OPT2023: 15th Annual Workshop on Optimization for Machine Learning

Riemannian Optimization for Euclidean Distance Geometry

Chandler Smith CHANDLER.SMITH@TUFTS.EDU
Samuel Lichtenberg SPLICHTE@GMAIL.COM
Department of Mathematics, Tufts University, Medford, MA 02155, USA.

HanQin Cai HQCAI@UCF.EDU
Department of Statistics and Data Science and Department of Computer Science, University of Central Florida, Orlando, FL 32816, USA.

Abiy Tasissa ABIY.TASISSA@TUFTS.EDU
Department of Mathematics, Tufts University, Medford, MA 02155, USA.

Abstract

The Euclidean distance geometry (EDG) problem is a crucial machine learning task that appears in many applications. Utilizing the pairwise Euclidean distance information of a given point set, EDG reconstructs the configuration of the point system. When only partial distance information is available, matrix completion techniques can be incorporated to fill in the missing pairwise distances. In this paper, we propose a novel dual basis Riemannian gradient descent algorithm, coined RieEDG, for the EDG completion problem. The numerical experiments verify the effectiveness of the proposed algorithm. In particular, we show that RieEDG can precisely reconstruct various datasets consisting of 2- and 3-dimensional points by accessing a small fraction of pairwise distance information.

1. Introduction

Due to the rapid and continually accelerating proliferation of data collection across domains, data sizes have exploded dramatically. Oftentimes these data are high dimensional or corrupted, posing a considerable challenge for analysis. Compounding this challenge, these data are often incomplete. One example where this manifests is in the Euclidean distance geometry (EDG) problem. Given partial pairwise distance information of a set of n points in $\mathbb{R}^k$, the EDG completion problem is to reconstruct the configuration of the point system. The applications of this model are numerous, with appearances of the EDG problem appearing in sensor localization [1, 4], dimensionality reduction [20], computational chemistry [6, 7, 13, 22], robot kinematics and position analysis [15, 17], and more recently in visualization of antibody-virus interactions [5]. In these applications, collecting all pairwise distances is often infeasible or too costly.

To introduce this problem mathematically, let $\{\mathbf{p}_i\}_{i=1}^n \subset \mathbb{R}^k$ be a given k-dimensional dataset, and denote $\mathbf{P} := [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_n]^T \in \mathbb{R}^{n\times k}$ as the matrix where the i-th row is the i-th point in the dataset. The squared Euclidean distance between the points $\mathbf{p}_i$ and $\mathbf{p}_j$ is given by $d_{i,j}^2 := \|\mathbf{p}_i - \mathbf{p}_j\|_2^2 = \|\mathbf{p}_i\|_2^2 + \|\mathbf{p}_j\|_2^2 - 2\mathbf{p}_i^T\mathbf{p}_j$. The collection of pairwise distances can be neatly represented in symmetric matrix form, termed *Squared Euclidean Distance Matrix* $\mathbf{D} := [d_{i,j}^2] \in \mathbb{R}^{n\times n}$— heretofore referred to as *Distance Matrix*. In addition to the distance matrix, we can also construct the *Gram Matrix* $\mathbf{X} := \mathbf{P}\mathbf{P}^T \in \mathbb{R}^{n\times n}$ of the dataset. In this paper we are focusing on the point configurations centered at the origin, i.e., $\mathbf{P}^T\mathbf{1} = \mathbf{0}$, and the corresponding *centralized* Gram matrix for computational ease. The centralized Gram matrix and distance matrix store the same point

configuration information, albeit represented differently, and closed-form relationships between $\mathbf{D}$ and $\mathbf{X}$ exist. The distance matrix $\mathbf{D}$ can be computed from the Gram matrix by $\mathbf{D} = \mathbf{1}\text{diag}(\mathbf{X})^T + \text{diag}(\mathbf{X})\mathbf{1}^T - 2\mathbf{X}$ where $\text{diag}(\cdot)$ is an operator that takes the diagonal elements of a matrix and represents it as a column vector and $\mathbf{1} \in \mathbb{R}^n$ is a vector of ones. On the other hand, the Gram matrix $\mathbf{X}$ and the configuration matrix $\mathbf{P}$ are not unique for a given $\mathbf{D}$ since a point configuration can shift and rotate without changing the relative positions. See Figure 1 for an illustration. However, for all the configurations that are centered at the same location, the Gram matrix is fixed no matter how the configuration is rotated. Let $\mathbf{J} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T$, where $\mathbf{I} \in \mathbb{R}^{n\times n}$ is the identity matrix. The centralized Gram matrix can be computed from the distance matrix by $\mathbf{X} = -\frac{1}{2}\mathbf{J}\mathbf{D}\mathbf{J}$. Then, one of the equivalent point configurations can be computed by $\mathbf{P} = \mathbf{U}\mathbf{\Lambda}^{1/2}$ where $\mathbf{X} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ is the eigenvalue decomposition. For more theoretical details about the equivalence of the shifted and rotated configurations, we refer the interested reader to [8, 21].

