Manifold Reconstruction and Denoising from Scattered Data in High Dimension via a Generalization of L_1-Median

Shira Faigenbaum-Golovin[1,*]   David Levin[1]

[1] School of Mathematical Sciences, Tel Aviv University, Israel

[*] Corresponding author, E-mail address: alecsan1@post.tau.ac.il

November 17, 2022

Abstract

In this paper, we present a method for denoising and reconstruction of low-dimensional manifold in high-dimensional space. We suggest a multidimensional extension of the Locally Optimal Projection algorithm which was introduced by Lipman et al. in 2007 for surface reconstruction in 3D. The method bypasses the curse of dimensionality and avoids the need for carrying out dimensional reduction. It is based on a non-convex optimization problem, which leverages a generalization of the outlier robust L1-median to higher dimensions while generating noise-free quasi-uniformly distributed points reconstructing the unknown low-dimensional manifold. We develop a new algorithm and prove that it converges to a local stationary solution with a bounded linear rate of convergence in case the starting point is close enough to the local minimum. In addition, we show that its approximation order is $O(h^2)$, where h is the representative distance between the given points. We demonstrate the effectiveness of our approach by considering different manifold topologies with various amounts of noise, including a case of a manifold of different co-dimensions at different locations.

keywords: Manifold learning, Manifold denoising, Manifold reconstruction, High dimensions, Dimensional reduction

MSC classification: 65D99
(Numerical analysis - Numerical approximation and computational geometry)

1   Introduction

High-dimensional data is increasingly available in many fields, and the problem of extracting valuable information from such data is of primal interest. Often, the data suffers from the presence of noise, outliers, and non-uniform sampling, which can influence the result of the mining task. We can address this problem by denoising a single sample, an approach extensively used in the last decades (the denoising method is often data-driven). However, it is still a challenge to produce a good noise-free result from a single sample with a large amount of noise present. Frequently, classical denoising algorithms lose the battle, since they denoise a single sample and overlook the intrinsic connections between different samples acquired from a

chosen domain. As a result, obtaining a dataset of samples with certain properties can boost the denoising process. A common practice is to assume that the high-dimensional input data lies on an intrinsically low-dimensional Riemannian manifold.

For instance, with the development of image processing, the task of image denoising gained a lot of attention (see, e.g., [15, 32, 39]). Thus, given a single image, the task is to find its noise-free image. Now, let us consider a collection of noisy images depicting a single object, controlled by several parameters (such as a set of faces or written letters rotated in different directions). This collection can be modeled by a manifold, and this representation can be utilized to produce a superior denoising result. A real-life case, which motivated the current research, is cryo-electron microscopy [37]. In this problem a single image is a projection of a three-dimensional macromolecule into a two-dimensional representation (Figure 1 (A)). Cryo-electron microscopy images are known to suffer from extremely low signal to noise ratio (Figure 1 (C)), and consequently classical denoising methods usually do not perform well on such samples. Nevertheless, using the fact that the images are sampled from a manifold (each corresponding to the molecule projected in a different direction) can facilitate the denoising task. Figure 1 (B) shows a collection of images, each depicting a projection of the simulated molecule in Figure 1 (A), captured in various directions. Thus, we transfer the problem from single image denoising to denoising the entire image set – which is treated as scattered data sampled from a manifold.
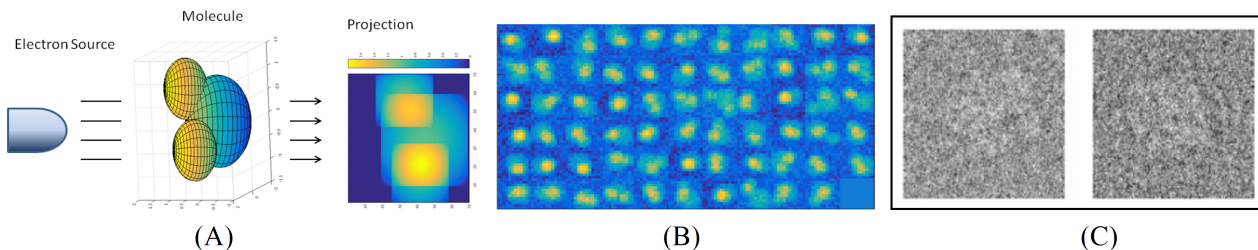

(A) (B) (C)

Figure 1: (A) Illustration of the cryo-electron microscopy projection process, where a 3D molecule is rotated and projected to 2D. (B) Collection of the artificial projections of the molecule with noise, where each image is the molecule rotated in a different direction. (C) Two real electron microscope images of the E. coli 50S ribosomal subunit (image is taken from [37]). These images demonstrate the denoising challenge of extremely low signal to noise ratio.

In this paper, we address the problem of manifold denoising and reconstruction. Let $\mathcal{M}$ be a d-dimensional manifold in $\mathbb{R}^n$, where $d \ll n$. Suppose that the scattered data $P = \{p_j\}_{j=1}^J$ were sampled near $\mathcal{M}$ and contain noise and outliers. We wish to find a noise-free reconstruction of the geometry of $\mathcal{M}$ in $\mathbb{R}^n$.

Before we turn to high-dimensional data, we first consider the simpler, yet challenging problem of surface reconstruction. While the problem of low-dimensional reconstruction was thoroughly studied along the years [3,6,11,26,30], there are still many challenges which modern applications (e.g., computer graphics) pose. One of them is surface reconstruction with preservation of features [22,44]. The available methods commonly assume almost noise-free data and rely on normal estimation. Unfortunately, in real-life cases, noise is often present, and normal estimation may not be robust enough (despite various processes for cleaning the normals). Let us mention here the Parameterization-free Projection method for geometry reconstruction proposed in [30], which offers a solution that can handle high levels of noise. This method does not require a well-defined surface parameterization, avoids using local surface approximation

and normal estimation, is cheap, and can be parallelized due to its local support. In [30], it was demonstrated by various examples that the method is stable with respect to outliers, different density of sampling and varying topology.

In the high-dimensional case, the problem of manifold reconstruction still requires additional attention. The era of proliferation of high-dimensional data raised the need for efficient denoising and reconstruction algorithms for manifolds. The application of classical approximation tools, developed for surfaces, to high-dimensional data, encounters various challenges, usually stemming from the high-dimension, and presence of noise. For instance, given a uniform sampling in $\mathbb{R}^n$ on a grid with spacing $h = 1/L$ requires L^n samples and when $L \ll 10$ this is already challenging for $n \ll 10$. Moreover, classical approximation methods assume smoothness of order s, which is closely related to the approximation error. For example, for J sample points, the reconstruction accuracy can be of the order of $O(J^{-s/n})$, which implies that we need to increase the amount of data as the domain dimension increases [4]. As a result, in the high-dimensional case, the problem of manifold reconstruction still requires additional attention especially to the problem of denoising and reconstructing manifold.

A common way of dealing with high-dimensional data is to use dimensionality reduction. The motivation often stems from the need to analyze, process, and visualize high-dimensional data. Along the years many dimensionality reduction techniques were developed (PCA [35], Multidimensional Scaling [13], Linear Discriminant Analysis [19], Locality Preserving Projections [20], Locally Linear Embedding [36], ISOMAP [41], Diffusion Maps [12], and Neural Networks in their general form, [28], to mention just a few). However, one has to be careful when performing dimensionality reduction, since meaningful information can be lost due to the assumptions made. One fundamental challenge of dimensionality reduction is knowing or estimating the dimension of the data. In addition, since the geometry of the data is usually unknown, it is common to use an assumption regarding its geometrical structure (and use linear/non-linear algorithms accordingly). As a result, in the case of real-life data, it is still a challenge to address these issues, mainly because such assumptions have a direct influence on the usage of dimensionality reduction methods, and may, therefore, hamper the results of the analysis performed. For a comprehensive survey of manifold learning methods that rely on dimensionality reduction, see [28].

An alternative practice for handling high-dimensional data is manifold learning in high-dimensional space. Thus, instead of making assumptions on the geometry of the manifold, its intrinsic dimension and reducing the dimension of the data, the mining task is performed in a high-dimensional space. This approach has several advantages, as well as disadvantages. On the one hand, there is no loss of information. On the other hand, the dimension of the data influences the efficiency and feasibility of the algorithms, and it is possible that one will not be able to see the forest for the tree. An additional important factor of high-dimensional data is noise, which is usually present in real-life scenarios. In Table 1 we give a short survey of manifold reconstruction methods that avoid performing dimensionality reduction. Among the first papers that addressed the manifold reconstruction problem was [10]. The method presented therein relied on Delaunay triangulation, and as the authors themselves noted, it was impractical, mainly because it requires a very dense and noise-free sample, and also because it makes use of (weighted) Delaunay triangulation in higher dimensions. Next, in [34] it was proposed to use simplicial complexes. In that paper, the authors also address the challenge of noisy samples, under certain conditions. This work was followed by [8], which aimed at avoiding computing the Delaunay triangulation of the given set of points by using a Witness complex via an iterative process, and by [7] which addressed the problem using a Tangential Delaunay complex. Unfortunately, this method dealt only with noise-free samples. Next in [31],

the authors proposed to learn a data-dependent dictionary from clean data in the chosen resolution level and use it for the manifold reconstruction of possibly noisy data. Later, in [18], it was suggested to use a covering of the manifold by discs to deal with a small amount of Gaussian noise. The recent paper [38] proposed to address manifold denoising under various noisy scenarios, when the intrinsic dimension of the manifold is known, by extending the Moving Least Squares method [26] to the high-dimensional case. Finally, the paper [1] address the problem manifold reconstruction and of tangent space and curvature estimation by using local polynomials.

Table 1: Survey of manifold reconstruction methods, that avoid performing dimensionality reduction

Authors	Algorithm key features	Assumptions on the data	Sampling	Handle noise	Error	Numerical exp.	Complexity, N is #points, d is ID, n is the dim. of the ambient space
Cheng, et al. [10]	Weighted Delaunay triangulation	Compact manifold, smooth, no boundary	Sufficiently dense point sample	Noise-free sample	homeo-morphic	N/A	$O(N \log(N))$
Niyogi, et al. [34]	Simplicial complex	Suffcient amount of points		Bounded/ specific models of noise	homeo-morphic	N/A	N/A
Boissonnat, et al. [8]	Witness complex	Positive reach (i.e. C^1-continuous)	Not necessarily uniformly sampled, minimal local density	Low noise level	homeo-morphic	N/A	$N^2 d^{O(d^2)}$
Chazal, et al. [9]	Distance functions with probability distribution		Regularity of the input data	Bounded/ specific models of noise	homotopic ✓		N/A
Boissonnat, et al. [7]	Tangential Delaunay complex	Smooth manifold, positive reach	Sampling ratio, point sparsity, and the reach hold a condition	Noise-free sample	homeo-morphic	N/A	$O(n)N^2 + n2^{O(d^2)}N$
Maggioni, et al. [31]	Dictionary	Smooth closed manifold, d is known	Homogeneous, reconstruct new noisy samples	Additive noise, dictionary is built from clean samples	✓	✓	$O(C^d(n + d^2)\epsilon^{-(1-\frac{d}{2})}log\frac{1}{\epsilon} + dn)$, where C is a constant, and ϵ is reconstruction error
Fefferman, et al. [18]	Disk stitching	Reach is bounded		Additive noise	✓	N/A	N/A
Sober, Levin [38]	Moving Least Squares	d is known, bounded reach		Additive noise	✓	✓	$O(d^3m + Nd^m + NI)$, I-#points in supp., m is the approx. degree
Aamari, Levrard [1]	Local Polynomials	d and order of regularity are known		Bounded/ specific models of noise	✓	N/A	N/A

The methods listed in the table provide a strong theoretical background, but most of them are not accompanied by numerical examples (except [9, 31, 38]), which is an important aspect of evaluating the method execution. In addition, unfortunately, as can be seen from the table, handling noisy data, non-uniformly sampled, with no assumption on the data, is still a challenge in high-dimensional cases. In this paper, we propose denoising and reconstructing the manifold geometry in a high-dimensional space in the presence of high amounts of noise and outliers. We will tackle the manifold approximation question by extending the Locally Optimal Projection algorithm [30] to the high-dimensional case. The proposed algorithm is simple, fast and efficient, and does not require any additional assumption. Our theoretical analysis is accompanied by numerical examples of various manifolds with different amounts of noise.

