p}}_{ij} - \mathbf{p}_j^{\text{avg}})]\Big) \\ &= \text{tr}\Big(N_j E_i[(\widehat{\mathbf{p}}_{ij} - \mathbf{p}_j^{\text{avg}})(\widehat{\mathbf{p}}_{ij} - \mathbf{p}_j^{\text{avg}})^T] N_j^T\Big), \end{aligned}$$

where we used the linearity and the circular property of the trace operator. Observing that $K_j = E_i[(\widehat{\mathbf{p}}_{ij} - \mathbf{p}_j^{\text{avg}})(\widehat{\mathbf{p}}_{ij} - \mathbf{p}_j^{\text{avg}})^T]$ is the empirical covariance matrix of the set $\{\widehat{\mathbf{p}}_{ij}\}_i$, it is PSD. We can rewrite the problem as:

$$\begin{aligned} \underset{N_j \in \mathbb{R}^{(d-k)\times d}}{\text{minimize}} \quad & \text{tr}\Big(N_j K_j N_j^T\Big) \\ \text{subject to} \quad & N_j N_j^T = I. \end{aligned} \tag{$Q_{\text{fit}}$}$$

To solve this optimization problem, we state the following lemma by Horn and Johnson [53] (Corollary 4.3.39). We omit the proof to avoid digressing too much, and recommend this last text for more details.

**Lemma 3.2** (Horn and Johnson [53])**.** *Let $K \in \mathbb{R}^{d\times d}$ be a symmetric matrix and $N \in \mathbb{R}^{(d-k)\times d}$ such that $NN^T = I$. Then, $tr\Big(NKN^T\Big)$ is minimized by taking $N$ to contain the eigenvectors of $K$ in its rows corresponding to the $d-k$ smallest eigenvalues. Additionally, the minimum is given by the sum of these $d-k$ smallest eigenvalues.*

Therefore, we may take the $d-k$ eigenvectors of $K_j$ corresponding to the smallest $d-k$ eigenvalues as the normal vectors of the LV. Then, the direction vectors of the LV $j$, ascribed in the columns of $\widehat{\Phi}_j = [\widehat{\mathbf{v}}_j^1, \ldots, \widehat{\mathbf{v}}_j^k] \in \mathbb{R}^{d\times k}$ can be taken to be an orthonormal basis for the orthogonal complement of the row space of $N_j$, i.e. $\mathcal{R}(N_j^T)^\perp$. These are in fact given by the $k$ eigenvectors of $K_j$ corresponding to the $k$ largest eigenvalues.

Finally, we are left to find the intercept $\widehat{\mathbf{p}}_j$. In Equation 2.9, we found that the point $\widetilde{\mathbf{p}}_{ij}$ minimizing the distance between the LV $j$ and the point $i$ can be written as $\widetilde{\mathbf{p}}_{ij} = \mathbf{p}_j + \Phi_j\Phi_j^T\mathbf{x}_i$. Having the estimates $\widehat{\mathbf{p}}_{ij}$, $\widehat{\Phi}_j$ and $\widehat{\mathbf{x}}_i$ found previously, we can estimate the intercepts, noted as $\widehat{\mathbf{p}}_j$, using :

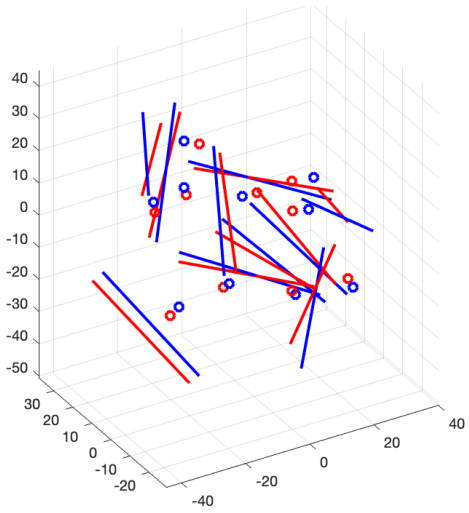
$$\widehat{\mathbf{p}}_j = E_i[\widehat{\mathbf{p}}_{ij} - \widehat{\Phi}_j\widehat{\Phi}_j^T\widehat{\mathbf{x}}_i] = \frac{1}{n}\widehat{P}_j\mathbf{1}_n - \frac{1}{n}\widehat{\Phi}_j\widehat{\Phi}_j^T\widehat{X}\mathbf{1}_n, \tag{3.11}$$

where $\widehat{P}_j = [\widehat{\mathbf{p}}_{1j}, \ldots, \widehat{\mathbf{p}}_{nj}] \in \mathbb{R}^{d\times n}$ and $\widehat{X} = [\widehat{\mathbf{x}}_1, \ldots, \widehat{\mathbf{x}}_n] \in \mathbb{R}^{d\times n}$.

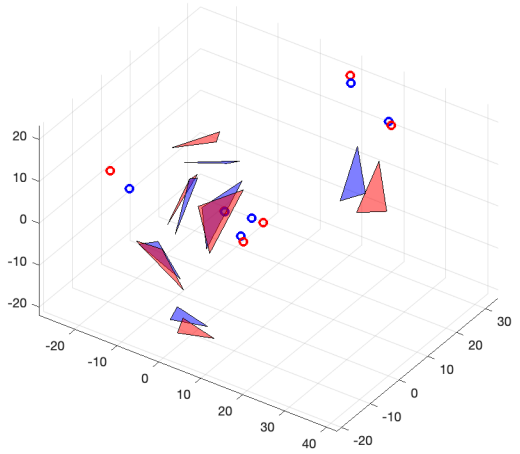
The procedure is summarized in Algorithm 6.

We note that if the estimates $\{\widehat{\mathbf{x}}_i\}$, $\{\widehat{\mathbf{p}}_{ij}\}$ are exact, i.e. the same as the original points, the statistical fit would give the true LV.