Assuming $\mathbf{P}$ is full rank, i.e., points $\{\mathbf{p}_i\}_{i=1}^n$ are not all embedded in some low-dimensional subspace of $\mathbb{R}^k$, the rank of $\mathbf{X}$ is k and the rank of $\mathbf{D}$ is at most $k+2$. When only partial distance information is observed in $\mathbf{D}$, the classical approach is to recover the entire distance matrix $\mathbf{D}$ with some off-the-shelf matrix completion algorithms such as [3, 9, 11, 16, 23], then compute $\mathbf{X}$ and $\mathbf{P}$ from the recovered $\mathbf{D}$. However, the classical approach ignores the implicit constraints of the Euclidean distances, e.g., non-negativity and triangle inequality, and thus has sub-optimal sampling complexity on the pairwise distances. Some recent works [12, 18, 19] show the convex formulation that enforces the positive semidefinite constraint on the Gram matrix $\mathbf{X}$ using a dual basis approach will implicitly enforce the constraints of the Euclidean distances and achieve optimal sampling complexity. However, those convex algorithms are computationally and/or memory expensive. Hence, high-efficient non-convex dual basis algorithms are valuable for the EDG completion problem.

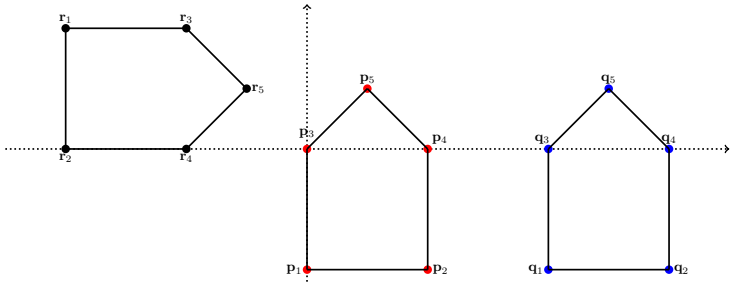

Figure 1: The set of 5 points $\{\mathbf{p}_1, ..., \mathbf{p}_5\}$,$\{\mathbf{q}_1, ..., \mathbf{q}_5\}$ and $\{\mathbf{r}_1, .., \mathbf{r}_5\}$ have different configurations but the Euclidean distance matrix is the same for all of them. In other words, these sets of points are equivalent up to translation and rotation.

Notation. Lowercase and uppercase boldface letters (e.g., $\mathbf{v}$ and $\mathbf{M}$) denote column vectors and matrices, respectively. Blackboard-bold letters (e.g., $\mathbb{S}$) denote spaces. $\|\cdot\|_2$ and $||\cdot||_F$ denote the ℓ_2-norm and Frobenius norm, respectively. $\langle\cdot,\cdot\rangle$ denotes the trace inner product. The $\mathbf{1}$ denotes a vector whose entries are all 1. Given a matrix $\mathbf{A}$, $\mathbf{a}_i$ denotes its i-th column, $\sigma_{\max}(\mathbf{A})$ and $\sigma_{\min}(\mathbf{A})$ denote the largest and smallest non-zero singular values of $\mathbf{A}$. $\{\mathbf{e}_j\}$ and $\{\mathbf{E}_{i,j}\}$ represent the standard basis of vector and matrix spaces, respectively. $\texttt{EVD}_k(\mathbf{A})$ denotes the rank-k truncated eigenvalue decomposition of a symmetric matrix $\mathbf{A}$.

2. Preliminaries

In this section, we provide a brief summary of the dual basis for EDG developed in [18]. More preliminaries about the Riemannian matrix completion algorithm [23] are provided in the appendix.

In the EDG problem, we observe some entries of a squared distance matrix $\mathbf{D}$, and wish to recover the underlying Gram matrix $\mathbf{X}$. The relationship between $\mathbf{X}$ and $\mathbf{D}$ is $X_{i,i} + X_{j,j} - 2X_{i,j} = D_{i,j}$. Given $\mathbf{D}$, $\mathbf{X}$ is only unique up to translations and rotations, so it is assumed that $\mathbf{X1} = \mathbf{0}$. We are especially interested in the case when $\mathbf{X}$ is low rank, which frequently occurs in practice.