2 High-Dimensional Denoising and Reconstruction

The Locally Optimal Projection (LOP) method was introduced in [30] to approximate two-dimensional surfaces in $\mathbb{R}^3$ from point set data.The procedure does not require the estimation of local normals and planes, or parametric representations. In addition, the method performs well in the case of noisy samples. Due to its flexibility and satisfactory results, it has been extended to address other challenges related to surfaces [21, 22, 40].

Herein we generalize the LOP mechanism to perform what we call *Manifold Locally Optimal Projection (MLOP)*. The vanilla LOP is not able to cope with high-dimensional data, mainly due to the sensitivity of the norm to noise and outliers (as will be discussed in details in subsection 3.1). In addition, other adaptations are required due to practical reasons (as will be described in the end of this section).

First, we adapt the h-ρ condition defined for scattered-data approximation functions (in [25], defined for low-dimensional data), to handle finite discrete data on manifolds.

Definition 1. h-ρ *sets of fill-distance* h*, and density* $\leq \rho$ *with respect to the manifold* $\mathcal{M}$*. Let* $\mathcal{M}$ *be a d-dimensional manifold in $\mathbb{R}^n$ and consider a set of data points* $P = \{p_j\}_{i=1}^{J}$ *sampled from* $\mathcal{M}$*. We say that P is an h-ρ set if:*
1. h_0 is the fill-distance, i.e., $h_0 = \max_{y \in M} \min_{p_j \in P} \|y - p_j\|$.
2. The density of the points can be bounded as $\#\{P \cap \bar{B}(y, kh_0)\} \leq \rho k^d, \quad k \geq 1, \quad y \in \mathcal{M}$*.*
Here $\#Y$ denotes the number of elements in a set Y and $\bar{B}(x, r)$ denotes the closed ball of radius r centered at x.

Note that the last condition regarding the point separation δ defined in [25], which states that there $\exists \delta > 0$ such that $\|p_i - p_j\| \geq \delta, \quad 1 \leq i \leq j \leq J$, is redundant in the case of finite data.

The setting for the high-dimensional reconstruction problem is the following: Let $\mathcal{M}$ be a manifold in $\mathbb{R}^n$, of unknown intrinsic dimension $d \ll n$. One is given a noisy point-cloud $P = \{p_j\}_{j=1}^{J} \subset \mathbb{R}^n$ situated near the manifold $\mathcal{M}$, such that P is a h-ρ set. We wish to find a new point-set $Q = \{q_i\}_{i=1}^{I} \subset \mathbb{R}^n$ which will serve as a noise-free approximation of $\mathcal{M}$. We seek a solution in the form of a new point-set Q, which will replace the given data P, provide a noise-free approximation of $\mathcal{M}$, and which is quasi-uniformly distributed. This is achieved by leveraging the well-studied weighted L_1-median [42] used in the LOP algorithm and requiring a quasi-uniform distribution of points $q_i \in Q$. These ideas are encoded by the cost function

$$G(Q) = E_1(P, Q) + \Lambda E_2(Q) = \sum_{q_i \in Q} \sum_{pj \in P} \|q_i - p_j\|_{H_\epsilon} w_{i,j} + \sum_{q_i \in Q} \lambda_i \sum_{q_{i'} \in Q \setminus \{q_i\}} \eta(\|q_i - q_i'\|) \hat{w}_{i,i'}, \quad (1)$$

where the weights $w_{i,j}$ are given by rapidly decreasing smooth functions. In our implementation we used $w_{i,j} = \exp\left\{-\|q_i - p_j\|^2/h_1^2\right\}$ and $\widehat{w}_{i,i'} = \exp\left\{-\|q_i - q_i'\|^2/h_2^2\right\}$. Here, we replace the L_1-norm used in [30] by the "norm" $\|\cdot\|_{H_\epsilon}$ introduced in [27] as $\|v\|_{H_\epsilon} = \sqrt{v^2 + \epsilon}$, where $\epsilon > 0$ is a fixed parameter (in our case we take $\epsilon = 0.1$). As shown in [27], using $\|\cdot\|_{H_\epsilon}$ instead of $\|\cdot\|_1$ has the advantage that one works with a smooth cost function and outliers can be removed. In addition, h_1 and h_2 are the support size parameters of $w_{i,j}$ and $\hat{w}_{i,i'}$ that guarantee a sufficient amount of P or Q points for the reconstruction. We provide additional details on how to estimate the support size, in Subsection 3.2. Also, $\eta(r)$ is a decreasing function such that $\eta(0) = \infty$; in our case we take $\eta(r) = \frac{1}{3r^3}$. Finally, $\{\lambda_i\}_{i=1}^I$ are constant balancing parameters.

We will now give some intuition about the definition of the cost function G. We can describe the cost function in (1) in terms borrowed from electromagnetism, where an electron generates an electric field that exerts an attractive force on a particle with a positive charge, such as the proton, and a repulsive force on a particle with a negative charge. In our scenario, we have attraction forces between the Q-points and the original P-points, and repulsion forces between the Q-points to themselves in order to make them spread out in a quasi-uniform manner (Figure 2). An additional way of looking at the target function is to view the solution using a service center approach: placing a distribution of service centers $q_i \in Q$ to best serve the customers P, such that the service centers are spread uniformly. Thus, in case we have more points in P than in the reconstruction, each center $q_i \in Q$ will serve a certain amount of P-points in its neighborhood.

Remark 2.1. We do not require that the amount of the points in the reconstruction (Q), and the size of the original sample set (P) be the same. This flexibility allows downsampling and upsampling in order to decode or encode manifold information.
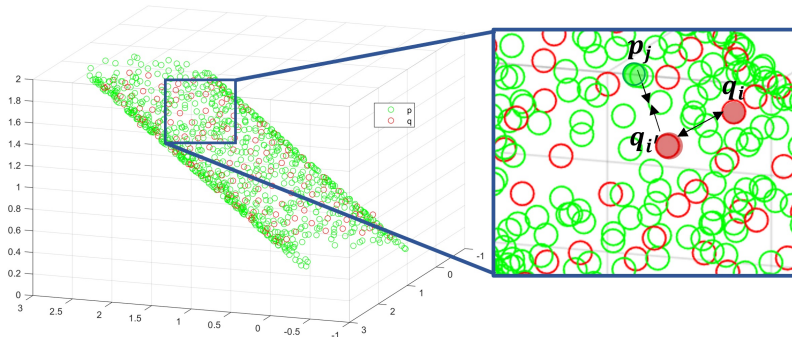

Figure 2: Illustration of the cost function during manifold reconstruction: each point from the reconstruction set Q (red points) is attracted to points in P (green dots), and repelled by other points in Q according to their distance.

In order to solve the problem with the cost function (1), we look for a point-set Q that minimizes $G(Q)$. The solution Q is found via the gradient descent iterations

$$q_{i'}^{(k+1)} = q_{i'}^{(k)} - \gamma_k \nabla G(q_{i'}^{(k)}), \qquad i' = 1, \ldots, I, \tag{2}$$

where the initial guess $\{q_i^{(0)}\}_{i=1}^I = Q^{(0)}$ consists of points are sampled from P. The gradient of G is given by

$$\nabla G(q_{i'}^{(k)}) = \sum_{j=1}^J \left(q_{i'}^{(k)} - p_j\right)\alpha_j^{i'} - \lambda_{i'} \sum_{\substack{i=1 \\ i \neq i'}}^I \left(q_{i'}^{(k)} - q_i^{(k)}\right)\beta_i^{i'}, \tag{3}$$

6

with the coefficients $\alpha_j^{i'}$ and $\beta_j^{i'}$ given by the formulas

$$\alpha_j^{i'} = \frac{w_{i,j}}{\|q_i - p_j\|_{H_\epsilon}} \left(1 - \frac{2}{h_1^2}\|q_i - p_j\|_{H_\epsilon}^2\right) \tag{4}$$

and

$$\beta_i^{i'} = \frac{\widehat{w}_{i,i'}}{\|q_i - q_{i'}\|} \left(\left|\frac{\partial \eta \left(\|q_i - q_{i'}\|\right)}{\partial r}\right| + \frac{2\eta \left(\|q_i - q_{i'}\|\right)}{h_2^2}\|q_i - q_{i'}\|\right), \tag{5}$$

for $i = 1, ..., I$, $i \neq i'$. In order to balance the two terms in $\nabla G(q_{i'}^{(k)})$, the factors $\lambda_{i'}$ are initialized in the first iteration as

$$\lambda_{i'} = -\frac{\left\|\sum_{j=1}^{J} \left(q_{i'}^{(k)} - p_j\right)\alpha_j^{i'}\right\|}{\left\|\sum_{i=1}^{I} \left(q_{i'}^{(k)} - q_i^{(k)}\right)\beta_i^{i'}\right\|}. \tag{6}$$

Balancing the contribution of the two terms is important in order to maintain equal influence of the attraction and repulsion forces in $G(Q)$. The step size in the direction of the gradient γ_k is calculated following the procedure suggested by Barzilai and Borwein in [5], as

$$\gamma_k = \frac{\langle \triangle q_{i'}^{(k)}, \triangle G_{i'}^{(k)} \rangle}{\langle \triangle G_{i'}^{(k)}, \triangle G_{i'}^{(k)} \rangle}, \tag{7}$$

where $\triangle q_{i'}^{(k)} = q_{i'}^{(k)} - q_{i'}^{(k-1)}$ and $\triangle G_{i'}^{(k)} = \nabla G_{i'}^{(k)} - \nabla G_{i'}^{(k-1)}$.

The reconstruction process is summarized in Algorithm 1 below:

Algorithm 1 MLOP: Iterative Manifold Reconstruction

1: **Input:** $P = \{p_j\}_{j=1}^{J} \subset \mathbb{R}^n$, $\epsilon > 0$
2: **Output:** $Q = \{q_i\}_{i=1}^{I} \subset \mathbb{R}^n$
3: Initialize $Q^{(0)}$ as a subsample of P
4: Estimate h_1 and h_2
5: **repeat**
6:     **for** each $q_{i'}^{(k)} \in Q^{(k)}$ **do**
7:         Calculate $\nabla G(q_{i'}^{(k)})$ by assessing $\alpha_j^{i'}$, $\beta_i^{i'}$
8:         $q_{i'}^{(k+1)} = q_{i'}^{(k)} - \gamma_k \nabla G(q_{i'}^{(k)})$
9:     **end for**
10: **until** $\|\nabla G(q_{i'}^{(k)})\| < \epsilon$

Naturally, several changes were made to the LOP algorithm when shifting from the low-dimension to high-dimensional case. The **main enhancements of the LOP algorithm which were introduced in MLOP for high-dimensional space** can be summarized in the following list:

1. The problem is reformulated in terms of looking for a new set Q which will maintain the conditions in (1). This change is taken into account when taking the derivatives.

2. The L_1 norm used in E_1 is replaced with the H_ϵ, defined in [27] as $\|v\|_{H_\epsilon} = \sqrt{v^2 + \epsilon}$, where $\epsilon > 0$ is a fixed parameter. The motivation behind this is to have a "norm" which is less sensitive to outliers. Instead of squares of errors or the absolute values of the errors, we will use an error measure that behaves as squared error for small errors and as an absolute error if the error is large. Please note that we change the norm only in the first term in (1) to cope with the outliers in P.