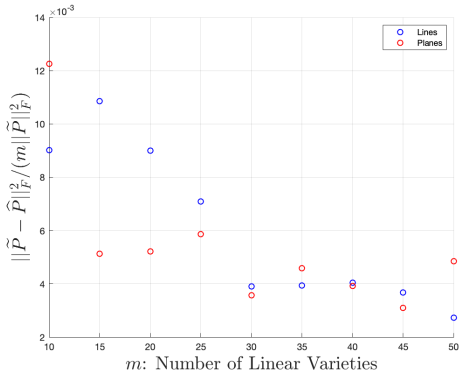
**Experimental Observations:** Some examples of reconstructions are given in Figure 3.6. In practice, we remark that the trickiest part in the reconstruction is correctly estimating the "ghost" points $\{\widetilde{\mathbf{p}}_{ij}\}$, which is directly linked to the EDM completion of $D_i$. The number of missing entries is very large, therefore there may be problems for correctly estimating it. In applications, the results are better when we know the distances between the points $\{\mathbf{x}_i\}$, i.e. edm$(X)$ therefore using the expression in 3.9 instead of 3.7. This is because we already have some parts of the EDM filled which implicitly implies geometric constraints and helps us better estimate the missing entries. Additionally, we also need to know the position of a subset of $\{\mathbf{x}_i\}$ to use them as anchor points to keep the overall structure (otherwise $\{\widetilde{\mathbf{p}}_{ij}\}_i$ may not be on the same linear variety). Moreover, we found that the best algorithm for completion is the SDR and that the regularization term $\lambda$ in the algorithm is important for a correct reconstruction of estimates. Indeed, choosing incorrectly this value may lead to estimates with different dimension than $d$, where the $d$ largest eigenvalues in line 18 of Algorithm 5 are not significantly larger than the rest. We chose $\lambda$ to be the square root of the number of zeros or missing entries of $D_i$, because of empirically good results, as stated in [27], therefore $\lambda = \sqrt{(n+m)^2 - n^2 - 2m + n}$ in the case of Equation 3.9. We also observed that the results are generally better when the number of points $n$ is larger than the number of LVs $m$. Overall, the problem of reconstruction of PLDMs can be reduced to a problem of very sparse EDM completion, where the missing entries are in block structure. This case is found to be harder than completing randomly structured missing entries, because there is a lot of information missing for $\widetilde{\mathbf{p}}_{ij}$'s. Therefore, the method we propose is to first use Algorithm 4 to denoise and complete the observed PLDM. Then reconstruct an estimation of the points and LVs with Algorithm 5. Finally, use these estimations as input, i.e. starting points, for the non linear programming algorithm presented in Subsection 3.1.2, if necessary.



(a) Example with points and lines in 3D. The original configuration is represented in red and the estimation in blue. Originally, this result is obtained with 40 points and 40 lines but we chose to plot less for better visualization.



(b) Example with points and planes in 3D. The original configuration is represented in red and the estimation in blue. Originally, this result is obtained with 40 points and 40 planes but we chose to plot less for better visualization.




(c) Plot of the error of reconstruction $||\widetilde{P} - \widehat{P}||_F^2/(m||\widetilde{P}||_F^2)$ after applying Orthogonal Procrustes as the number of linear varieties $m$ increases. $\widetilde{P}$ corresponds to the matrix of true points $\widetilde{\mathbf{p}}$ and $\widehat{P}$, to the estimations $\widehat{\mathbf{p}}$. Each point is obtained by averaging 10 independent tests.

Figure 3.6: Results of reconstruction obtained with the algorithm PLDM Reconstruction.

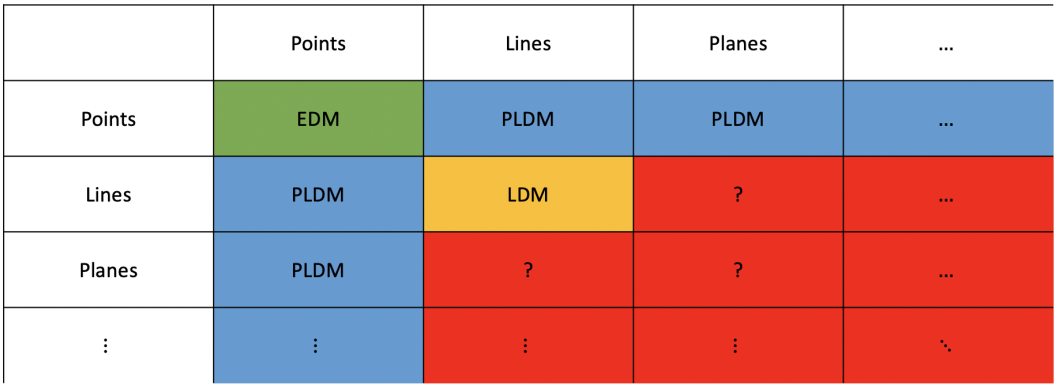



Figure 3.7: EEDM table revisited.

We end this chapter by revisiting the table we started with in Figure 1.2. We indicated in green in Figure 3.7, EDMs to represent the fact that they have been studied by various authors and many algorithms exist for problems involving them. The blue entries represent PLDMs for which we found similar properties, such as rank, to EDMs and described methods closely related to EDM's for inverse problems. For LDMs, we achieved to reconstruct a set of lines based on their distance with a general minimization algorithm. The rest remains to be explored, but looking at the theory from the previous section, EEDMs (as we defined them) seem to be difficult to manipulate for the entries of the table indicated in red. A non-rigorous but somewhat interesting analogy of the EEDMs would be to compare them to scalars, vectors and matrices. We may at first identify the extension of EDMs to be the distance matrices containing the distances between lines, and then the same for planes etc. But we found that with the definitions of the previous chapter, PLDMs had more properties in common and similarities with EDMs, than LDMs. The analogy mentioned above can be interpreted as follows; scalars share the properties of matrices, as they can be interpreted as $1 \times 1$ matrices, but on the other hand they are also vectors of length 1. Then, "seeing" EDMs as scalars, PLDMs as vectors and the rest as matrices, we may convince ourselves that there exist some hidden structures in EEDMs that link everything together, and that the way that we approached the problem allowed us to mostly decipher PLDMs. However, for a unified theory we may need to study higher order properties.

# Chapter 4

# Hilbert distance matrices

Inspired by David Hilbert's generalization of Euclidean space properties to functions, we discuss in this section the extension of EDMs to Hilbert spaces, therefore using the name Hilbert distance matrix (HDM). The objects we consider here are not vectors anymore but functions, and we will see that this extension comes naturally. We note that in this chapter, we refer to objects belonging to $\mathbb{R}^d$ as "vectors", in contrast to "functions", which will be used for continuous functions. We first define the spaces and functions we consider to avoid ambiguities, the definitions are based from the chapter "From Euclid to Hilbert" of the book by Vetterli et al. [40] where further explanations can be found.

As we considered the $\ell_2$ norm as the distance measure for vectors, we will use here the $\mathcal{L}^2$ distance. Therefore, we require the functions to be square integrable on their domains $I \subseteq \mathbb{R}$, and such functions belong to $\mathcal{L}^2(I)$. For two functions $x, y \in \mathcal{L}^2(I)$, the inner product that induces the $\mathcal{L}^2$ norm is

$$\langle x, y \rangle = \int_I x(t) y^*(t) dt, \tag{4.1}$$

which is the equivalent to the dot product for functions.