To handle this, the authors in [18] formulated the EDG problem as matrix recovery with respect to the operator basis $\mathbf{w}_{\boldsymbol{\alpha}} = \mathbf{E}_{\alpha_1,\alpha_1} + \mathbf{E}_{\alpha_2,\alpha_2} - \mathbf{E}_{\alpha_1,\alpha_2} - \mathbf{E}_{\alpha_2,\alpha_1}$ for $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$ and $\alpha_1 < \alpha_2$, which spans the dimension $L = \frac{n(n-1)}{2}$ linear subspace $\mathbb{S} = \{\mathbf{X} \in \mathbb{R}^{n\times n} | \mathbf{X} = \mathbf{X}^T,\ \mathbf{X} \cdot \mathbf{1} = \mathbf{0}\}$. This basis was chosen because it reconstructs the measurements $D_{i,j}$ via inner products $\langle \mathbf{X}, \mathbf{w}_{i,j} \rangle$. Unfortunately, this basis is not orthogonal, so a biorthogonal dual basis was introduced: given $\{\mathbf{w}_{\boldsymbol{\alpha}}\}_{\alpha=1}^{L}$, define the matrix $\mathbf{H}$ as $\mathbf{H}_{\alpha,\,\beta} = \langle \mathbf{w}_{\boldsymbol{\alpha}}\,, \mathbf{w}_{\boldsymbol{\beta}} \rangle$; the set of matrices $\mathbf{v}_{\boldsymbol{\alpha}} = \sum_{\beta} \mathbf{H}^{-1}_{\alpha,\,\beta} \mathbf{w}_{\boldsymbol{\beta}}$ then forms a dual basis to $\{\mathbf{w}_{\boldsymbol{\alpha}}\}$ satisfying $\langle \mathbf{v}_{\boldsymbol{\alpha}}\,, \mathbf{w}_{\boldsymbol{\beta}} \rangle = \delta_{\alpha,\,\beta}$. In the dual basis expansion, $\mathbf{X} = \sum_{\alpha} \langle \mathbf{X}\,, \mathbf{w}_{\boldsymbol{\alpha}} \rangle \mathbf{v}_{\boldsymbol{\alpha}}$, so the EDG problem becomes the recovery of a low-rank matrix $\mathbf{X}$ of rank k given a few of its expansion coefficients. The difficulty with a dual basis approach can be finding an explicit form for $\mathbf{v}_{\boldsymbol{\alpha}}$. In the case of the EDG problem, the form of $\mathbf{v}_{\boldsymbol{\alpha}}$ is known [14].

3. A Non-Convex Approach for EDG Completion

In this section, we propose a new algorithm for the EDG completion problem. The main goal is fusing non-orthogonal matrix completion [18] for the EDG problem with a Riemannian-based gradient descent approach [2, 10, 23]. Our algorithm is based on the sampling operator defined as:

$$\mathcal{R}_{\Omega} : \mathbf{X} \in \mathbb{S} \longrightarrow \frac{L}{m} \sum_{\boldsymbol{\alpha} \in \Omega} \langle \mathbf{X}\,, \mathbf{w}_{\boldsymbol{\alpha}} \rangle \mathbf{v}_{\boldsymbol{\alpha}}, \tag{1}$$

where $\Omega \subset [L]$, $|\Omega| = m$. In the prior work, the goal was to construct a Riemannian-based non-convex algorithm for directly observed entries. In the EDG problem, we want to complete the Gram matrix $\mathbf{X}$ through entries of the distance matrix $\mathbf{D}$. By analogy to [23], we can now define the following objective function using the operator $\mathcal{R}^{\star}_{\Omega}\mathcal{R}_{\Omega}$:

$$\begin{aligned} \underset{\mathbf{X} \in \mathbb{S}^{+}}{\text{minimize}} \quad & \langle \mathbf{X} - \mathbf{M}, \mathcal{R}^{\star}_{\Omega}\mathcal{R}_{\Omega}(\mathbf{X} - \mathbf{M}) \rangle \\ \text{subject to} \quad & \text{rank}(\mathbf{X}) = k, \end{aligned} \tag{2}$$

where $\mathcal{R}^{\star}_{\Omega}\mathcal{R}_{\Omega}$ is defined as:

$$\mathcal{R}^{\star}_{\Omega}\mathcal{R}_{\Omega} : \mathbf{X} \in \mathbb{S} \rightarrow \frac{L^2}{m^2} \sum_{\boldsymbol{\alpha}, \boldsymbol{\beta} \in \Omega} \langle \mathbf{X}, \mathbf{w}_{\boldsymbol{\alpha}} \rangle \langle \mathbf{v}_{\boldsymbol{\alpha}}, \mathbf{v}_{\boldsymbol{\beta}} \rangle \mathbf{w}_{\boldsymbol{\beta}}, \tag{3}$$

and $\mathbb{S}^{+} = \mathbb{S} \cap \{\mathbf{X} \in \mathbb{R}^{n\times n} \mid \mathbf{X} \succeq \mathbf{0}\}$ is the set of positive semidefinite matrices centered at the origin. For the EDG problem, the rank constraint corresponds to the dataset dimension. Note that in the setting where all the distance measurements are available, the sampling operator is the identity operator, up to a scaling factor. Consequently, the sampling operator defined in equation (3) can be viewed as an approximation of the identity operator on set S, given the partial measurements $\langle \mathbf{X}, \mathbf{w}_{\boldsymbol{\alpha}} \rangle_{\boldsymbol{\alpha} \in \Omega}$ and $\{\mathbf{v}_{\boldsymbol{\alpha}}\}_{\boldsymbol{\alpha} \in \Omega}$, up to a scaling factor. It could be surmised that a more natural and alternative substitute is $\mathcal{R}_{\Omega}$. Nevertheless, utilizing $\mathcal{R}_{\Omega}$ is not feasible since the optimization based on it would necessitate knowledge of $\langle \mathbf{M}, \mathbf{v}_{\boldsymbol{\alpha}} \rangle_{\boldsymbol{\alpha} \in \Omega}$-information that is not accessible. This arises from the fact that, in the EDG problem, the inputs only consist of the scalars $\{\langle \mathbf{M}, \mathbf{w}_{\boldsymbol{\alpha}} \rangle\}_{\mathbf{w}_{\boldsymbol{\alpha}} \in \Omega}$. In the following theorem, we establish a direct relationship between $\mathcal{R}_{\Omega}$ and $\mathcal{P}_{\Omega}$. This relationship will be utilized later to construct a computationally tractable representation of $\mathcal{R}^{\star}_{\Omega}\mathcal{R}_{\Omega}$.