3. The norm calculation is modified to cope with high-dimensional data with noise, by using the sketching technique. For more details see Section (3.1).

4. From practical reasons, we replace the fixed point iterations used in [30], with a gradient descent. The motivation behind it was to use a methodology that will allow easier theoretical analysis of the already challenging non-convex function G.

5. A new definition for the balancing terms λ_i is suggested, such that the λ_i does not change along the iterations (and there is no need to take the their derivatives).

6. Different support sizes are used when looking at the support of a given point q_i with respect to P and with respect to Q. This is natural when the number of points in P and Q differ. In addition, we propose a procedure for estimating these parameters (see Section (3.2)).

3 Practical Details

In Section 2 we introduced the method for high-dimensional denoising and reconstruction, by optimizing a cost function that leverages the proximity to the original data and asks for quasi-uniform reconstruction. In the following two subs-sections, we will discuss several practical aspects related to robust high-dimensional distance calculation, as well as the optimal selection of the support of the weight function $w_{i,j}$.

3.1 Robust Distance Calculation in High Dimensions

The reasoning in terms of Euclidean distances, which is the cornerstone of Algorithm 1, works well in low dimensions, e.g., for the reconstruction of surfaces in 3D, but breaks down in high dimensions once noise is present. For example, consider three points A, B and C in $\mathbb{R}^2$ (Figure 3 (A)), where the points A and B are close, whereas the point C is far. Next, we embed these points in to $\mathbb{R}^{60}$ with a uniformly additive noise distribution $U(-0.2, 0.2)$ (for example in Figure 3 (D) we plot one of the points in $\mathbb{R}^{60}$). Unfortunately, the noise completely wipes out the signal and as a result far points cannot be distinguished from adjacent ones, see Figure 3 (B) (see [2, 14]).

To deal with this issue, we perform dimension reduction via random linear sketching [43]. It should be emphasized that the dimension reduction procedure is utilized solely for the calculation of norms, and the manifold reconstruction is performed in the high-dimensional space. Given a point $x \in \mathbb{R}^n$, we project it to a lower dimension $m \ll n$ using a random matrix, S, with certain properties (its construction is described in detail in Algorithm 2). Subsequently, the norm of $\|S^t x\|$ will approximate $\|x\|$. Figure 3 (C) shows that calculating the distance in lower-dimensional space solves the distance conflicts.

In Algorithm 2 we present the details of finding the matrix $S \in \mathbb{R}^{n \times m}$. For given scattered data points $P = \{p_j\}_{j=1}^J \subset \mathbb{R}^n$ we construct matrix S only once during the initialization process of Algorithm 1. Next, given a new point $x \in \mathbb{R}^n$, its norm is approximated as $\|S^t x\|$ and utilized only for the gradient calculations in (3). In this paper, we choose to perform a global linear projection. However, for additional accuracy, it is possible to find a local transformation for each neighborhood.

Remark 3.1. How should we choose the dimension m of the space on which we project the data? First, if the dimension of the manifold $\mathcal{M}$ is known, this information can be utilized for setting m. Alternatively, one can calculate a rough estimate, or apply a local PCA, and use the number of the dominant eigenvalues. In our examples, the typical size of m was set to 10.

Algorithm 2 Robust Distance Calculation in High Dimensions

1: **Input:** $P = \{p_j\}_{j=1}^J \subset \mathbb{R}^n$, m
2: **Output:** S - an $n \times m$ matrix
3: Sample $G \in \mathbb{R}^{J \times m}$ with $G \sim N(0,1)$.
4: Compute $B \in \mathbb{R}^{n \times m}$ as $B := P^t G$.
5: Calculate the QR decomposition of B as $B = SR$, where $S \in \mathbb{R}^{n \times m}$ has orthonormal columns and $R \in \mathbb{R}^{m \times m}$ is upper triangular.
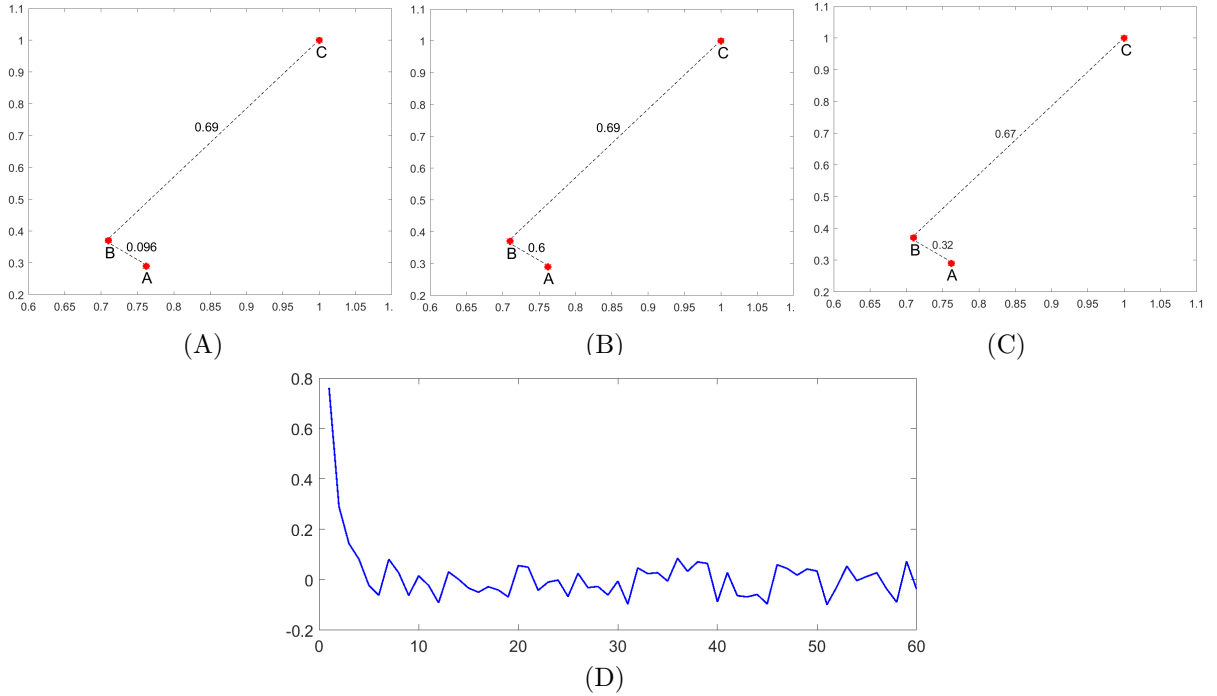
Figure 3: Calculating distances in low-and high-dimensional space. (A) Distance calculation of points in $\mathbb{R}^2$. (B) Distance calculation of points in $\mathbb{R}^2$ embedded into $\mathbb{R}^{60}$ + noise $U(-0.2; 0.2)$; (C) Distance calculation of points in $\mathbb{R}^2$ embedded into $\mathbb{R}^{60}$ + noise: after sketching, (D) Point A embedded into $\mathbb{R}^{60}$ + noise.

3.2 Optimal Neighborhood Selection

In this subsection we consider the support size of the locally supported weight functions $w_{i,j}$ utilized in (1) for manifold reconstruction. Specifically, given a point-set $X = \{x_k\}_{k=1}^K$, we address the problem of choosing a support size h that will guarantee a sufficient amount of points from X in the neighborhood of a point q_i during the MLOP approximation. Although, the LOP technique has gained much popularity, and many extensions were suggested. However, the proper choice of neighboring points to be used in the reconstruction still remains an important open problem. From the one side, taking points far from the tested point can be influenced by the changing geometry of the manifold, from the other side if the neighborhood size is too

small we can lose the robustness to noise property. As a result, support size selection is a critical point when dealing with a fast decaying weight function, and it is important to find an estimate to it (e.g., see the analysis for the MLS case in [29]).

There is a high degree of freedom in choosing the points participating in the approximation since the number of data points is usually very large. Naturally, one would like to make use of these large degrees of freedom to achieve the "best" reconstruction. In what follows, we use the service centers considerations in order to approximate h as a radius of the ball containing the K-nearest neighbors. It should be noted that naturally, we look for two parameters h_1, and h_2, defined as the support sizes of q_i with respect to P and Q, respectively. The reason for having different supports is due to the fact that the number of points in P and Q can differ, and this should be reflected in the choice of their support size. As will be demonstrated in the numerical examples section, our approach outperforms the heuristic choice of support size in approximation quality and stability.

The support sizes h_1, and h_2 are closely related to the fill-distance of the P points and the Q points. Let J and I be the sizes of the sets P, and Q respectively. In case $I \leq J$, each q_i can be viewed as a service center that serves approximately $\nu = \lfloor \frac{J}{I} \rfloor$ points from the p_j's. We use this observation to calculate the fill-distance of P, then estimate the support that guarantees at least ν points in the neighborhood of p_j, as well as the practical support size of the Gaussian $w_{i,j}$ (see the illustration in Figure 4).

Unlike the standard definition of fill-distance in scattered data function approximation [25], we introduce

Definition 2. *The fill-distance of the set P is*

$$h_0 = \max_{y \in M} \min_{p_j \in P} \|y - p_j\|.. \tag{8}$$

Definition 3. *Given two point-clouds $P = \{p_j\}_{j=1}^J \subset \mathbb{R}^n$ and $Q = \{q_i\}_{i=1}^I \subset \mathbb{R}^n$, situated near a manifold $\mathcal{M}$ in $\mathbb{R}^n$, such that their sizes obey the constraint $I \leq J$, denote $\nu = \lfloor \frac{J}{I} \rfloor$. Then we say that the radius that guarantees approximately ν points from P in the support of each point q_i is $\hat{h}_0 = c_1 h_0$, with c_1 given by*

$$c_1 = argmin\{c : \#(\bar{B}_{ch_0}(q_i) \cap P) \geq \nu, \forall q_i \in Q\}. \tag{9}$$

where $\#(B_r(x) \cap P)$ is the number of points in a ball $B_r(x)$ of radius r centered at the point x.

Remark 3.2. Let σ be the variance of a Gaussian $w(r) = e^{-\frac{r^2}{\sigma^2}}$. For the normal distribution, four standard deviations away from the mean account for 99.99% of the set. In our case, by the definition of $w_{i,k}$, since h is the square root of the variance, $4\sigma = 4\frac{h}{\sqrt{2}} = 2\sqrt{2}h_1$ covers 99.99% of the support size of $w_{i,k}$.

The following theorem indicates how the parameters h_1 and h_2 should be selected.