**Remark.** *We will use the notation $\langle \cdot, \cdot \rangle$ for the inner product between functions, i.e. $x, y \in \mathcal{L}^2, \langle x, y \rangle = \langle x, y \rangle_{\mathcal{L}^2}$, whereas for vectors $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathbb{R}^d$, we denote the inner product as $\langle \boldsymbol{\alpha}, \boldsymbol{\beta} \rangle_{\ell_2} = \boldsymbol{\beta}^T \boldsymbol{\alpha}$. Since in many applications, for example in signal processing or communications, functions may be complex valued, we consider $x \in \mathbb{C}^I$ but we restrain ourselves to real expansion coefficients, as defined in the next paragraph.*

This gives us the distance between $x$ and $y$:

$$d(x, y) = ||x - y||^2 = ||x||^2 + ||y||^2 - \langle x, y \rangle - \langle y, x \rangle \tag{4.2}$$

$$= ||x||^2 + ||y||^2 - 2\Re(\langle x, y \rangle), \tag{4.3}$$

where $||x||^2 = \langle x, x \rangle$. Moreover, to have analogous results to EDMs, we will consider finite dimensional Hilbert spaces $H \subset \mathcal{L}^2(I)$, where we use the same definition of dimension as in Definition 1.3. Hence, considering $H$ is $d$-dimensional, any element $x \in H$ could be written as a finite sum:

$$x(t) = \sum_{i=1}^{d} \alpha_i \phi_i(t) = \Phi \boldsymbol{\alpha}, \tag{4.4}$$

where $\{\alpha_i\}_{1 \leq i \leq d}$ are called the expansion coefficients and are unique; and $\{\phi_i\}_{1 \leq i \leq d}$ are linearly independent functions such that $H = \text{span}(\Phi)$, since $H$ is finite dimensional. The last term is an

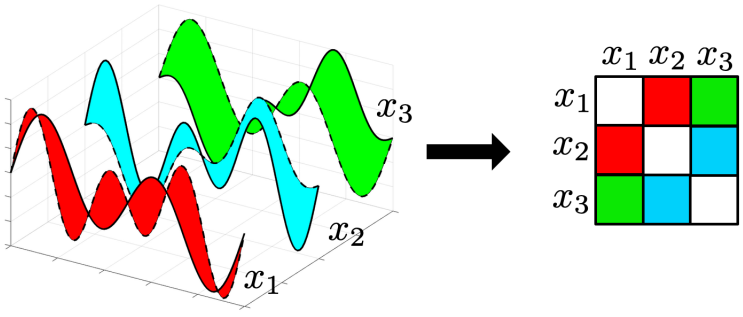



Figure 4.1: Example for HDM with sinusoids over $I = [0, 1]$. The solid lines represent $x_i$'s and the dashed ones $x_j$'s ($j = i \mod 3 + 1$). As shown in the figure, the entries of the HDM can be interpreted intuitively as the "area" squared between the curves.

equivalent notation to express the sum using the linear operator $\Phi : \mathbb{R}^d \to H$. Furthermore, if $\{\phi_i\}_{1 \leq i \leq d}$ form an orthonormal basis of $H$, $\alpha_i = \langle x, \phi_i \rangle$ and $\langle \phi_i, \phi_j \rangle = \mathbb{1}\{i = j\}$. This former expression can be written compactly as $\boldsymbol{\alpha} = \Phi^* x$. We will assume that it is the case in the remaining parts of this text.

Let us now also define $y = \Phi\boldsymbol{\beta} \in H$. Then, replacing this expression and the one in Equation 4.4, in Equation 4.2, we obtain a familiar result given below.

**Lemma 4.1.** *Consider a Hilbert space $H \subset \mathcal{L}^2(I)$ of finite dimension $d$, and an orthonormal basis $\Phi = \{\phi_i\}_{1 \leq i \leq d}$ of $H$. For two elements $x = \Phi\boldsymbol{\alpha}$ and $y = \Phi\boldsymbol{\beta}$ of $H$, the distance in 4.2 can be written as:*

$$d(x, y) = ||\boldsymbol{\alpha}||^2 + ||\boldsymbol{\beta}||^2 - 2\boldsymbol{\beta}^T\boldsymbol{\alpha}. \tag{4.5}$$

*Proof.* This result is a direct consequence of Parseval's theorem, but we do the derivations for completeness. The inner product between $x$ and $y$ can be developed as follows:

$$\begin{aligned}
\langle x, y \rangle = \int_I x(t)y^*(t)dt &= \int_I \sum_{i=1}^{d} \alpha_i \phi_i(t) \sum_{j=1}^{d} \beta_j^* \phi_j^*(t) dt \\
&= \sum_{i=1}^{d} \sum_{j=1}^{d} \alpha_i \beta_j \int_I \phi_i(t)\phi_j^*(t)dt \\
&= \sum_{i=1}^{d} \sum_{j=1}^{d} \alpha_i \beta_j \mathbb{1}\{i = j\} = \boldsymbol{\beta}^T\boldsymbol{\alpha}.
\end{aligned}$$

The other terms in 4.2 can be developed similarly to obtain the desired result. □

This lemma implies an important result; the computation of the distance can either be done in the Hilbert space $H$, or in $\mathbb{R}^d$. The latter corresponds exactly to the distance computations in the case of EDMs. To gain an intuition on why this is the case, we may see $\Phi$ as a linear operator (since it satisfies the linearity properties). $\Phi : \mathbb{R}^d \to H$ is called a basis synthesis operator and $\Phi^* : H \to \mathbb{R}^d$, the basis analysis operator, where $\mathbb{R}^d$ is the space of (square summable) vectors, or sequences, of length $d$.

**Lemma 4.2.** *Considering an orthonormal basis $\Phi$ of a $d$-dimensional vector space $H$, $\Phi^*\Phi = I$ on $\mathbb{R}^d$ and $\Phi\Phi^* = I$ on $H$.*

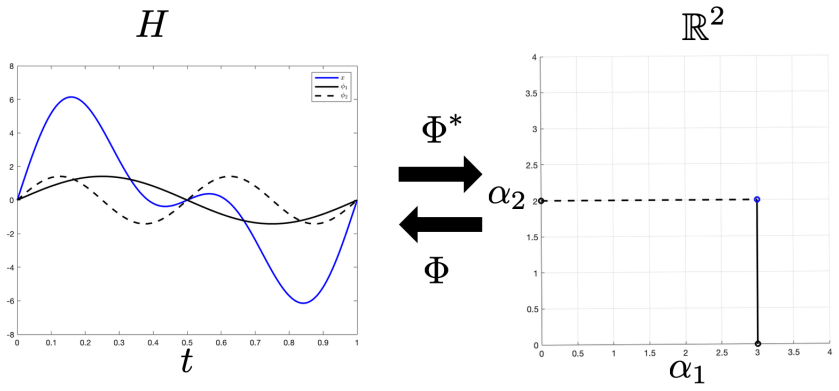



Figure 4.2: In this example, we take $d$=2 and $\Phi = \{\sqrt{2}\sin(2\pi t), \sqrt{2}\sin(4\pi t)\}$ an orthonormal basis for $H = \text{span}(\Phi) \subset \mathcal{L}^2([0,1])$. $x \in H$ is given by $x(t) = \alpha_1\sqrt{2}\sin(2\pi t) + \alpha_2\sqrt{2}\sin(4\pi t)$, with $\alpha_1 = 3$ and $\alpha_2 = 2$.