Theorem 3.1 *Let* $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2 ..., \mathbf{p}_n]^T \in \mathbb{R}^{n\times k}$ *denote the given set of points and consider any subset* $\Omega \in [L]$. *For* $\mathbf{X} = \mathbf{P}\mathbf{P}^T$ *and* $\mathbf{D} = [d_{ij}^2]$ *where* $d_{ij}^2 = \|\mathbf{p}_i - \mathbf{p}_j\|_2^2$, *it follows that*

$$\mathcal{R}_\Omega(\mathbf{X}) = -\frac{L}{m} \cdot \frac{1}{2}\mathbf{J}\mathcal{P}_\Omega(\mathbf{D})\mathbf{J}, \tag{4}$$

where $\mathcal{P}_\Omega(\mathbf{D}) = \sum_{(i,j)\in\Omega} (\langle \mathbf{D},\, \mathbf{E}_{i,j}\rangle \mathbf{E}_{i,j} + \langle \mathbf{D},\, \mathbf{E}_{j,i}\rangle \mathbf{E}_{j,i})$.

The proof is deferred to the Appendix. This problem equivalence allows us to instead study completion using $\mathcal{R}_\Omega$ instead of completion using $\mathcal{P}_\Omega$, and leads us into the definition of our proposed algorithm. Following the procedure laid out in Section A, we define the following algorithm to solve the partial EDG problem.

Algorithm 1: $\mathcal{R}_\Omega^\star\mathcal{R}_\Omega$ Riemannian Gradient Descent for EDG Completion (RieEDG)
Input: $\mathcal{P}_\Omega(\mathbf{D})$: the observed distance information, k: the dimension of the points and η: the step size.
$\mathbf{X}_0 = \texttt{EVD}_k(\mathcal{R}_\Omega^\star\mathcal{R}_\Omega(\mathbf{X})) = \mathbf{U}_0\mathbf{\Lambda}_0\mathbf{U}_0^T$
For $i = 0, 1...$
  $\mathbf{G}_l = \mathcal{R}_\Omega^\star\mathcal{R}_\Omega(\mathbf{X} - \mathbf{X}_l)$;
  $\mathbf{W}_l = \mathbf{X}_l + \eta\mathcal{P}_{\mathbb{T}_l}(\mathbf{G}_l)$;
  $\mathbf{X}_{l+1} = \texttt{EVD}_k(\mathbf{W}_l)$;
End
Output: $\mathbf{X}_{\text{rev}}$

In what follows, we provide the per-iteration computational complexities of each steps.
Computation of gradient: The gradient term requires computing efficiently $\mathcal{R}_\Omega^\star\mathcal{R}_\Omega(\mathbf{Y})$. Let $T(\mathbf{X})$ denote the map from a Gram matrix to a squared distance matrix. Some calculation yields that,

$$\mathcal{R}_\Omega^\star\mathcal{R}_\Omega(\mathbf{Y}) = \frac{L^2}{m^2} \cdot \frac{1}{4}T^* \left(\mathcal{P}_\Omega(\mathbf{J}\mathcal{P}_\Omega(T(\mathbf{Y}))\mathbf{J})\right),$$

where T^* is the adjoint operator of T. The total complexity of computing $\mathcal{R}_\Omega^\star\mathcal{R}_\Omega$ is $O(|\Omega|)$. The proof of this form of $\mathcal{R}_\Omega^\star\mathcal{R}_\Omega$ is deferred to the appendix.
Gradient descent and eigenvalue decomposition: Using the technique introduced in [23], $\mathbf{W}_l$ does not need to be directly computed; rather we only compute quantities relevant to the next step of truncated eigenvalue decomposition. The main steps are: (a) forming two intermediate matrices that costs $O(|\Omega|k)$, (b) computing QR factorizations of the two $n \times k$ matrices that costs $O(nk^2)$ and (c) rank-k truncated eigenvalue decomposition of $(2k \times 2k)$ matrix, which costs $O(k^3)$.