Theorem 3.3. *Let $\mathcal{M}$ be a d-dimensional manifold in $\mathbb{R}^n$. Suppose given two point-clouds $P = \{p_j\}_{j=1}^J \subset \mathbb{R}^n$ and $Q = \{q_i\}_{i=1}^I \subset \mathbb{R}^n$ situated near a manifold $\mathcal{M}$ in $\mathbb{R}^n$, such that their sizes obey the constraint $I \leq J$, and let $\nu = \lfloor \frac{J}{I} \rfloor$. Let $w_{i,j}$ be the locally supported weight function given by $w_{i,j} = \exp\{-\|q_i - p_j\|^2/h^2\}$. Then a neighborhood size of $h = 2\sqrt{2}\hat{h}_0$ guarantees $2^{1.5d}\nu$ points in the support of $w_{i,j}$, where $\widehat{h}_0 = c_1 h_0$, with c_1 given by (9).*

Proof. Given a point q_i we look for the amount of points from P in the support of $w_{i,j}$. Using Remark 3.2 we can estimate the support size of $w_{i,j}$ as 4σ, where $4\sigma = 2\sqrt{2}h_1$. We denote the amount of points from P in the support of q_i by $S_{4\sigma}$. In what follows we assume that the proportion of the number of points in a support does not change with radius changes. Thus, $S_{4\sigma}$ can be determined from the ratio of the volume to the amount of served points: $\frac{V_1}{V_2} = \frac{S_\sigma}{S_{4\sigma}}$, where the volume of a ball with radius $\hat{h}_0$ in $\mathbb{R}^d$ is $V_1 = \pi^{d/2}\hat{h}_0^d/c(d)$, and the volume of a ball with radius 4σ is $V_2 = \pi^{d/2}(4\sigma)^d/c(d) = 2^{1.5d}\pi^{d/2}\hat{h}_0^d/c(d)$ (where c is Euler's gamma function). Thus, $S_{4\sigma} = \nu\frac{V_2}{V_1} = 2^{1.5d}\nu$. $\qquad\square$
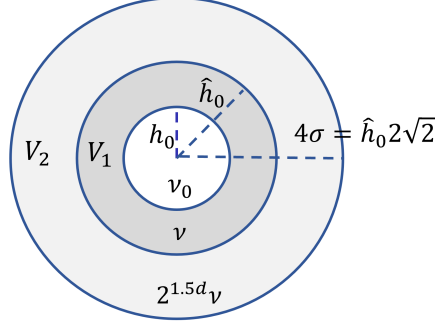

Figure 4: Scheme of the fill-distance and the size of the support of the weight function. h_0 is the radius that guarantees at least one point p_j in the support of q_i, $\hat{h}_0$ guarantees ν points, while the real number of points in the support is $2^{1.5d}\nu$.

Corollary 3.4. *Let P and Q be as defined in Theorem 3.3, and assume $J < I$, then the number of Q points in the support of each $p_j \in P$ is $2^{1.5d}\nu$.*

Proof. Each p_j can be viewed as a service center that serves approximately $\nu = I/J$ points q_i from Q. All the preceding definitions remain valid, except that the roles of P and Q are switched. Namely, h_0 is the fill-distance of the set P within the set Q, $\hat{h}_0$ guarantees ν points from Q near each point from P, and the actual number of Q points in the support of P is $2^{1.5d}\nu$. $\qquad\square$

Remark 3.5. **Practical considerations for the support size calculations.** As mentioned above, given a point q_i we estimate two different support sizes h_1 and h_2 with respect to the sets P and Q to be used in 1. Assume $I < J$, then h_1 is set to be $\hat{h}_0$, which is calculated using definition 3. Since we don't have any knowledge about the uniformity of distribution of the Q points over $\mathcal{M}$, we estimate h_2 as follows. We sample I points uniformly from P, and denote this set by Q^{rand}. Next, we estimate h_2 as $\hat{h}_0$ using definition 3, when substituting both of the sets P and Q to be Q^{rand}. This gives a rough estimation of h_2 in the scenario when the Q points are equality distributed over $\mathcal{M}$.

Remark 3.6. The reach τ_M of $\mathcal{M} \subset \mathbb{R}^n$ is defined as the largest number such that any point at distance less than τ_M from $\mathcal{M}$ has a unique nearest point on $\mathcal{M}$ [17]. We note that h should be smaller than the reach τ_M of the manifold $\mathcal{M}$. The reason for this is to prevent a situation where the weighted summations used in the cost function (1) may be influenced by points in another branch of $\mathcal{M}$ if this constraint is violated.

4 Main Results

Although LOP became popular for surface reconstruction, very important theoretical aspects of the methodology didn't gain attention. The main goal of the analysis presented in this

section is to complete the missing parts of the puzzle for the high-dimensional case. We will prove the convergence of the MLOP method, order of approximation, convergence rate as well as its complexity (presented in Theorem 4.3, Theorem 4.4 and Theorem 4.6, respectively). In addition, we will discuss the uniqueness of the MLOP solution (see Subsection 4.4).

4.1 Convergence to a Stationary Point

We are now ready to state our main convergence theorem. The fact that the cost function is non-convex poses a challenge for the proof of the convergence of the proposed method. First, we define h as described in Section 3.2 and assume that the h-ρ condition, defined above, is satisfied. Next, we utilize the following general non-convex convergence theorem presented in [24] to prove the convergence of our method.

Theorem 4.1. *Let $f : \mathbb{R}^d \to \mathbb{R}$, not necessarily convex, be twice continuously differentiable and has Lipschitz gradient, with constant L, i.e., $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$. Let its the gradient descent of f be $x^k = x^{k-1} - \alpha \nabla f(x^{k-1})$, with bounded step size $0 < \alpha < 1/L$. Suppose, all saddle points of the function f are strict-saddle (i.e., for all critical points x^* of f, $\lambda_{\min} \nabla^2(f(x^*)) < 0$). Then the gradient descent with random initialization and sufficiently small constant step size converges almost surely to a local minimizer or to minus infinity. i.e., if x^* is a strict saddle then $\Pr(\lim x_k = x^*) = 0$.*

We also recall the following theorem on eigenvalue bounds, due to Iyengar et al. [23].

Theorem 4.2. *The highest and lowest eigenvalues of a self-adjoint matrix X, with entries $x_{i,j}$, lie in the range*

$$\lambda_{\min}, \lambda_{\max} \in [l, u],$$

where

$$l = \min_{i \in I} \left(x_{i,i} - \sum_j |x_{i,j}| \right) \text{ and } u = \max_{i \in I} \left(x_{i,i} + \sum_j |x_{i,j}| \right).$$

Theorem 4.3 (Convergence to a stationary point). *Let $\mathcal{M}$ be a d-dimensional manifold in $\mathbb{R}^n$, where d is an unknown intrinsic dimension. Suppose that the scattered data points $P = \{p_j\}_{j=1}^J$ were sampled near the manifold $\mathcal{M}$, h_1 and h_2 are set as defined in Section 3.2, and the h-ρ set condition is satisfied with respect to $\mathcal{M}$. Let the points $Q^{(0)} = \{q_i^{(0)}\}_{i=1}^I$ be sampled from P. Then the gradient descent iterations (1) converge almost surely to a local minimizer Q^*.*

Proof. We proceed by verifying that the conditions of Theorem 4.1 hold. At a high level, our proof consists of the following steps:

1. Calculate the Hessian of the cost function (1).
2. Bound the eigenvalues of the Hessian.
3. Show that the minimal eigenvalue is negative.
4. Bound the norm of the Hessian.

We rephrase the minimization problem from (1) by writing E_1 and E_2 in a matrix form as

$$E_1 = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}^t \begin{pmatrix} \|q_1 - p_1\|w_{1,1} & \dots & \|q_1 - p_J\|w_{1,J} \\ \vdots & \vdots & \vdots \\ \|q_I - p_1\|w_{I,1} & \dots & \|q_I - p_J\|w_{I,J} \end{pmatrix} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix},$$

$$E_2 = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_I \end{pmatrix}^t \begin{pmatrix} 0 & \eta(\|q_1 - q_2\|)w_{1,2} & \dots & \eta(\|q_1 - q_I\|)w_{1,I} \\ \eta(\|q_2 - q_1\|)w_{2,1} & 0 & \dots & \eta(\|q_2 - q_I\|)w_{2,I} \\ \vdots & \vdots & \vdots & \vdots \\ \eta(\|q_I - q_1\|)w_{I,1} & \dots & \eta(\|q_I - q_{I-1}\|)w_{I,I-1} & 0 \end{pmatrix} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}.$$

The cost function is rewritten as

$$G(Q) = \vec{1}^t \Phi \vec{1} + \vec{\Lambda}^t \Psi \vec{1},$$

where $\phi_{i,j} = \|q_i - p_j\|w_{i,j}$ are the entries of Φ, $\psi_{i,j} = \eta(\|q_i - q_{i'}\|)\hat{w}_{i,i'}$ are the entries of Ψ, and the vector of balancing parameters $\vec{\Lambda} = (\lambda_1, \dots, \lambda_I)$ is defined in (6).

The proof relies on the fact that the weights $w_{i,j}$ are defined by rapidly decreasing functions with respect to a point $q_i \in Q$. Although the weight function $w_{i,j}$ in definition (1) does not have compact support, for practical reasons it can be assumed that the Gaussian with 4σ covers 99% of the support size. As a result, the matrices Φ and Ψ are sparse, and the number of their non-zero entries depend on the support size of $w_{i,j}$. Following Definition 3.2, we estimate the number of non-zero entries in each row of the matrices Φ and Ψ, in the kth iteration of our algorithm, as

$$\Phi_{q_i^{(k)}} = \#\{B_h(q_i^{(k)}) \cap P\},$$
$$\Phi_{p_j^{(k)}} = \#\{B_h(p_j^{(k)}) \cap Q^{(k)}\},$$
$$\Psi_{q_i^{(k)}} = \#\{B_h(q_i^{(k)}) \cap Q^{(k)}\},$$

where $B_h(x)$ is a ball centered at x with radius h.

Using these definitions, we calculate the Hessian and its eigenvalues for our cost function in (1),

$$H = \nabla^2 G(Q) = \nabla^2 E_1 + \Lambda \nabla^2 E_2.$$

For simplicity, we denote $r_{i,j} = q_i - p_j$; then with $w_{i,j} = \exp\left\{-\|q_i - p_j\|^2/h_1^2\right\}$, $\frac{\partial E_1}{\partial q_i}$ can be rewritten as

$$\frac{\partial E_1}{\partial q_i} = \sum_{j=1}^{J} \frac{r}{\|r_{i,j}\|} \left(1 - \frac{2}{h_1^2}\|r_{i,j}\|^2\right) w_{i,j}.$$

We notice that, by definition, $\frac{\partial^2 E_1}{\partial q_i \partial q_i'} = 0$, and by the chain rule we have

$$\frac{\partial^2 E_1}{\partial q_i^2} = \sum_{j=1}^{J} a(r_{i,j})w_{i,j},$$

where $a(r) = -\frac{2}{h_1^2}\|r\| \left(1 + \frac{2}{h_1^2}\|r\|^2\right) < 0$.

For the second term in expression (1), we denote $\hat{r}_{i,i'} = q_i - q_i'$, and recall that $\eta(r) = \frac{1}{r^3}$. Then the first derivative of E_2 is

$$\frac{\partial E_2}{\partial q_i} = \sum_{i'=1}^{I} \left(-\frac{\hat{r}_{i,i'}}{\|\hat{r}_{i,i'}\|^5} - \frac{2\hat{r}_{i,i'}}{3h_2^2\|\hat{r}_{i,i'}\|^3}\right) \hat{w}_{i,i'}.$$

The second derivatives can be expressed as

$$\frac{\partial^2 E_2}{\partial q_i \partial q_i'} = -b(\hat{r}_{i,i'})\hat{w}_{i,i'},$$

where $b(\hat{r}) = \frac{4}{\|\hat{r}\|^5} + \frac{3\frac{1}{3}}{h_2^2\|\hat{r}\|^3} + \frac{4}{3h_2^4\|\hat{r}\|} > 0$, and

$$\frac{\partial^2 E_2}{\partial q_i^2} = \sum_{i'=1}^{J} b(\hat{r}_{i,i'})\hat{w}_{i,i'}.$$

Thus,

$$H = \begin{pmatrix} \sum_{j=1}^{J} a(r_{1,j})w_{1,j} + \lambda_1 \sum_{i'=1}^{I} b(\hat{r}_{1,i})w_{1,i'}; & -\lambda_1 b(\hat{r}_{1,2]})w_{1,2}; & \dots & -\lambda_1 b(\hat{r}_{1,I})w_{1,I} \\ \vdots & \vdots & \vdots & \vdots \\ -\lambda_I b(\hat{r}_{I,1})w_{I,1}; & \dots & -\lambda_I b(\hat{r}_{I,I-1})w_{I,I-1}; & \sum_{j=1}^{J} a(r_{I,j})w_{I,j} + \lambda_I \sum_{i'=1}^{I} b(\hat{r}_{i,i'})\hat{w}_{i,i'} \end{pmatrix}.$$