*Proof.* We have $\langle x, \phi_i\rangle = \sum_{j=1}^{d} \alpha_j\langle\phi_j, \phi_i\rangle = \alpha_i$, which leads to $\Phi^* x = \boldsymbol{\alpha}$, and, since $\Phi$ is an orthonormal basis for $H$, $(\Phi\Phi^* x)(t) = \sum_{i=1}^{d}\langle x, \phi_i\rangle\phi_i(t) = x(t), \forall x \in H$. Therefore, $\Phi\Phi^* = I$ on $H$. On the other hand, $\langle\Phi\boldsymbol{\alpha}, \phi_i\rangle = \sum_{j=1}^{d}\alpha_j\langle\phi_j, \phi_i\rangle = \alpha_i \Rightarrow \Phi^*\Phi\boldsymbol{\alpha} = \boldsymbol{\alpha}, \forall\boldsymbol{\alpha} \in \mathbb{R}^d$. Hence, $\Phi^*\Phi = I$ on $\mathbb{R}^d$. □

Combining previous arguments leads to $x = \Phi\Phi^* x, \forall x \in H$, and $\boldsymbol{\alpha} = \Phi^*\Phi\boldsymbol{\alpha}, \forall\boldsymbol{\alpha} \in \mathbb{R}^d$. Therefore, as stated by Vetterli et al. [40], we have a one-to-one correspondence between the two spaces, where the geometrical properties are preserved, i.e. isometry, which explains the identical results. We can see this as a change of basis; in the case of EDMs, we have $H = \mathbb{R}^d$, thus, $x$ is a $d$-dimensional real vector. A natural choice of basis for $\mathbb{R}^d$ is the standard basis $\{\mathbf{e}_i\}_{1\leq i\leq d}$. This implies $x = \sum_i \alpha_i\mathbf{e}_i = \boldsymbol{\alpha}$. Hence, for EDMs, we are doing the computations directly in $\mathbb{R}^d$. For HDMs, the space of the objects can be replaced by any finite dimensional Hilbert space $H$, and $\boldsymbol{\alpha}$ is the representation of $x$ in $\mathbb{R}^d$ given an orthonormal basis for $H$. An analogy in the context of digital communications would be to go from $d$-dimensional vectors in constellations to signals by pulse shaping. A visual example is given in Figure 4.2.

Coming back to EDMs, consider $n$ points $\{\boldsymbol{\alpha}_j\}_{1\leq j\leq n}$, each belonging to $\mathbb{R}^d$, ascribed to the columns of $A \in \mathbb{R}^{d\times n}$. Combining Equations 1.1 and 4.5, the EDM $D \in \mathbb{R}^{n\times n}$ generated by $A$ is

$$D = \text{edm}(A) = \mathbf{1}_n\text{diag}(A^T A)^T - 2A^T A + \text{diag}(A^T A)\mathbf{1}_n^T. \tag{4.6}$$

Additionally, by Lemma 4.2, $\Phi^*\Phi\boldsymbol{\alpha} = \boldsymbol{\alpha}$, therefore $A = \Phi^*\Phi A$, which implies $A^*A = A^T A = A^*(\Phi^*\Phi A) = (\Phi A)^*(\Phi A)$. Defining $x_j = \Phi\boldsymbol{\alpha}_j$ and $X$ containing $x_j$'s on its "columns", we have $X = \Phi A$, and finally $A^*A = X^*X$. This is the advantage of dealing with Gram matrices, the equivalence $\langle x_i, x_j\rangle = \boldsymbol{\alpha}_j^T\boldsymbol{\alpha}_i$ shown above is also reflected to them. The final result is stated explicitly and summarized below.

**Lemma 4.3.** *With $A, X$ and $\Phi$ as defined above, the HDM $D \in \mathbb{R}^{n\times n}$ of $X$ is given by:*

$$D \triangleq \textit{hdm}(X) \triangleq \mathbf{1}_n\,\textit{diag}(X^*X)^T - 2\Re(X^*X) + \textit{diag}(X^*X)\mathbf{1}_n^T = \textit{edm}(A). \tag{4.7}$$

This lemma makes the link with EDMs and HDMs, therefore any property and algorithm described previously for EDMs can be applied for HDMs. In particular, the invariance to rigid motion is still present, since for an orthogonal matrix $R$ and a vector $\mathbf{b}$, $\text{hdm}(X) = \text{edm}(A) = \text{edm}(RA + R\mathbf{b}\mathbf{1}_n^T) = \text{edm}(R\Phi^* X + R\Phi^*\Phi\mathbf{b}\mathbf{1}_n^T) = \text{edm}(R\Phi^*(X + \Phi\mathbf{b}\mathbf{1}_n^T))$. But $\mathcal{R}(\Phi^*) = \mathcal{R}(R\Phi^*)$, because $R$ is orthogonal. Hence multiplying $X$ by $R\Phi^*$ corresponds to a change of basis, and adding $\Phi\mathbf{b} \in H$