4. Numerical Results

To test these algorithms, various 2- and 3-dimensional datasets were used and are referred to in Table 1 with their corresponding number of datapoints. The objective of RieEDG is to recover the full set of points $\mathbf{P}$ up to orthogonal transformation from a subset of entries of $\mathbf{D}$ chosen using a Bernoulli sampling model, where each entry has a probability γ of being selected for $\gamma \in [0, 1]$, with an expected γL entries chosen. RieEDG outputs the Gram matrix $\mathbf{X} = \mathbf{P}\mathbf{P}^T$, from which $\mathbf{P}$ can be recovered. The comparison referenced in Tables 1 and 2 are the relative error between the recovered matrix $\mathbf{X}_{\text{rev}}$ and the ground truth matrix $\mathbf{X}$ in Frobenius norm. Each run was terminated at either 500 iterations or a relative difference between iterates of 10^{-7}.

Table 1: Relative recovery error $\|\mathbf{X} - \mathbf{X}_{\text{rev}}\|_F / \|\mathbf{X}\|_F$ between the recovered Gram matrix and the true Gram matrix averaged over 10 trials using RieEDG. The 5% timing is the measured average time to achieve recovery at a 5% sampling rate.

Dataset \ γ	5%	3%	2%	1%	5% Timing (sec)
Sphere (3D, $n = 1002$)	6.2e-07	1.2e-06	9.52e-03	1.08	4.62
U.S. Cities (2D, $n = 2920$)	5.90e-07	1.613-03	0.0168	0.0796	135
Cow (3D, $n = 2601$)	5.58e-07	8.62e-06	1.50e-06	0.0095	67.4
Swiss Roll (3D, $n = 2048$)	5.04e-07	8.84e-07	1.14e-06	0.0604	30.9

Table 2: Relative recovery error $\|\mathbf{X} - \mathbf{X}_{\text{rev}}\|_F / \|\mathbf{X}\|_F$ between the recovered Gram matrix and the true Gram matrix averaged over 10 trials using [18]. The 5% timing is the measured average time to achieve recovery at a 5% sampling rate.

Dataset \ γ	5%	3%	2%	1%	5% Timing (sec)
Sphere (3D, $n = 1002$)	4.3e-04	0.0013	0.0026	0.62	5.41
U.S. Cities (2D, $n = 2920$)	4.4e-04	6.7e-04	7.6e-04	0.0016	41.0
Cow (3D, $n = 2601$)	2.9e-04	3.5e-04	4.7e-04	0.0010	26.6
Swiss Roll (3D, $n = 2048$)	5.3e-04	6.4e-04	7.6e-04	0.0041	21.1

The strong reconstruction properties of RieEDG indicate its viability for use as a tool for the EDG problem. The transition from effective reconstruction to poor reconstruction at around 2% is indicative of the probabilistic nature of reconstruction, as some trials were successful and others performed very poorly, but with increasing dataset size this problem is mitigated. This indicates effective scaling of RieEDG for large datasets in the small sampling rate regime. When compared to the algorithm developed in [18], RieEDG largely provides a higher degree of accuracy while giving speedups for smaller datasets, particularly in the high sampling regime. It is noticeably slower for larger datasets and is outperformed in the small sample regime, indicating better scaling and transition properties for [18]. Work is currently being done to better understand how to initialize RieEDG to improve low-sampling recovery.

5. Conclusion and Future Directions

In this project we constructed an efficient non-convex algorithm for the Euclidean distance geometry problem via fusing a Riemannian gradient descent-based approach with a dual basis approach to provide a more natural constraint set for the optimization routine. RieEDG demonstrates strong reconstruction properties on par with existing methods, and is based on strong theoretical foundations. Future work is predominately dedicated towards fully proving theoretical guarantees for convergence, as well as considering optimal ways to initialize the algorithm.

Acknowledgments

This work is partially supported by NSF DMS 2208392 and NSF DMS 2304489.

References

[1] Pratik Biswas, Tzu-Chen Lian, Ta-Chung Wang, and Yinyu Ye. Semidefinite programming based algorithms for sensor network localization. *ACM Transactions on Sensor Networks (TOSN)*, 2(2):188–220, 2006.

[2] HanQin Cai, Jian-Feng Cai, and Ke Wei. Accelerated alternating projections for robust principal component analysis. *Journal of Machine Learning Research*, 20(1):685–717, 2019.

[3] Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

[4] Yichuan Ding, Nathan Krislock, Jiawei Qian, and Henry Wolkowicz. Sensor network localization, euclidean distance matrix completions, and graph realization. *Optimization and Engineering*, 11(1):45–66, 2010.

[5] Tal Einav, Yuehaw Khoo, and Amit Singer. Quantitatively visualizing bipartite datasets. *Physical Review X*, 13(2):021002, 2023.

[6] Xingyuan Fang and Kim-Chuan Toh. Using a distributed sdp approach to solve simulated protein molecular conformation problems. In *Distance Geometry*, pages 351–376. Springer, 2013.

[7] W Glunt, TL Hayden, and M Raydan. Molecular conformations from distance matrices. *Journal of Computational Chemistry*, 14(1):114–120, 1993.

[8] J. C. Gower. Properties of euclidean and non-euclidean distance matrices. *Linear Algebra and its Applications*, 67:81–97, 6 1985. ISSN 0024-3795. doi: 10.1016/0024-3795(85)90187-9.