Let us check that the eigenvalues $\lambda_{\min}$, and $\lambda_{\max}$ of the MLOP Hessian $H \in \mathbb{R}^{I \times I}$ are bounded and negative. By Theorem 4.2, the eigenvalues of H belong to the range $\lambda_{\min}, \lambda_{\max} \in [l, u]$, where in our case

$$l = \min_{i \in I} \left(\sum_{j=1}^{J} a(r_{i,j})w_{i,j} + \lambda_i \sum_{i'=1}^{I} b(\hat{r}_{i,i})\hat{w}_{i,i'} - \sum_{i'=1}^{I} |\lambda_i b(\hat{r}_{i,i'})\hat{w}_{i,i'}| \right).$$

Let $h = min(h_1, h_2)$. Using the expressions for $a(r)$ and $b(r)$, and the fact that from Definition 3.2 $\|r\| = 4\sigma = \frac{4h}{\sqrt{2}}$, it can be verified that $0 < \min(b(r)) \le \frac{c_1}{h^5}$, $\min(a(r)) \le \frac{-c_2}{h\sqrt{2}}$, $\max(a(r)) \le 0$, where c_1, c_2 are constants and $c_1, c_2 > 0$. Thus, since $\lambda_i < 0$ from (6), and the number of points from P and Q in the support of q_i estimated by $\Phi_{q_i^{(k)}}$ and $\Psi_{q_i^{(k)}}$, respectively, we have

$$u \le -\frac{c_2}{h\sqrt{2}} \max_{i \in I}(\Phi_{q_i^{(k)}}) < 0, \tag{10}$$

$$l \le -\frac{c_2}{h\sqrt{2}} \max_{i \in I}(\Phi_{q_i^{(k)}}) - \frac{2c_1}{h^5} \max_{i \in I}(|\lambda_i|) \max_{i \in I}(\Psi_{q_i^{(k)}} - 1) < 0. \tag{11}$$

Since the eigenvalues are negative, all saddle points of the MLOP target function are strict-saddle, and the second condition of Theorem 4.1 holds. Let us also check that the first condition in Theorem 4.1 is satisfied, i.e., that the norm of the Hessian is bounded: $\|H\| \le L$, and find L. Indeed,

$$\|H\|_2 = \lambda_{\max}(H'H) = \lambda_{\max}(H^2) = \max\{\lambda^2 \mid \lambda \text{ is an eigenvalue of } H\} = \max\{\lambda_{\max}^2, \lambda_{\min}^2\},$$

so the required bound holds with $L = \max\{\lambda_{\max}^2, \lambda_{\min}^2\} \le \max\{u^2, l^2\} = l^2$.

To summarize, all the conditions of Theorem 4.1 are satisfied. It follows that the gradient descent with random initialization and a sufficiently small constant step size converges almost surely to a local minimizer or minus infinity. $\square$

4.2 Order of Approximation

The support size of the locally supported function defining the weight function $w_{i,j}$ which is tightly related to the fill-distance of available sample data P, plays an important role in the order of approximation of the MLOP algorithm. The following theorem guarantees an $O(h^2)$ order of approximation, which is asymptotic as $h \to 0$. Here, $h = \max(h_1, h_2)$, where h_1 and h_2 are defined in Remark 3.5.

Theorem 4.4 (Order of approximation). *Let $P = \{p_j\}_{j=1}^J$ be a set of points that are sampled (without noise) from a d–dimensional C^2 manifold $\mathcal{M}$, and satisfy the h-ρ condition. Then for a fixed ρ, and a finite support of size h of the weight functions $w_{i,j}$, the set of points Q defined by the MLOP algorithm has an order of approximation $O(h^2)$ to $\mathcal{M}$.*

Proof. We break the proof into the following steps.

1. **The MLOP cost function can be rewritten in matrix form as $AQ = R$.** We look for a solution Q that will minimize the cost function in (1), i.e., such that the gradient $\nabla G(Q) = 0$. Thus equation (3) can be recast as a system of equations

$$(1 - \tau_{i'})q_i + \tau_i \sum_{i' \in I \setminus \{i\}} q_{i'}^{(k)} \frac{\beta_i^{i'}}{\sum\limits_{i' \in I \setminus \{i\}} \beta_i^{i'}} = \sum_{j=1}^J p_j \frac{\alpha_j^{i'}}{\sum\limits_{j \in J} \alpha_j^{i'}}, \tag{12}$$

where we express $\lambda_{i'}$ in the form $\lambda_{i'} = \tau_{i'} \dfrac{\sum\limits_{j \in J} \alpha_j^{i'}}{\sum\limits_{i' \in I \setminus \{i\}} \beta_i^{i'}}$.

As a result, the problem can be written in matrix form as $AQ = R$, where both A, and R depend on Q. In the new notations, we need to show that the points $Q = A^{-1}R$ lie at a distance of $O(h^2)$ from $\mathcal{M}$.

2. **The R term has order of approximation of $O(h^2)$ to $\mathcal{M}$.** Let J_k be the indices of points from P which lie at the distance h from a given poin $q_{i'}$ (the set is not empty due to the optimal neighborhood selection in Subsection 3.2). Let t be the index of the closest point in $\{p_j\}_{j \in J_k}$ to the projection of $q_{i'}$ on the manifold $\mathcal{M}$ (Figure 5 left), and T be the tangent space to $\mathcal{M}$ at that point. Then the sum $\sum_{j=1}^J p_j \frac{\alpha_j^{i'}}{\sum_{j \in J} \alpha_j^{i'}}$, is a local convex combination of points p_k within a distance h from $q_{i'}$, and thus it also lies in T, which is affine. Since $\mathcal{M}$ is C^2, T approximates $\mathcal{M}$ in the order of $O(h^2)$, the right hand side of (12) can be written as $F + O(h^2)$, where $F = \{f_i\}_{i \in I}$ are points on $\mathcal{M}$. Thus, $AQ = F + O(h^2)$.

3. **Then norm of the matrix A^{-1}, $\|A^{-1}\|_\infty$ and its entries $(A^{-1})_{l,m}$ are bounded.** For $\tau_i \in [0, 0.5)$, the matrix A is strictly diagonally dominant and therefore we can bound $\|A^{-1}\|_\infty \leq c_1(\tau_i)$, as well as $|(A^{-1})_{l,m}| < c_2(\tau_i)$ for two points q_l and q_m lying at a distance of at least h, where the influence of distant points decays exponentially with distance. We also note that since the rows of A sum up to one, so do the rows of A^{-1}.

4. **The MLOP reconstruction is of order $O(h^2)$ to the manifold.** The MLOP reconstruction can be written as $Q = A^{-1}F + O(h^2)$, where each element of $(A^{-1}F)_{i'}$ is the affine average of f_i over the manifold, with exponentially decaying weights $w_{i,j}$. Let T be the tangent space to the manifold $\mathcal{M}$ at the point $f_{i'}$, and let t_i be the projection of f_i

on T (Figure 5 right). If we rewrite f_i using its projection as $f_i = t_i + r_i$, it follows that $(A^{-1}F)_{i'} = \sum_{i \in I} A_{i',i}^{-1}(t_i + r_i) = \sum_{i \in I} A_{i',i}^{-1}t_i + \sum_{i \in I} A_{i',i}^{-1}r_i$. We would first like to show that $\|\sum_{i \in I} A_{i',i}^{-1}t_i - f_{i'}\| = O(h)$, and since $\sum_{i \in I} A_{i',i}^{-1}t_i$ is on T, and T approximates the manifold with $O(h^2)$, it will follow that $\sum_{i \in I} A_{i',i}^{-1}t_i$ is of order $O(h^2)$ distance from $\mathcal{M}$. In addition, we show that $\sum_{i \in I} A_{i',i}^{-1}(r_i) = O(h^2)$.

In more details:

(a) For a given $q_{i'}$, we denote by I_k its q_i neighbors at the distance $\|q_i - q_{i'}\| \in [kh, (k+1)h]$. We use the fact that the sum of the rows of A^{-1} equals one, and rewrite and estimate $\sum_{i \in I} A_{i',i}^{-1}t_i$ as

$$\|\sum_{i \in I} A_{i',i}^{-1}t_i - f_{i'}\| = \|\sum_{i \in I} A_{i',i}^{-1}(t_i - f_{i'})\| \leq \sum_{i \in I} c_2(\tau)\|t_i - f_{i'}\| = O(h). \tag{13}$$

For the last step we note that $\|t_i - f_{i'}\| = \|t_i - f_i + f_i - f_{i'}\| \leq \|t_i - f_i\| + \|f_i - f_{i'}\| \leq O(h) + (k+1)h$, due to the local approximation property and the distance constraint on the point q_i. Thus, the sum $\sum_{i \in I} A_{i',i}^{-1}t_i$ is an affine combination of points t_i on T and therefore lies in T as well (in a distance $\leq O(h)$), therefore it will follow that it is an $O(h^2)$ from the manifold.

(b) Next, similar considerations show that $\|r_i\| \leq \|f_i - f_{i'}\|^2 \leq c_3((k+1)h + O(h))^2$.

To conclude, that based on items (a) and (b), the MLOP order of approximation to the manifold is $O(h^2)$.
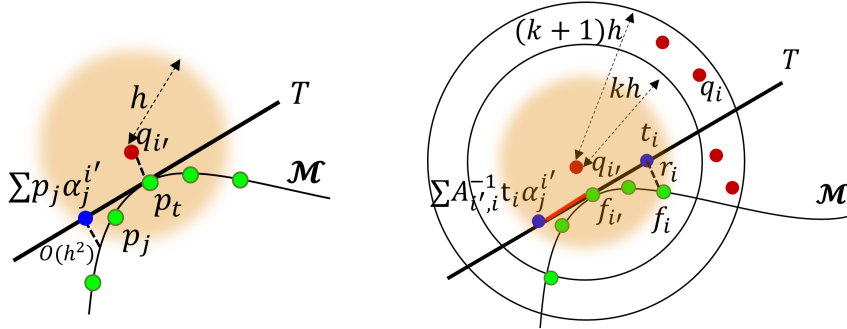
$\square$

Figure 5: Illustration of the points participating in the estimate of the order of approximation. Left: demonstration why the affine combination of the p_j points, in the neighborhood of $q_{i'}$, is of order $O(h^2)$. Right: Illustration of the elements used in the estimation of the order of approximation. The P points are marked in green, the Q points in red, while the auxiliary points in the proof are marked in blue.

4.3 Rate of Convergence

First, let us consider the gradient-descent rate of convergence of a Lipschitz-continuous strongly convex function. This rate of convergence depends on the condition number of the Hessian of

the cost function, and so on the ratio between the smallest and the largest eigenvalues of the Hessian, i.e., $|1 - c\frac{\lambda_{\min}}{\lambda_{\max}}|$, with $0 < c < 2$. Therefore, if our cost function would be convex, the rate of convergence could be $O(1 - c/h^4)$. However, for non-convex optimization, the situation is much more complex. In our setting, where there is no convexity, one can analyze convergence to ϵ-first-order stationary points, as defined below.

Definition 4. *A differentiable function $f(\cdot)$ is called L-smooth if for any x_1, x_2*

$$\|\nabla f(x_1) - \nabla f(x_2)\| \leq L\|x_1 - x_2\|.$$

Definition 5. *If $f(\cdot)$ is a differentiable function, we say that x is an ϵ-first-order stationary point if $\|\nabla f(x)\| \leq \epsilon$.*

For the rate of convergence of our method, we will use the following theorem proved by Nestrove in [33].

Theorem 4.5. *Let $f(\cdot)$ be an L- smooth function that is bounded below. Then for any $\epsilon > 0$, for the gradient descent with step size $\gamma = \frac{1}{\epsilon}$ and stop criterion $\|\nabla f(x)\| \leq \epsilon$, the output will be an ϵ-first-order stationary point, which will be reached after $k = \frac{L(f(x_0) - f^*)}{\epsilon^2}$ iterations. In case the starting point is close enough to the local minimum, the convergence is linear.*

It follows that in our case the rate of convergence is bounded.