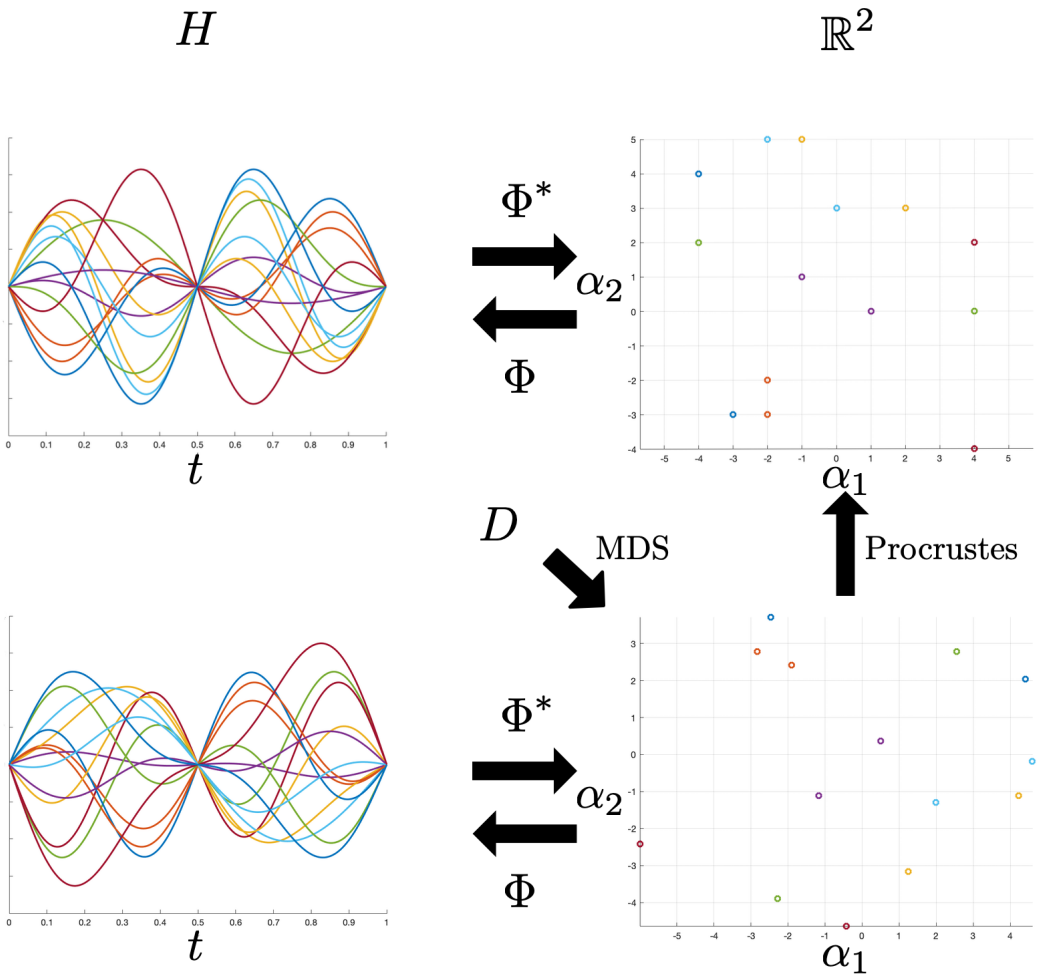



Figure 4.3: We take again $d$=2 and $\Phi = \{\sqrt{2}\sin(2\pi t), \sqrt{2}\sin(4\pi t)\}$ in this example. The point in $\mathbb{R}^2$ generating $D$ have been reconstructed using MDS. Then, Procrustes is applied to match the set to anchor points and finally, we project the points from $\mathbb{R}^2$ to $H = \text{span}(\Phi)$.

to $X$ corresponds to a translation in $H$. On the other hand, the rank properties are still valid. For example, an HDM obtained from functions $x_i = \Phi\boldsymbol{\alpha}_i$ with the same energy, i.e. $||x_i||^2 = C$, $\forall i$ and for a constant $C$, will have rank equal to $d+1$, since $||x_i||^2 = ||\boldsymbol{\alpha}_i||^2$ by Parseval's relationship, and in $\mathbb{R}^d$, $||\boldsymbol{\alpha}_i||^2 = C$ implies that the points lie on the relative boundary of a hypersphere, and the result comes from Theorem 1.4. The procedure to reconstruct the signals for HDMs is very similar to the EDM case, the observed distance matrix $D$ can be decomposed, for example using classical MDS, to obtain the points $A = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_n] \in \mathbb{R}^{d\times n}$ such that $\text{edm}(A) = D$. Then, if we have anchor signals in practice, their expansion coefficients and $A$ could be used for the Orthogonal Procrustes problem to match one another. Finally, the coefficients are "modulated" to the known basis $\Phi$ to obtain the signals. An example is given in Figure 4.3. We also note that this extension can be used on functions $x \in H \subset \mathcal{L}^2(I)$ such that $\textbf{dom}\ x = I$, $x$ is periodic of period $I$, or $\text{supp}(x) = I$, where supp denotes the support (also including isolated and countable 0's).

**Practical Application, Constellation finding:** We end our discussion by giving a simple example application of HDMs in the context of Digital Communications. Consider the setting where we do not know the which constellation the transmitter chose for modulation and we want to estimate it. We suppose that the signaling model is such that each transmitted signal is of

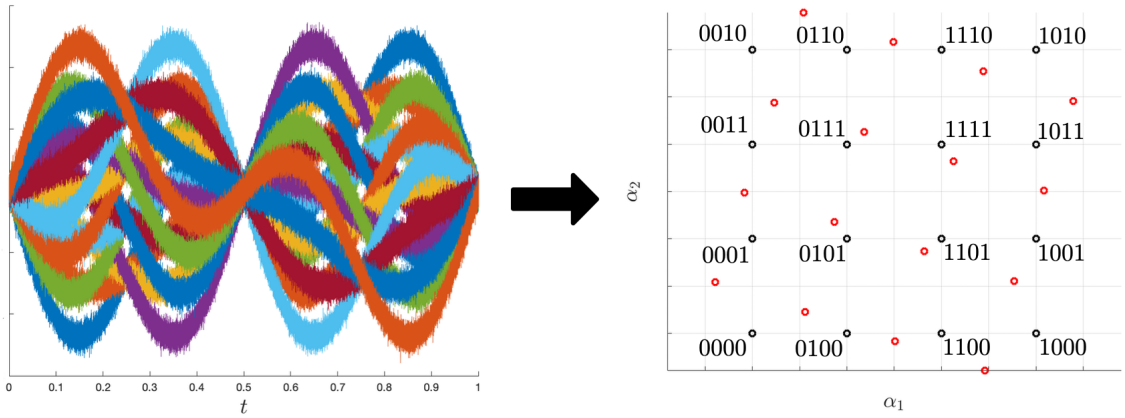



Figure 4.4: Example with $d$=2, $n = 16$ and $\Phi = \{\sqrt{2}\sin(2\pi t), \sqrt{2}\sin(4\pi t)\}$ an orthonormal basis for $H = \text{span}(\Phi) \subset \mathcal{L}^2([0,1])$. The signals on the left are noisy observations of $x_i$'s. The black points correspond to the original constellation and the red ones is obtained by applying the classical MDS Algorithm to the HDM constructed from the noisy observations.

the form $x_i = \Phi\boldsymbol{\alpha}_i$ where the orthonormal basis, or waveforms, $\Phi$ and the length of the signals are known to us. We consider that there are $n$ possible transmitted signals, therefore $\log_2(n)$ bits are encoded per signal, and that the dimensionality is $d$. Additionally, we receive a "training" set containing all possibilities $\{x_i\}_{1\leq i\leq n}$. The channel is supposed to be Additive White Gaussian Noise, therefore we observe $x_i(t) + \epsilon(t)$ where $\epsilon$ is a white Gaussian continuous process of variance $\sigma^2$. Computing the HDM of these signals $\tilde{D}$, which is a noisy observation of the true HDM, we convert the problem into multidimensional scaling, to find the constellation points $\{\boldsymbol{\alpha}_i\}$. An example is given in Figure 4.4. Of course, we would need some additional knowledge to correctly know which point corresponds to which binary symbol, for example the signals could be sent in some known order. Additionally, we would also need to know the average energy used by the transmitter to find a correct translation of the estimated constellation, but in practice, they are usually centered around 0 and Classical MDS as presented above already finds a configuration satisfying this property.