[9] David Gross. Recovering low-rank matrices from few coefficients in any basis. *Information Theory, IEEE Transactions on*, 57(3):1548–1566, 2011.

[10] Keaton Hamm, Mohamed Meskini, and HanQin Cai. Riemannian CUR decompositions for robust principal component analysis. In *Topological, Algebraic and Geometric Learning Workshops 2022*, pages 152–160. PMLR, 2022.

[11] Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 665–674, 2013.

[12] Rongjie Lai and Jia Li. Solving partial differential equations on manifolds from incomplete interpoint distance. *SIAM Journal on Scientific Computing*, 39(5):A2231–A2256, 2017.

[13] Leo Liberti, Carlile Lavor, and Nelson Maculan. A branch-and-prune algorithm for the molecular distance geometry problem. *International Transactions in Operational Research*, 15(1): 1–17, 2008.

[14] Samuel Lichtenberg and Abiy Tasissa. A dual basis approach to multidimensional scaling: spectral analysis and graph regularity, 2023.

[15] Josep M Porta, Nicolás Rojas, and Federico Thomas. Distance geometry in active structures. *Mechatronics for Cultural Heritage and Civil Engineering*, pages 115–136, 2018.

[16] Benjamin Recht, Maryam Fazel, and Pablo A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[17] Nicolàs Rojas. *Distance-based formulations for the position analysis of kinematic chains*. PhD thesis, Universitat Politècnica de Catalunya, 2012.

[18] Abiy Tasissa and Rongjie Lai. Exact reconstruction of euclidean distance geometry problem using low-rank matrix completion. *IEEE Transactions on Information Theory*, 65(5):3124–3144, 2018.

[19] Abiy Tasissa and Rongjie Lai. Low-rank matrix completion in a general non-orthogonal basis. *Linear Algebra and its Applications*, 625:81–112, 2021.

[20] Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

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

[22] Michael W Trosset. Applications of multidimensional scaling to molecular conformation, 1997.

[23] Ke Wei, Jian-Feng Cai, Tony F. Chan, and Shingyu Leung. Guarantees of Riemannian optimization for low rank matrix completion. *Inverse Problems and Imaging*, 14(2):233–265, 2020. ISSN 1930-8337. doi: 10.3934/ipi.2020011.

Appendix A. Riemannian Matrix Completion

In [23], the authors propose the following non-convex objective function for matrix completion:

$$\begin{aligned} \underset{\mathbf{X}\in\mathbb{R}^{n\times n}}{\text{minimize}} \quad & \langle \mathbf{X}-\mathbf{M}, \mathcal{P}_\Omega(\mathbf{X}-\mathbf{M})\rangle \\ \text{subject to} \quad & \text{rank}(\mathbf{X}) = k, \end{aligned} \tag{5}$$

for some fixed rank k and where

$$\mathcal{P}_\Omega : \mathbf{X} \in \mathbb{R}^{n\times n} \longrightarrow \sum_{(i,j)\in\Omega} \langle \mathbf{X},\, \mathbf{E}_{i,j}\rangle \mathbf{E}_{i,j}. \tag{6}$$

The work in [23] employs a Riemannian-based gradient descent scheme to solve this problem, with strong guarantees that the true matrix $\mathbf{M}$ is recovered with high probability given a good initialization. More specifically, the algorithm in [23] is a projected gradient descent algorithm, where the estimates are updated in the Riemannian gradient descent direction of the objective described in (5) followed by a retraction mapping onto the manifold of rank-k matrices via rank-k truncated singular value decomposition (SVD).

More mathematically, let $\mathbf{X}_l = \mathbf{U}_l\mathbf{\Sigma}_l\mathbf{V}_l^T$ be the estimate at the l-th iteration of the algorithm and let $\mathbb{T}_l$ be the tangent space of the rank k manifold centered at $\mathbf{X}_l$. Explicitly, $\mathbb{T}_l$ has the form $\mathbb{T}_l = \{\mathbf{U}_l\mathbf{Y} + \mathbf{Z}\mathbf{V}_l^T : \mathbf{Y} \in \mathbb{R}^{k\times n}, \mathbf{Z} \in \mathbb{R}^{n\times k}\}$. To update to $\mathbf{X}_{l+1}$, the gradient descent step is taken on the tangent space with the step size α_l followed by SVD truncation step. The updates have the following form

$$\mathbf{X}_{l+1} = \texttt{SVD}_k(\mathbf{X}_l + \eta_l \mathcal{P}_{\mathbb{T}_l}\mathcal{P}_\Omega(\mathbf{X}-\mathbf{X}_l)), \tag{7}$$

where $\texttt{SVD}_k$ denotes the rank-k truncated SVD. The step size η_l is optimally determined using exact line search.