Theorem 4.6 (Rate of convergence)**.** *Let the points-set $P = \{p_j\}_{j=1}^J$ be sampled near a d-dimensional manifold in $\mathbb{R}^n$ and let the assumptions in Theorem 4.3 be satisfied. Let the cost function G, defined as in (1), be an L-smooth function. For any $\epsilon > 0$, let Q^* be a local fixed-point solution of the gradient descent iterations, with step size $\gamma = \frac{1}{\epsilon}$. Set the termination condition as $\|\nabla G(Q)\| \leq \epsilon$. Then Q^* is an ϵ-first-order stationary point that will be reached after $k = \frac{L(G(Q^{(0)}) - G(Q^*))}{\epsilon^2}$ iterations, where $L = l^2$ and l is given in (11).*

Proof. It is quite easy to verify that $G(Q)$ satisfies all the conditions of Theorem 4.5; in particular, the L-smoothness condition was proven above. □

Remark 4.7. In our case, due to the bound on l in (11), we see that k is of order $\frac{1}{h^{10}}$. However, in practice, in our numerical examples, fewer iterations were needed to achieve convergence. In an example presented in the following section, with approximately 800 noisy points P and 160 points in Q (sampled in a certain area around a specific point), of a two-dimensional manifold embedded into a 60-dimensional space, the method converged in approximately 500 iterations which took around 90 seconds. When the initial set Q was randomly sampled from P, we observed convergence in 50 iterations which took 11 seconds.

Remark 4.8. It should be emphasized that the calculations of the gradient for each point are independent of one another, and in order to reduce the execution time, they can be run in multiple threads.

4.4 Uniqueness

As shown in the previous section, convergence to a local minimum is guaranteed. However, since the cost function in (1) is non-convex, a unique global solution can not be ensured. In order to address the uniqueness question, we have to rephrase the notion of uniqueness for our

case. We do not refer to the uniqueness of the set Q, since there may be many sets Q which satisfy the cost function (1), but to a common property of these optimal Q sets, the fill-distance of their points. For instance, given a solution, its linear transformation can still minimize (1). This scenario is illustrated in Figure 6. In this example, which will be explained in detail in the experimental section, the orthogonal matrices in $\mathbb{R}^2$, which are represented by their angle, form a manifold. Although the two sets in Figure 6 (left and right) differ, they can still be solutions to the problem.
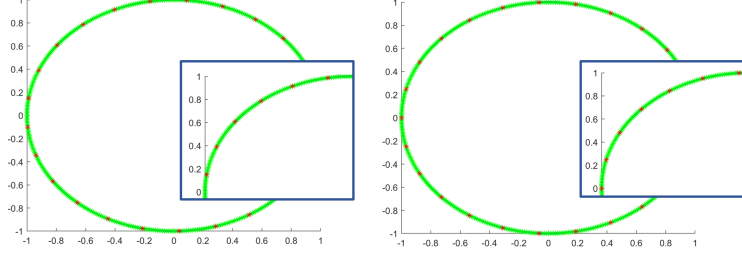
Figure 6: Manifold of orthogonal matrices: each matrix is represented by means of an angle (green), sampled with the same fill-distance, in two manners (red).

Thus the appropriate notation of uniqueness of the solution is as follows:

Definition 6. *Let Q_1 and Q_2 be two point-sets uniformly sampled from a manifold $\mathcal{M}$, with fill-distance h_2^1 and h_2^2, respectively. Then Q_1 and Q_2 are said to be "distribution equivalent" if their fill-distances coincide ($h_2^1 = h_2^2$). For a fixed fill-distance h_q, the corresponding class of distribution equivalent sets is denoted here by $[h_q]$.*

Remark 4.9. Let Q^* be a solution of the optimization problem (1), from points P. Then Q^* is unique up to the equivalence class $[h_q]$. This follows from the definition of h_q, which specifies the number of P points served by a single q_i, which uniquely define the equivalence class $[h_q]$ of the solution Q^*.

4.5 Complexity of the MLOP Algorithm

The complexity of the MLOP algorithm described in Algorithm 1 is based on a pre-step and a gradient decent iterations. As described in Section 3.1, due to the curse of dimensionality and presence of noise all the norms are calculated in a lower dimension m. Thus, a pre-step to the MLOP algorithm is reducing the dimension of P from n to m (where $m \ll n$), and have the complexity nmJ. In addition in every gradient descent step, and for every q_i we reduce the dimension of current Q which results in the complexity of nmI. As a result, a single gradient descent step is $O(I(nmI + I + J))$. With efficient neighboring calculation, this can be reduced to $O(I(nm\hat{I} + \hat{J}))$, where $\hat{I}$ and $\hat{J}$ are the numbers of points in the support of the weight function with respect to the Q and P sets, respectively (for instance, in the numerical examples below $\hat{J}$ was around 30 points, instead of 900 points in P). These operations are repeated k times until convergence, where k is bounded as in Theorem 4.6. Thus, the overall complexity is $O(nmJ + kI(nm\hat{I} + \hat{J}))$.

Corollary 4.10. *Given a point-set $P = \{p_j\}_{j=1}^J$ sampled near a d-dimensional manifold $\mathcal{M} \in \mathbb{R}^n$, let $Q = \{q_i\}_{i=1}^I$ be a set of points that will provide the desired manifold reconstruction. Then the complexity of the MLOP algorithm is $O(nmJ + kI(nm\hat{I} + \hat{J}))$, where the number of iterations k is bounded as in Theorem 4.6, $m \ll n$ is the smaller dimension to which we reduce*

the dimension of the data, and $\widehat{I}$ and $\widehat{J}$ are the numbers of points in the support of the weight functions $\hat{w}_{i,i'}$, $w_{i,j}$ with the Q-set and P-set, respectively. Thus, the approximation is linear in the ambient dimension n, and does not depend on the intrinsic dimension d.

5 Numerical Examples

In this section, we present some numerical examples which demonstrate the validity of our method, as well as its robustness under different scenarios, for example, diverse manifold topologies, different amounts of noise, and many intrinsic dimensions. In all the examples the input points P were sampled uniformly in the parameter space. Next, a uniform noise $U(-\sigma, \sigma)$ with magnitude σ was added. Then the set Q was initialized by sampling from the set P around a certain selected point. In what follows we illustrate the results of applying the MLOP algorithm.

One-Dimensional Orthogonal Matrices

Consider the case of the manifold $O(2)$ of orthogonal matrices, embedded into a 60-dimensional linear space by using the parameterization

$$\hat{p} = [\cos(\theta), -\sin(\theta), \sin(\theta), \cos(\theta), 0, \ldots, 0],$$

where $\theta \in [-\pi, \pi]$. The input data $\hat{P}$ were constructed by sampling 500 equally distributed points in the parameter space. Next, we randomly sampled an orthogonal matrix $A \in \mathbb{R}^{60 \times 60}$, and created a new point-set via non-trivial vector embedding

$$P = A\hat{P}. \tag{14}$$

Later we added a uniform noise $U(-0.2, 0.2)$, and initialized the set Q selecting 50 points around a certain point. Figure 7 left illustrates the first two coordinates of the points in our set (after a multiplication with A^{-1}). The noisy sampled points are shown in green, while the initial reconstruction points are shown in red. Figure 7 right shows the reconstructed and denoised manifold of orthogonal matrices, after 500 iterations of the MLOP algorithm (red).
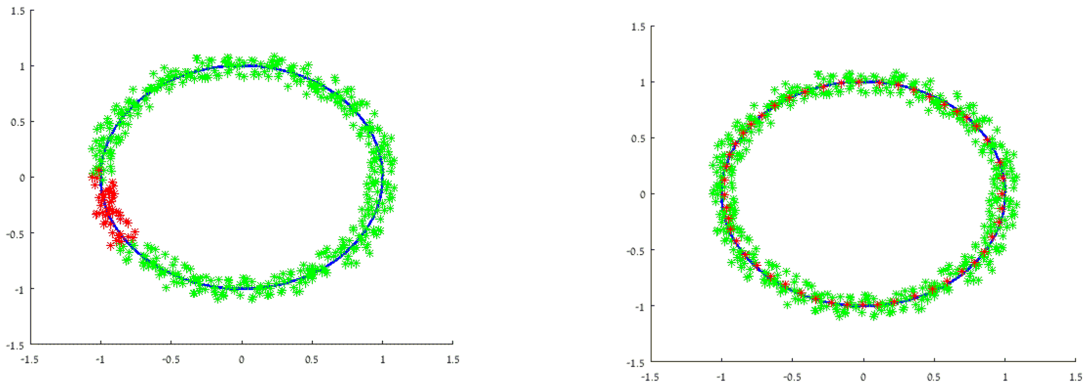

Figure 7: Manifold of orthogonal matrices embedded into a 60-dimensional space. Shown are the first two coordinates of the point-set (after multiplication with A^{-1}). Left: Scattered data with uniformly distributed noise $U(-0.2; 0.2)$ (green), and the initial point-set $Q^{(0)}$ (red) Right: The resulting point-set of MLOP algorithm after 500 iterations, $Q^{(500)}$ (red) overlaying the noisy samples (green).

Three-Dimensional Cone Structure

Next, we demonstrate the ability of the MLOP to cope with a geometric structure of different dimensions at different locations. Here we combined a 3-dimensional manifold, namely, a cone structure, with a one-dimensional manifold, namely, a line segment. This object was embedded into a 60-dimensional linear space. The cone's parameterization used was

$$p = tv_1 + \frac{e^{-R^2}}{\sqrt{2}}(\cos(u)v_2 + \sin(u)v_3),$$

where $v_1 = [1, 1, 1, 1, 0, \ldots, 0], v_2 = [0, 1, -1, 0, 0, \ldots, 0], v_3 = [1, 0, 0, -1, 0, \ldots, 0], (v_1, v_2, v_3) \in \mathbb{R}^{60}$, $t \in [0, 2]$, $R \in [0, 2.5]$, and $u \in [0.1\pi, 1.5\pi]$. We sampled 720 points from the structure with added uniformly distributed noise of magnitude 0.2. The initial set $Q^{(0)}$ of size 144 was selected (Figure 8 left), and 500 iterations of the MLOP were performed to reconstruct and denoise the geometrical structure (Figure 8 right).
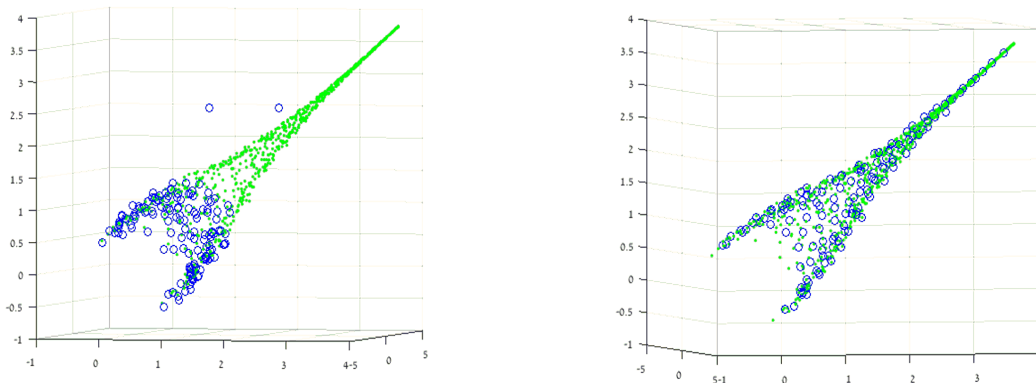

Figure 8: Geometrical structure of changing dimension. Combination of a cone and a line segment, embedded into a 60-dimensional space. The first three coordinates of the point-set are shown. Left: Scattered data with uniformly distributed noise $U(-0.2; 0.2)$ (green), and the initial point-set $Q^{(0)}$ (blue) Right: The point-set generated by the MLOP algorithm after 500 iterations, $Q^{(500)}$ (blue) overlaying the noisy samples (green).