# Conclusion

EDMs have been shown by various studies in the past to possess very interesting theoretical properties which lead to algorithms to reconstruct the original data points that generated the observed matrix, by exploiting their implicit structures. Inspired by these results, we proposed in this text an extension of EDMs to higher dimensional objects. Namely, we asked ourselves if we have similarities between EDMs and matrices with components containing the minimum distance between linear varieties of higher dimensions.

Our analysis showed that in the particular case of PLDMs; which are the special cases where the distances are computed between two subsets, one of the subsets containing points and the other linear varieties of higher dimension; some familiar properties such as rank deficiency are present and closely related to EDMs. For the remaining of the EEDMs, our definitions did not allow to reach similarities with EDMs. An interesting future work would be to study different metrics, replacing our choice of minimum $\ell_2$ distance. For example Grassmannians $G(k, \mathbb{R}^d)$ are spaces which parametrize all $k$-dimensional linear subspaces of $\mathbb{R}^d$, and have been extended to affine subspaces by Lim et al. [54]. There has been various studies on metrics over Grassmannians, and helpful properties could be extracted for our problem. Moreover, Conway et al. [55] looks at how $n$ $k$-dimensional subspaces in ambient space of dimension $d$ should be arranged so that they are as far apart as possible. It could be interesting to find a way to formulate the EEDM inverse problem so as to ask ourselves this question. This is generally referred to as subspace packing in mathematics, and has several applications, such as in Coding Theory. Problem descriptions making use of geometric or multilinear algebra, such as tensors, could also be considered as potential options, since with linear algebra, we saw in Chapter 2 that we have many cases where a closed form solution cannot be written without prior assumptions.

In Chapter 3, we studied inverse problems, involving PLDMs and LDMs, our aim being to reconstruct the original set of linear varieties that generated the observed distance measurements. For PLDMs, we used the rank deficiency of such matrices to denoise and complete the observations using low-rank approximation theory (Algorithm 4), and proposed a way to describe the reconstruction of the original configurations as an EDM completion problem (Algorithm 5). The EDMs to complete are very sparse, in a blocked structure, and obtaining exact reconstructions using existing algorithms may be problematic, but the approximations were close to the original configurations. It would be interesting to adapt the EDM completion algorithms to better fit our particular problem, for example by tuning the parameters optimally, as a future work. Overall, for Algorithm 5 to give good results, knowledge of the distances between points and additionally, some anchors should be known in practice. Moreover, we saw that PLDMs and LDMs could be reconstructed in the presence of noise and missing measurements by minimizing over the points and direction vectors the Frobenius norm between the observed matrix and the EEDM constructed by the variables of the problem. The results were obtained using global optimization tools such as the interior point method. This latter method requires initial estimations as input and we observed that they should be relatively close to the original configurations. To tackle this problem for PLDMs, an approach would be to first use Algorithms 4 and 5 to obtain a good estimation, and then use the global optimization method with these estimations as input if necessary.

As a future study, a similar approach to Edge-MDS (E-MDS) presented by Macagnano and De Abreu [56] could be looked into. This method also takes into account the edges connecting the points, allowing to obtain the angle information in addition to the range information conveyed by EDMs. An improvement of this method is proposed by Baechler et al. [57] where additional constraints are enforced on the edge vectors. Studying more in depth the vectors connecting the minimum distance achieving pair of points in two linear varieties, an adaptation of E-MDS to EEDMs seems possible. This last paper also introduces Coordinate Difference Matrices, independently recovering the coordinates of the edge vectors. Remembering that the edge vectors between two linear varieties belong to the intersection of the spaces spanned by the rows of their normal vectors, a further study would be to express linear varieties implicitly in the problem formulation, instead of parametrically, as we mostly did.

Finally, in the last chapter, we showed the easy transition from vectors to finite dimensional Hilbert spaces for EDM related problems. The matrices obtained in this case, HDMs, share the same properties as EDMs. Essentially, they can be viewed as the same object, since points in $\mathbb{R}^d$ along with $d$ dimensional Hilbert spaces can generate the same distance matrix, as shown by the one-to-one correspondence between these spaces.

# Appendix A

# Derivation of the gradient of the objective in Section 3.1.1

We start with the LDM case and then derive the formula for EDMs by forcing $V = 0$. Denoting the entries of $\text{ldm}(P, V) = D \in \mathbb{R}^{n\times n}$ as $d_{ij}$ and the ones of $\tilde{D}$ as $\tilde{d}_{ij}$, we first rewrite $f$ as a sum of functions.

$$\begin{aligned} f(P,V) &= ||\text{ldm}(P,V) - \tilde{D}||_F^2 \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n}(d_{ij}(P,V))^2 + (\tilde{d}_{ij})^2 - 2d_{ij}(P,V)\tilde{d}_{ij} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} f_{ij}(P,V) \end{aligned}$$

Using the result of the minimum distance between lines from equation 2.5, the minimum distance between line $i$ and line $j$ is given by $d_{ij} = ||\mathbf{p}_i - \mathbf{p}_j||^2 - t_{i\to j}\mathbf{v}_i^T\mathbf{p}_j - t_{j\to i}\mathbf{v}_j^T\mathbf{p}_i$. Then, the partial derivative of $f$ with respect to the entry $(l, k)$ of $P$ and $V$ respectively can be written as:

$$\begin{aligned} \frac{\partial d_{ij}}{\partial P_{lk}} &= \mathbb{1}\{k=i\}\big[2P_{li} - 2P_{lj} - V_{lj}t_{j\to i} - \frac{\partial t_{j\to i}}{\partial P_{lk}}\mathbf{v}_j^T\mathbf{p}_i - \frac{\partial t_{i\to j}}{\partial P_{lk}}\mathbf{v}_i^T\mathbf{p}_j\big] \\ &\quad + \mathbb{1}\{k=j\}\big[2P_{lj} - 2P_{li} - V_{li}t_{i\to j} - \frac{\partial t_{i\to j}}{\partial P_{lk}}\mathbf{v}_i^T\mathbf{p}_j - \frac{\partial t_{j\to i}}{\partial P_{lk}}\mathbf{v}_j^T\mathbf{p}_i\big] \\ \frac{\partial d_{ij}}{\partial V_{lk}} &= \mathbb{1}\{k=i\}\big[-P_{lj}t_{i\to j} - \frac{\partial t_{i\to j}}{\partial V_{lk}}\mathbf{v}_i^T\mathbf{p}_j - \frac{\partial t_{j\to i}}{\partial V_{lk}}\mathbf{v}_j^T\mathbf{p}_i\big] \\ &\quad + \mathbb{1}\{k=j\}\big[-P_{li}t_{j\to i} - \frac{\partial t_{j\to i}}{\partial V_{lk}}\mathbf{v}_j^T\mathbf{p}_i - \frac{\partial t_{i\to j}}{\partial V_{lk}}\mathbf{v}_i^T\mathbf{p}_j\big] \end{aligned}$$