Appendix B. Proof of Theorem 3.1

Proof Given a sample Ω of indices $i < j \in [n] \times [n]$, assume we have the following sampling operators on the Gram matrix $\mathbf{X}$ and the distance matrix $\mathbf{D}$:

$$\begin{aligned} &\mathcal{P}_\Omega(\mathbf{D})_{a,b} = D_{a,b} \text{ if } (a,b) \text{ or } (b,a) \in \Omega, \text{ 0 otherwise} \\ &\mathcal{R}_\Omega(\mathbf{X}) = \sum_{\boldsymbol{\alpha}\in\Omega} \langle \mathbf{X}, \mathbf{w}_{\boldsymbol{\alpha}}\rangle \mathbf{v}_{\boldsymbol{\alpha}}. \end{aligned}$$

We aim to show that $\mathcal{R}_\Omega(\mathbf{X}) = -\frac{1}{2}\mathbf{J}\mathcal{P}_\Omega(\mathbf{D})\mathbf{J}$. We will start by expanding the sum for $\mathcal{R}_\Omega(\mathbf{X})$ at a particular entry (a,b) using the form of the dual basis $\mathbf{v}_{i,j}$ from [14]. We have that

$$\begin{aligned} \mathcal{R}_\Omega(\mathbf{X}) &= \sum_{\alpha\in\Omega} \langle \mathbf{X}, \mathbf{w}_{\boldsymbol{\alpha}}\rangle \mathbf{v}_\alpha \\ &= \sum_{i<j\in\Omega} D_{i,j}\mathbf{v}_{i,j} \\ &= -\frac{1}{2}\sum_{i<j\in\Omega} D_{i,j}\big(\mathbf{J}(:,i)\mathbf{J}(j,:) + \mathbf{J}(:,j)\mathbf{J}(i,:)\big). \end{aligned}$$

Taking the (a, b)-th entry, we get that

$$\big[\mathcal{R}_\Omega(\mathbf{X})\big]_{a,b} = \Big[-\frac{1}{2} \sum_{i<j\in\Omega} D_{i,j}\big((\mathbf{J}(:,i)\mathbf{J}(j,:) + \mathbf{J}(:,j)\mathbf{J}(i,:)\big)\Big]_{a,b} = -\frac{1}{2} \sum_{i<j\in\Omega} D_{i,j}\Big(J_{a,i}J_{j,b} + J_{a,j}J_{i,b}\Big).$$

We proceed to examine the $\mathbf{J}$-derived terms. We have that $\mathbf{J} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$, and so, for any a, i, we see that $J_{a,i} = I_{a,i} - \frac{1}{n} = \delta_a^i - \frac{1}{n}$. Therefore, we have

$$\begin{aligned}
\big[\mathcal{R}_\Omega(\mathbf{X})\big]_{a,b} &= -\frac{1}{2} \sum_{i<j\in\Omega} D_{i,j}\Big(J_{a,i}J_{j,b} + J_{a,j}J_{i,b}\Big) \\
&= -\frac{1}{2} \sum_{i<j\in\Omega} D_{i,j}\Big((\delta_a^i - \frac{1}{n})(\delta_b^j - \frac{1}{n}) + (\delta_a^j - \frac{1}{n})(\delta_b^i - \frac{1}{n})\Big) \\
&= -\frac{1}{2} \sum_{i<j\in\Omega} D_{i,j}\Big(\delta_a^i\delta_b^j - \frac{1}{n}\delta_a^i - \frac{1}{n}\delta_b^j + \frac{1}{n^2} + \delta_a^j\delta_b^i - \frac{1}{n}\delta_a^j - \frac{1}{n}\delta_b^i + \frac{1}{n^2}\Big) \\
&= -\frac{1}{2} \sum_{i<j\in\Omega} D_{i,j}\Big(\delta_a^i\delta_b^j + \delta_a^j\delta_b^i\Big) + \frac{1}{2} \sum_{i<j\in\Omega} D_{i,j}\frac{1}{n}\Big(\delta_a^i + \delta_a^j\Big) \\
&\quad + \frac{1}{2} \sum_{i<j\in\Omega} D_{i,j}\frac{1}{n}\Big(\delta_b^i + \delta_b^j\Big) - \frac{1}{2} \sum_{i<j\in\Omega} D_{i,j}(\frac{2}{n^2}) \\
&= -\frac{1}{2}\Big(\mathcal{P}_\Omega(\mathbf{D})_{a,b} - \mu\big(\mathcal{P}_\Omega(\mathbf{D})(:,a)\big) - \mu\big(\mathcal{P}_\Omega(\mathbf{D})(:,b)\big) + \mu\big(\mathcal{P}_\Omega(\mathbf{D})\big)\Big) \\
&= \big[-\frac{1}{2}\mathbf{J}\mathcal{P}_\Omega(\mathbf{D})\mathbf{J}\big]_{a,b},
\end{aligned}$$

which is as desired. In the above, we had to recognize that our sampling operator only picks $(i < j)$, but either $(a < b)$ or $(b < a)$ can occur, so we need to check both permutations to compute the column averages correctly. ■