Two-Dimensional Cylindrical Structure

In the next example, we embedded a two-dimensional cylindrical structure into a 60-dimensional linear space. We sampled the structure using the parameterization

$$p = tv_1 + \frac{R}{\sqrt{2}}(\cos(u)v_2 + \sin(u)v_3),$$

where $v_1 = [1, 1, 1, 1, 1, \ldots, 1]$, $v_2 = [0, 1, -1, 0, 0, \ldots, 0], v_3 = [1, 0, 0, -1, 0, \ldots, 0], (v_1, v_2, v_3 \in \mathbb{R}^{60})$, $t \in [0, 2]$ and $u \in [0.1\pi, 1.5\pi]$. Using this representation 816 equally distributed (in parameter space) points were sampled with uniformly distributed noise (i.e., $U(-0.1, 0.1)$). As can be seen in Figure 9 left, the initial set $Q^{(0)}$ of size 163 was selected very roughly, and 500 iterations of the MLOP were performed to reconstruct the cylindrical structure, shown in Figure 9 right.
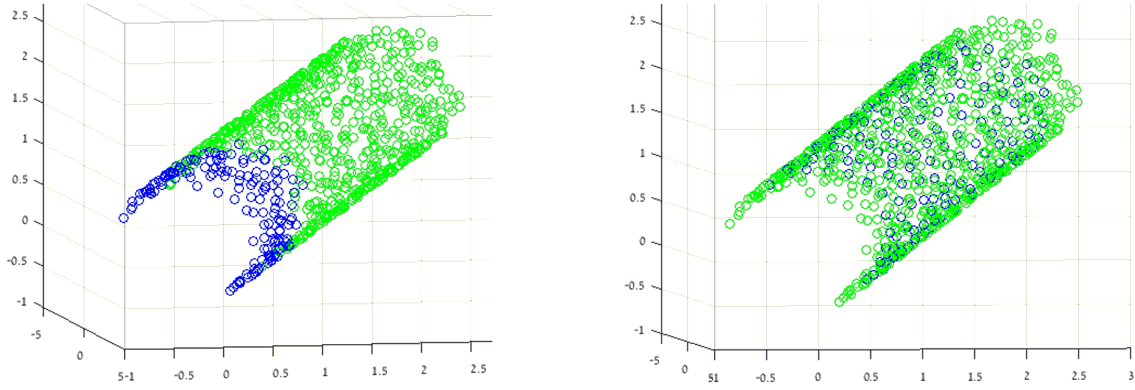
Figure 9: Cylindrical structure embedded into a 60-dimensional space. The first three co-ordinates of the point-set are shown. Left: Scattered data with uniformly distributed noise $U(-0.1; 0.1)$ (green), and the initial point-set $Q^{(0)}$ (blue) Right: The point-set generated by the MLOP algorithm after 500 iterations, $Q^{(500)}$ (blue) overlaying the noisy samples (green).

Robustness to Noise

The noise level has a direct influence on the accuracy of the reconstruction. Here we examine the robustness of the MLOP under various levels of noise. Our test was performed on the two-dimensional cylindrical structure embedded into 60-dimensions, with various amounts on noise magnitude (0, 0.1, 0.2, and 0.5). The accuracy was calculated as the relative error of the reconstruction Q, against a densely sampled noise-free cylindrical structure. The norm used for accuracy calculations was the one that is based on linear sketching, as defined in Section 3.1. As can be seen in Figure 10, even with a noise level of 0.5, the reconstruction quality is satisfactory (with a relative error of 0.15).
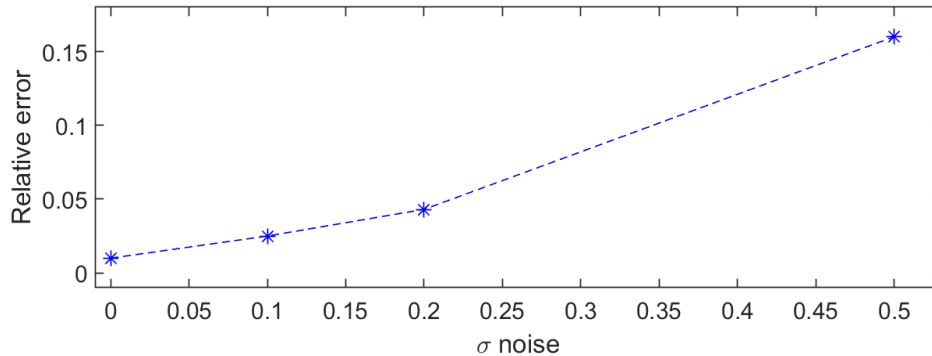

Figure 10: Effect of noise level on the reconstruction accuracy of a cylindrical structure embedded into a 60-dimensional space.

Six-dimensional cylindrical structure

Finally, we tested our method on manifolds of the higher dimension by utilizing an n-sphere to generate an $(n + 1)$-dimensional cylinder (in the example of the two-dimensional cylinder, we used a circle to generate the structure). Here, we utilized a five-dimensional sphere to build a six-dimensional manifold, using the parameterization

$$x_1 = R \cos(u_1), \quad x_2 = R \sin(u_1) \cos(u_2), \quad \ldots, \quad x_6 = R \sin(u_1) \sin(u_2) \cdots \sin(u_5) \sin(u_6).$$

21

We then embedded the sampled data in a 60-dimensional space

$$p = tv_0 + R^2[x_1, x_2, x_3, x_4, x_5, x_6, 0, \ldots, 0], \qquad (15)$$

where $R = 1.5$, $t \in [0,2]$, $u_i \in [0.1\pi, 0.6\pi]$, and $v_0 \in \mathbb{R}^{60}$ is a vector with 1's in positions $1, ..., d+1$ and 0 in the remaining positions. In this test, we sampled 1200 points from this manifold and added a noise $U(-0.1, 0.1)$. The initial reconstruction set was chosen to consist of randomly selected 460 points. The method converged after approximately 300 iterations. To avoid trying to visualize a six-dimensional manifold, we plot in Figure 11 the cross-section of the cylindrical structure in three-dimensions. We evaluate the efficiency of the denoising effect by calculating the maximum relative error, root mean square error, and variance of both the initial $Q^{(0)}$ points and the noise-free reconstruction set $Q^{(300)}$ with respect to the closest point in the clean reference data. As a result, the errors if $Q^{(0)}$ are 0.083, 0.32 ± 0.0007, and of the noise-free reconstruction are 0.058, 0.28 ± 0.0006. Thus, we see that in this scenario of non-trivial intrinsic dimension of the manifold the error decrease dramatically. In addition, the fill-distance of the initial random $Q^{(0)}$ set was 0.36, and 0.32 in the reconstruction. Thus, we also observe the effect of quasi-uniform sampling after applying the MLOP.
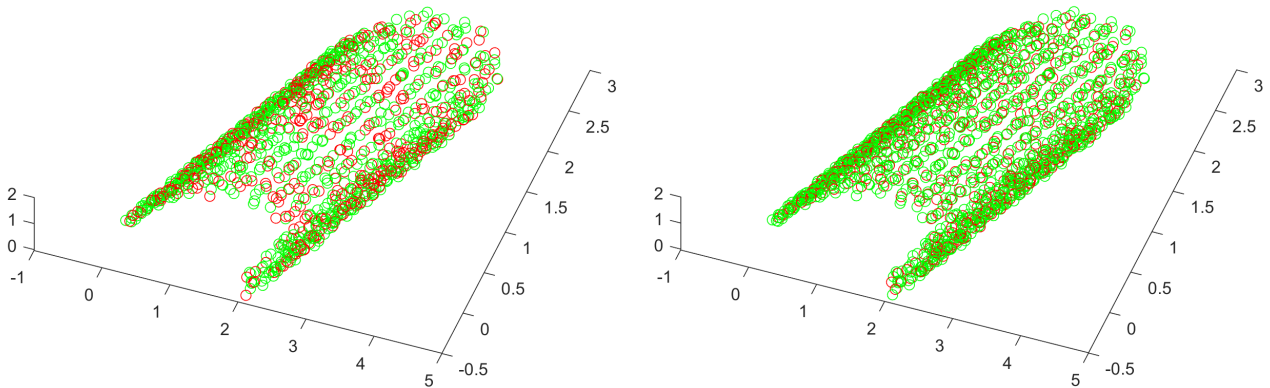

Figure 11: Six-dimensional cylindrical structure embedded in a 60-dimensional space. The cross-section of the six-dimensional cylindrical structure is plotted in three-dimensions. Left: Scattered data with uniformly distributed noise $U(-0.1; 0.1)$ (green), and the initial point-set $Q^{(0)}$ (red) Right: The point-set generated by the MLOP algorithm after 300 iterations, $Q^{(300)}$ (red) overlaying the noisy samples (green).

Applications to Image Processing

Manifold denoising and reconstruction methodology can be also applied to image processing problems. At the beginning of this paper, we described the cryo-EM (in Figure 1) which motivated our study. In this framework a manifold is created by acquiring images of a single object in various directions. As a preliminary example, before addressing the real case of cryo-EM, we simulated data that resemble the cryo-EM conditions. Specifically, we sampled 900 images of ellipses of size 20×20. The ellipses were centered and no rotations were used. Thus, we have 900 samples of a 2-dimensional submanifold embedded in $\mathbb{R}^{400}$. We added a Gaussian noise $N(0; 0.05)$ to each pixel. Figure 12 shows the sample of the manifold (with some zoom-in examples), along with a graph where the (x, y) - coordinates of each point are the ellipse radii. For the execution of the MLOP, we took 180 ellipses as the initial sample points (Figure 13 left). As can be seen in Figure 13 right, after 1000 iterations the samples were cleaned, while the radii distribution graph shows that the radii domain is fully sampled.