Since $t_{i\to j} = \frac{\mathbf{v}_i^T\mathbf{p}_j + (\mathbf{v}_i^T\mathbf{v}_j)(\mathbf{v}_j^T\mathbf{p}_i)}{1-(\mathbf{v}_i^T\mathbf{v}_j)^2}$ if $i \neq j$, we have:

$$\frac{\partial t_{i\to j}}{\partial P_{lk}} = \mathbb{1}\{k=i\}\big[\frac{V_{lj}\mathbf{v}_i^T\mathbf{v}_j}{1-(\mathbf{v}_i^T\mathbf{v}_j)^2}\big] + \mathbb{1}\{k=j\}\big[\frac{V_{li}}{1-(\mathbf{v}_i^T\mathbf{v}_j)^2}\big]$$

and

$$\begin{aligned} \frac{\partial t_{i\to j}}{\partial V_{lk}} &= \mathbb{1}\{k=i\}\big[\frac{P_{lj} + V_{lj}\mathbf{p}_i^T\mathbf{v}_j + 2V_{lj}t_{i\to j}\mathbf{v}_i^T\mathbf{v}_j}{1-(\mathbf{v}_i^T\mathbf{v}_j)^2}\big] \\ &\quad + \mathbb{1}\{k=j\}\big[\frac{P_{li}\mathbf{v}_i^T\mathbf{v}_j + V_{li}\mathbf{p}_i^T\mathbf{v}_j + 2V_{li}t_{i\to j}\mathbf{v}_i^T\mathbf{v}_j}{1-(\mathbf{v}_i^T\mathbf{v}_j)^2}\big] \end{aligned}$$

The results for $t_{j\to i}$ are very similar and obtained by switching $j$ by $i$ and vice versa. Replacing these expressions in the results above, we obtain:

$$\begin{aligned}\frac{\partial d_{ij}}{\partial P_{lk}} &= 2\mathbb{1}\{k=i\}[P_{li}-P_{lj}-V_{lj}t_{j\to i}]\\ &\quad + 2\mathbb{1}\{k=j\}[P_{lj}-P_{li}-V_{li}t_{i\to j}]\\ \frac{\partial d_{ij}}{\partial V_{lk}} &= 2\mathbb{1}\{k=i\}[-P_{lj}t_{i\to j}-V_{lj}t_{i\to j}t_{j\to i}]\\ &\quad + 2\mathbb{1}\{k=j\}[-P_{li}s_{j\to i}-V_{li}t_{i\to j}t_{j\to i}]\end{aligned}$$

On the other hand, we may write:

$$\frac{\partial f}{\partial V_{lk}} = 2\sum_{i,j}\frac{\partial d_{ij}}{\partial V_{lk}}(d_{ij}-\tilde{d}_{ij})$$

and a similar expression for $P_{lk}$. Putting everything together, we obtain:

$$\begin{aligned}\frac{\partial f}{\partial P_{lk}} &= 4\sum_{i=1}^{n}(P_{lk}-P_{li}-V_{li}t_{i\to k})(2d_{ik}-\tilde{d}_{ik}-\tilde{d}_{ki})\\ \frac{\partial f}{\partial V_{lk}} &= -4\sum_{i=1}^{n}(P_{li}t_{k\to i}+V_{li}t_{k\to i}t_{i\to k})(2d_{ik}-\tilde{d}_{ik}-\tilde{d}_{ki})\end{aligned}$$

Finally, for EDMs, taking $f(X)=||\mathrm{edm}(X)-\tilde{D}||_F^2$, $X=P$ and $V=0$, we have:

$$\frac{\partial f}{\partial X_{lk}} = 4\sum_{i=1}^{n}(X_{lk}-X_{li})(2d_{ik}-\tilde{d}_{ik}-\tilde{d}_{ki})$$

The results in Subsection 3.1.1 are obtained by putting the result in matrix form.

# Bibliography


[1] Miranda Krekovic, Ivan Dokmanic, and Martin Vetterli. Look, no beacons! optimal all-in-one echoslam. *arXiv preprint arXiv:1608.08753*, 2016.

[2] Miranda Kreković, Ivan Dokmanić, and Martin Vetterli. Omnidirectional bats, point-to-plane distances, and the price of uniqueness. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3261–3265. Ieee, 2017.

[3] Karl Menger. Untersuchungen über allgemeine metrik. *Mathematische Annalen*, 100(1):75–163, 1928.

[4] Isaac J Schoenberg. Remarks to maurice frechet's article"sur la definition axiomatique d'une classe d'espace distances vectoriellement applicable sur l'espace de hilbert. *Annals of Mathematics*, pages 724–732, 1935.

[5] Gale Young and Alston S Householder. Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3(1):19–22, 1938.

[6] John Clifford Gower. Euclidean distance geometry. *Math. Sci*, 7(1):1–14, 1982.

[7] John Clifford Gower. Properties of euclidean and non-euclidean distance matrices. *Linear Algebra and its Applications*, 67:81–97, 1985.

[8] Tom L Hayden, Jim Wells, Wei-Min Liu, and Pablo Tarazaga. The cone of distance matrices. *Linear Algebra and its Applications*, 144:153–169, 1991.

[9] Jon Dattorro. *Convex optimization & Euclidean distance geometry*. Lulu. com, 2010.

[10] Rudolf Mathar. The best euclidian fit to a given distance matrix in prescribed dimensions. *Linear Algebra and its Applications*, 67:1–6, 1985.

[11] W Glunt, Tom L Hayden, S Hong, and J Wells. An alternating projection algorithm for computing the nearest euclidean distance matrix. *SIAM Journal on Matrix Analysis and Applications*, 11(4):589–600, 1990.

[12] Warren S Torgerson. Multidimensional scaling: I. theory and method. *Psychometrika*, 17(4): 401–419, 1952.

[13] Peter H Schönemann. On metric multidimensional unfolding. *Psychometrika*, 35(3):349–366, 1970.

[14] Marco Crocco, Alessio Del Bue, and Vittorio Murino. A bilinear approach to the position self-calibration of multiple sensors. *IEEE Transactions on Signal Processing*, 60(2):660–673, 2011.

[15] Peter Hans Schoenemann. *A solution of the orthogonal Procrustes problem with applications to orthogonal and oblique rotation*. PhD thesis, University of Illinois at Urbana-Champaign, 1964.

[16] Al Mead. Review of the development of multidimensional scaling methods. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 41(1):27–39, 1992.

[17] Joseph B Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

[18] Ivan Dokmanic and M Vetterli. Listening to distances and hearing shapes: Inverse problems in room acoustics and beyond. *EPFL, Lausanne*, 2015.