Lemma B.1 *Let T be the map that takes Gram matrices to distance matrices, explicitly defined as $T(\mathbf{X}) = \mathbf{1} \cdot diag(\mathbf{X})^T + diag(\mathbf{X}) \cdot \mathbf{1}^T - 2\mathbf{X}$. Let mtxdiag$(\cdot)$ denote the function that maps a column vector in $\mathbb{R}^n$ into an $n \times n$ diagonal matrix, where the diagonal elements correspond to the entries of the column vector. Then $T^\star(\mathbf{X}) =$ mtxdiag$(\mathbf{X}\mathbf{1}))$ + mtxdiag$(\mathbf{X}^T\mathbf{1}) - 2\mathbf{X}$ and $\mathcal{R}_\Omega^\star\mathcal{R}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega(\mathbf{J}\mathcal{P}_\Omega(\mathbf{D})\mathbf{J})$ − mtxdiag$(\mathcal{P}_\Omega(\mathbf{J}\mathcal{P}_\Omega(\mathbf{D})\mathbf{J}\mathbf{1})$ where $\mathbf{D} = T(\mathbf{X})$.*

Proof To compute the adjoint, note that

$$\langle T(\mathbf{X}), \mathbf{Y}\rangle = \langle \mathbf{1}\text{diag}(\mathbf{X})^T, \mathbf{Y}\rangle + \langle \text{diag}(\mathbf{X})\mathbf{1}^T, Y\rangle + \langle \mathbf{X}, -2\mathbf{Y}\rangle.$$

Next, we consider the first two terms on the right hand side of the above equation.

$$\langle \mathbf{1}\text{diag}(\mathbf{X})^T, \mathbf{Y}\rangle = \sum_{i,j} X_{i,i}Y_{j,i} = \left\langle \mathbf{X}, \text{mtxdiag}(\mathbf{Y}^T\mathbf{1})\right\rangle.$$

$$\langle \text{diag}(\mathbf{X})\mathbf{1}^T, \mathbf{Y}\rangle = \sum_{i,j} X_{i,i}Y_{i,j} = \left\langle \mathbf{X}, \text{mtxdiag}(\mathbf{Y}\mathbf{1})\right\rangle.$$

Using this, we can see that

$$T^\star(\mathbf{Y}) = \text{mtxdiag}(\mathbf{Y}\mathbf{1}) + \text{mtxdiag}(\mathbf{Y}^T\mathbf{1}) - 2\mathbf{Y}. \tag{8}$$

If $\mathbf{Y}$ is symmetric, the first two terms are equal. We can use this to derive an explicit form of $\mathcal{R}_\Omega^\star(\mathbf{X})$ for a symmetric matrix $\mathbf{X}$ in terms of T. Notice that

$$\langle \mathcal{R}_\Omega(\mathbf{X}), \mathbf{Y}\rangle = \frac{-L}{2m}\langle \mathbf{J}\mathcal{P}_\Omega(T(\mathbf{X}))\mathbf{J}\,, \mathbf{Y}\rangle = \langle \frac{-L}{2m}T^\star[\mathcal{P}_\Omega(\mathbf{J}\mathbf{Y}\mathbf{J})]\,, \mathbf{X}\rangle.$$

It follows that $\mathcal{R}_\Omega^\star(\mathbf{Y}) = \frac{-L}{2m}T^\star[\mathcal{P}_\Omega(\mathbf{J}\mathbf{Y}\mathbf{J})]$. These results can be combined to better understand the explicit relationship between $\mathcal{R}_\Omega^\star\mathcal{R}_\Omega(\mathbf{X})$ and $\mathcal{P}_\Omega(\mathbf{D})$, which enables an efficient implementation. In particular, we have

$$\begin{aligned}\mathcal{R}_\Omega^\star\mathcal{R}_\Omega(\mathbf{X}) &= \frac{L^2}{2m^2}\mathcal{R}_\Omega^\star\left[\frac{-1}{2}\mathbf{J}\mathcal{P}_\Omega(\mathbf{D})\mathbf{J}\right] \\ &= \frac{L^2}{2m^2}T^\star\left(\mathcal{P}_\Omega\left(\mathbf{J}\left[\frac{-1}{2}\mathbf{J}\mathcal{P}_\Omega(\mathbf{D})\mathbf{J}\right]\mathbf{J}\right)\right).\end{aligned}$$

Using the fact that $\mathbf{J}^2 = \mathbf{J}$, we obtain

$$\begin{aligned}\mathcal{R}_\Omega^\star\mathcal{R}_\Omega(\mathbf{X}) &= \frac{L^2}{2m^2}T^\star\left(\mathcal{P}_\Omega\left(\frac{-1}{2}\mathbf{J}\mathcal{P}_\Omega(\mathbf{D})\mathbf{J}\right)\right) \\ &= \frac{L^2}{2m^2}\left(\mathcal{P}_\Omega\left(\mathbf{J}\mathcal{P}_\Omega(\mathbf{D})\mathbf{J}\right)\right) - \frac{L^2}{2m^2}\text{mtxdiag}\left(\mathcal{P}_\Omega\left(\mathbf{J}\mathcal{P}_\Omega(\mathbf{D})\mathbf{J}\right)\mathbf{1}\right).\end{aligned}$$

This concludes the proof. ■