[19] Raghunandan H Keshavan, Andrea Montanari, and Sewoong Oh. Matrix completion from a few entries. *IEEE transactions on information theory*, 56(6):2980–2998, 2010.

[20] Raghunandan H Keshavan, Andrea Montanari, and Sewoong Oh. Matrix completion from noisy entries. *Journal of Machine Learning Research*, 11(Jul):2057–2078, 2010.

[21] Reza Parhizkar. *Euclidean distance matrices: Properties, algorithms and applications*. PhD thesis, Ph. D. dissertation, Ecole Polytechnique Federale de Lausanne (EPFL), 2013.

[22] Yoshio Takane, Forrest W Young, and Jan De Leeuw. Nonmetric individual differences multidimensional scaling: An alternating least squares method with optimal scaling features. *Psychometrika*, 42(1):7–67, 1977.

[23] Abdo Y Alfakih, Amir Khandani, and Henry Wolkowicz. Solving euclidean distance matrix completion problems via semidefinite programming. *Computational optimization and applications*, 12(1-3):13–30, 1999.

[24] Nathan Krislock and Henry Wolkowicz. Euclidean distance matrices and applications. In *Handbook on semidefinite, conic and polynomial optimization*, pages 879–914. Springer, 2012.

[25] Kilian Q Weinberger and Lawrence K Saul. Unsupervised learning of image manifolds by semidefinite programming. *International journal of computer vision*, 70(1):77–90, 2006.

[26] Pratik Biswas, T-C Liang, K-C Toh, Yinyu Ye, and T-C Wang. Semidefinite programming approaches for sensor network localization with noisy distance measurements. *IEEE transactions on automation science and engineering*, 3(4):360–371, 2006.

[27] Ivan Dokmanic, Reza Parhizkar, Juri Ranieri, and Martin Vetterli. Euclidean distance matrices: essential theory, algorithms, and applications. *IEEE Signal Processing Magazine*, 32(6): 12–30, 2015.

[28] Liisa Holm and Chris Sander. Protein structure comparison by alignment of distance matrices. *Journal of molecular biology*, 233(1):123–138, 1993.

[29] Timothy F Havel and Kurt Wüthrich. An evaluation of the combined use of nuclear magnetic resonance and distance geometry for the determination of protein conformations in solution. In *Nmr In Structural Biology: A Collection of Papers by Kurt Wüthrich*, pages 305–318. World Scientific, 1995.

[30] Erik D Demaine, Francisco Gomez-Martin, Henk Meijer, David Rappaport, Perouz Taslakian, Godfried T Toussaint, Terry Winograd, and David R Wood. The distance geometry of music. *Computational geometry*, 42(5):429–454, 2009.

[31] Ivan Dokmanić, Reza Parhizkar, Andreas Walther, Yue M Lu, and Martin Vetterli. Acoustic echoes reveal room shape. *Proceedings of the National Academy of Sciences*, 110(30):12186–12191, 2013.

[32] Ivan Dokmanić, Laurent Daudet, and Martin Vetterli. From acoustic room reconstruction to slam. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6345–6349. Ieee, 2016.

[33] Juri Ranieri, Amina Chebira, Yue M Lu, and Martin Vetterli. Phase retrieval for sparse signals: Uniqueness conditions. *arXiv preprint arXiv:1308.3058*, 2013.

[34] Shuai Huang and Ivan Dokmanić. Reconstructing point sets from distance distributions. *arXiv preprint arXiv:1804.02465*, 2018.

[35] Puoya Tabaghi, Ivan Dokmanić, and Martin Vetterli. Kinetic euclidean distance matrices. *arXiv preprint arXiv:1811.03193*, 2018.

[36] Alejandro Cornejo and Radhika Nagpal. Distributed range-based relative localization of robot swarms. In *Algorithmic Foundations of Robotics XI*, pages 91–107. Springer, 2015.

[37] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110, 2006.

[38] Miranda Kreković, Gilles Baechler, Ivan Dokmanić, and Martin Vetterli. Structure from sound with incomplete data. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3539–3543. IEEE, 2018.

[39] Miranda Krekovic, Ivan Dokmanic, and Martin Vetterli. Shapes from echoes: Uniqueness from point-to-plane distance matrices. *arXiv preprint arXiv:1902.09959*, 2019.

[40] Martin Vetterli, Jelena Kovačević, and Vivek K Goyal. *Foundations of signal processing*. Cambridge University Press, 2014.

[41] Arthur M DuPré and Seymour Kass. Distance and parallelism between flats in rn. *Linear Algebra and its Applications*, 171:99–107, 1992.

[42] Adi Ben-Israel and Thomas NE Greville. *Generalized inverses: theory and applications*, volume 15. Springer Science & Business Media, 2003.

[43] Jürgen Gross and Götz Trenkler. On the least squares distance between affine subspaces. *Linear Algebra and its Applications*, 237:269–276, 1996.

[44] George PH Styan. Hadamard products and multivariate statistical analysis. *Linear algebra and its applications*, 6:217–240, 1973.

[45] Norbert Gaffke and Rudolf Mathar. A cyclic projection algorithm via duality. *Metrika*, 36(1): 29–54, 1989.

[46] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[47] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[48] Zsolt Ugray, Leon Lasdon, John Plummer, Fred Glover, James Kelly, and Rafael Martí. Scatter search and local nlp solvers: A multistart framework for global optimization. *INFORMS Journal on Computing*, 19(3):328–340, 2007.

[49] Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

[50] Leon Mirsky. Symmetric gauge functions and unitarily invariant norms. *The quarterly journal of mathematics*, 11(1):50–59, 1960.

[51] Michael Grant, Stephen Boyd, and Yinyu Ye. Cvx: Matlab software for disciplined convex programming, 2008.

[52] Michael C Grant and Stephen P Boyd. Graph implementations for nonsmooth convex programs. In *Recent advances in learning and control*, pages 95–110. Springer, 2008.

[53] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[54] Lek-Heng Lim, Ken Sze-Wai Wong, and Ke Ye. The grassmannian of affine subspaces. *arXiv preprint arXiv:1807.10883*, 2018.

[55] John H Conway, Ronald H Hardin, and Neil JA Sloane. Packing lines, planes, etc.: Packings in grassmannian spaces. *Experimental mathematics*, 5(2):139–159, 1996.

[56] Davide Macagnano and Giuseppe Thadeu Freitas De Abreu. Algebraic approach for robust localization with heterogeneous information. *IEEE Transactions on Wireless Communications*, 12(10):5334–5345, 2013.

[57] Gilles Baechler, Frederike Diimbgen, Golnoosh Elhami, Miranda Krekovic, Robin Scheibler, Adam Scholefield, and Martin Vetterli. Combining range and direction for improved localization. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3484–3488. IEEE, 2018.