We evaluated the MLOP denoise performance on the ellipses samples Q. We measured the SNR as $SNR = \frac{\mu}{\sigma}$ on the background pixels of each ellipse image (where μ is the average signal value, while σ is the standard deviation). We observe that the median SNR of the set Q increased after applying the MLOP denoising, from 15.6 to 36.5. This gives us a quantitative measure of the denoising performed by the MLOP (as can also be seen in Figure 13 in the zoomed-in areas).
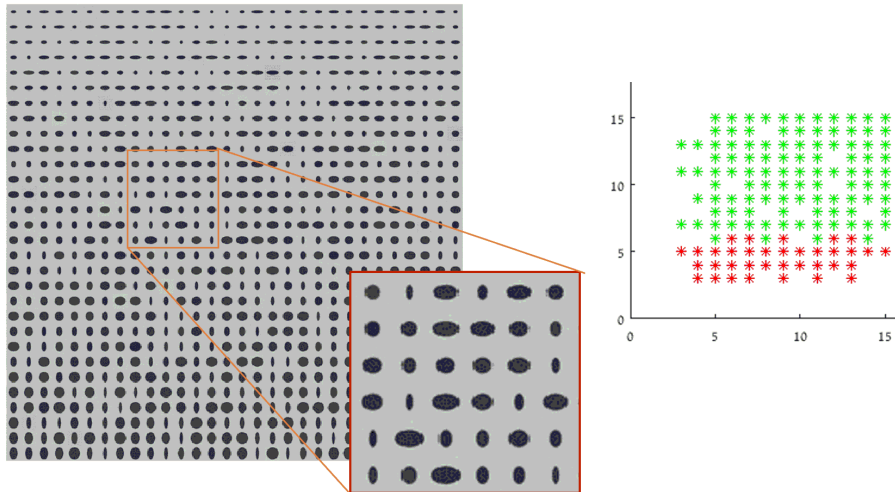

Figure 12: Left: Images of ellipses with varying radii that were sampled from a 2-dimensional manifold, prior to adding noise, which will form the P set. Right: a graph depicting the radii of the ellipses, with the coordinates of points given by these radii. The manifold samples are shown in green (P), while the initial set $Q^{(0)}$ is shown in red.
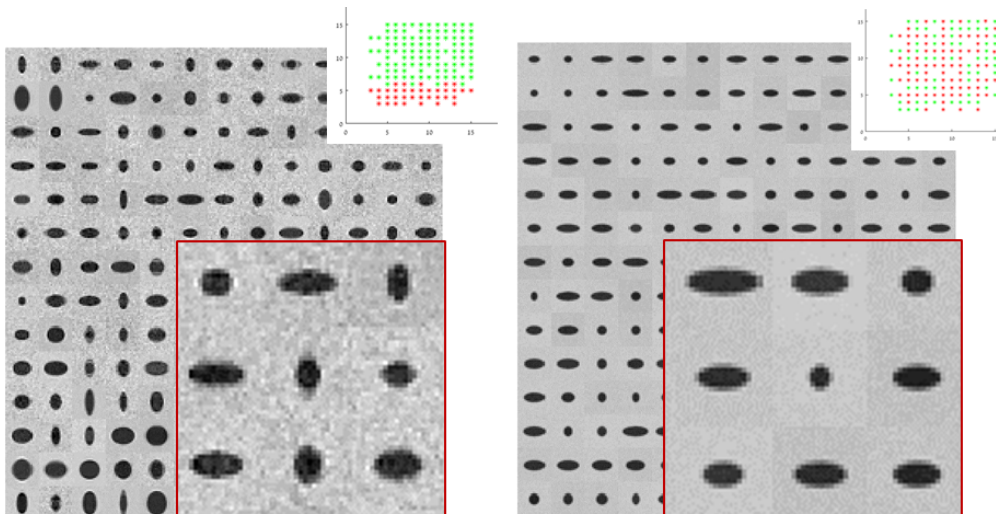

Figure 13: The samples that were used to reconstruct the manifold. Each side of the figure consists of an image of the samples, a zoomed-in area, and a graph of sample radii. The manifold samples are shown in green, while the initial set is shown in red. Left: the initial configuration of points sampled from the 2-dimensional manifold. Right: the manifold reconstruction configuration after 1000 iterations.

6 MLOP Denoise Benefits

The current section dealt both with manifold reconstruction and cleaning of high amounts of noise. The denoising property was induced by the first term in (1), which performs smoothing

of p_j samples in the neighborhood of the examined point q_i. This term is inspired by the L_1-median [42], and thus is robust to high amounts of noise. This fact was demonstrated in the "Robustness to Noise" subsection in [16], where the effect of various levels of noise on reconstruction accuracy was examined. The test demonstrated the robustness of the MLOP method to various amounts of noise magnitude (0, 0.1, 0.2, and 0.5), on a two-dimensional cylindrical structure embedded into 60-dimensions. The calculation of relative error of the reconstruction Q, against a densely sampled noise-free cylindrical structure, showed good results even at a noise level of 0.5 (with a relative error of 0.15). Thus, it is natural to use MLOP as a pre-processing step prior to performing mining tasks on the data.

In this section, we demonstrate the effectiveness of high-dimensional denoising in the case of local PCA. In our test we examine a set of points $X = \{x_i\}$, with a fill-distance h. We calculate PCA for each point x_i using its neighboring points x_j, which maintain the constraint $\|x_i - x_j\| < h$. Next, we extract the first eigenvector and evaluate its accuracy with respect to the first eigenvector of a PCA executed on clean reference data. Specifically, for each point x_i we find the closest point in the clean reference data and calculate the cosine distance between the corresponding PCA first eigenvectors (the error is given in degrees). Next, we determine the median of the errors stemming from all the points X. It is important to note that the error is tightly connected with the number of points in the set, with their fill-distance, and naturally with the noise levels. For example, on clean data with 160 points randomly sampled from a manifold, the error was 11.8, while with 7000 points, the error decrease to 0.2. This stems from the fact that taking a larger number of points in the neighbor of a point x_i leads to a more accurate eigenvector. This fact has to be taken into account in error analysis.

The numerical calculations were performed on the example of a two-dimensional cylindrical structure embedded into a 60-dimensional linear space. We sampled the structure using the parameterization

$$p = tv_1 + \frac{R}{\sqrt{2}}(\cos(u)v_2 + \sin(u)v_3),$$

where $v_1 = [1,1,1,1,1,\ldots,1]$, $v_2 = [0,1,-1,0,0,\ldots,0]$, $v_3 = [1,0,0,-1,0,\ldots,0]$ ($v_1, v_2, v_3 \in \mathbb{R}^{60}$), $t \in [0,2]$ and $u \in [0.1\pi, 1.5\pi]$. Using this representation, 816 uniformly distributed (in parameter space) points were sampled with uniformly distributed noise (i.e., $U(-0.2, 0.2)$). As can be seen in Figure 14 left, after 500 iterations of the MLOP algorithm, the cylindrical structure was reconstructed with high accuracy (red points).

The experiments testing the efficiency of MLOP denoising were carried out on five data sets, all of size 160:

1. Noise-free data.
2. Noise data with additive noise of 0.1.
3. Data denoised by the MLOP from the data in item 2.
4. Noisy data with additive noise of 0.2.
5. Data denoised by the MLOP from the data in item 4.

The results for noise levels of 0.1 and 0.2 are presented in Figure 14 right. To achieve a robust error value, we performed ten bootstrap iterations for the "noise-free", as well as "noisy data" data-sets, where we randomly sampled the manifold, and calculated the median PCA error of the iterations. As expected, the effect of the MLOP denoising is to improve the accuracy of the local PCA calculations. One can see that the noise level has a small effect on the error (increasing it from 7.9 to 8.2, for the 0.1 and 0.2 noise level respectively). An additional benefit

is that the accuracy of the denoised data is superior the one of the is noise-free data. The reason for this is the quasi-uniform manifold sampling which MLOP carries out accordingly due to the second term in (1), while the noise-free samples come from randomly sampled points (which not necessarily sample the manifold uniformly).
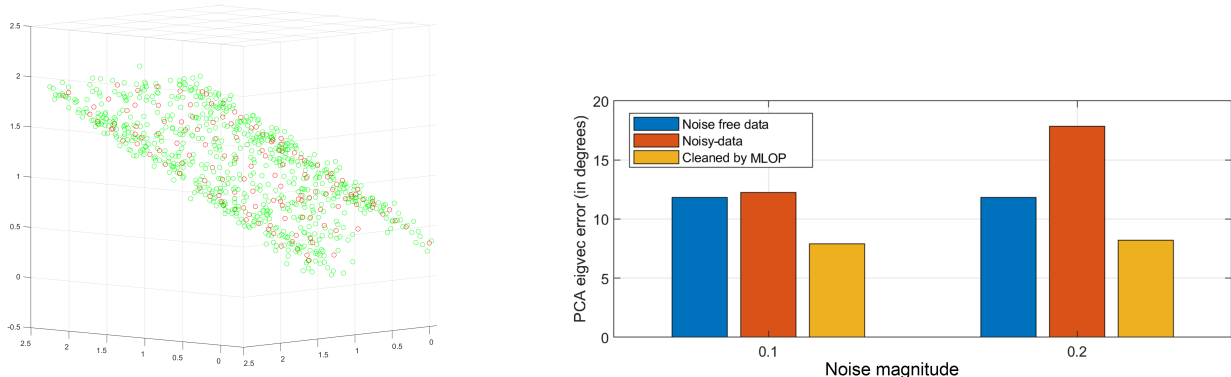

Figure 14: Left: Cylindrical structure, sampled with noise $U(-0.2, 0.2)$, and embedded in $\mathbb{R}^{60}$. The figure presents the first three coordinates of the points set. The point-set generated by the MLOP algorithm after 500 iterations, $Q^{(500)}$ (red) overlaying the noisy samples (green). Right: illustration of the MLOP denoising effect on the accuracy of PCA calculations. The graphs present the error of the first eigenvector of local PCA calculated on noise-free, noisy, and denoised data.

7   Discussion and Future Directions

The big-data era gave rise to many challenges related to processing, analyzing, and understanding high-dimensional data. Among these challenges are the presence of noise, outliers, incomplete data, or insufficient data. In this paper, we introduced a framework that can address these issues, raised by high-dimensional data, in an efficient and robust manner. We propose a method for manifold reconstruction and denoising in high-dimensional space. Over the years, several solutions were suggested to cope with the reconstruction problem in high-dimensional space. However, they have a hard time handling noisy data, non-uniformly sampled, with no assumption on the data. As a result, manifold reconstruction in noisy conditions in high-dimensional space is still an open question. In our research, we address the manifold approximation question by extending the LOP [30] algorithm to the high-dimensional case. We develop a new algorithm, called Manifold Locally Optimal Projection (MLOP). We look for a noise-free manifold reconstruction in high-dimensional space by solving a non-convex optimization problem which leverages L1-median generalization to high dimension, while requiring a quasi-uniform distribution of points in the reconstruction. We prove that the MLOP method converges to a local stationary solution with a bounded linear rate of convergence when the starting point is close enough to the local minimum. In addition, we showed that the manifold order of approximation is $O(h^2)$, where h is the representative distance between the points, and the complexity is linear in the ambient dimension and does not depend on the intrinsic dimension.

The numerical examples demonstrate the applicability of the proposed method to various high-dimensional scenarios. This opens the door to different applications. First, it is possible to extend the methodology for approximating function on a manifold in noisy conditions (both in the function domain and in its codomain). Next, it is possible to enhance the MLOP to address the manifold repairing in the high-dimension problem, where input data have holes,

and the target is to find a noise-free reconstruction of the manifold that will amend the holes and complete the missing information. Then, using the MLOP methodology it is possible to address the problem of multivariate k-L1-medians in high-dimensional cases. This can be achieved by finding the service centers by using the MLOP out-of-the-box. Last, but not least, the flexibility of selecting the amounts of points in the reconstruction and set the density paves the way for manifold upsampling and downsampling, and for manifold compression. Thus, we see the MLOP framework is a cornerstone method for handling high-dimensional noisy data.

Acknowledgments

References

[1] Aamari, E., Levrard, C., et al.: Nonasymptotic rates for manifold, tangent space and curvature estimation. The Annals of Statistics **47**(1), 177–204 (2019)

[2] Aggarwal, C.C., Hinneburg, A., Keim, D.A.: On the surprising behavior of distance metrics in high dimensional space. In: International Conference on Database Theory, pp. 420–434. Springer (2001)

[3] Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., Silva, C.T.: Computing and rendering point set surfaces. IEEE Transactions on Visualization and Computer Graphics **9**(1), 3–15 (2003)

[4] Bachmayr, M., Dahmen, W., DeVore, R., Grasedyck, L.: Approximation of high-dimensional rank one tensors. Constructive Approximation **39**(2), 385–395 (2014)

[5] Barzilai, J., Borwein, J.M.: Two-point step size gradient methods. IMA Journal of Numerical Analysis **8**(1), 141–148 (1988)

[6] Berger, M., Tagliasacchi, A., Seversky, L.M., Alliez, P., Guennebaud, G., Levine, J.A., Sharf, A., Silva, C.T.: A survey of surface reconstruction from point clouds. In: Computer Graphics Forum, vol. 36, pp. 301–329 (2017)

[7] Boissonnat, J., Ghosh, A.: Manifold reconstruction using tangential Delaunay complexes. Discrete & Computational Geometry **51**(1), 221–267 (2014)

[8] Boissonnat, J., Guibas, L.J., Oudot, S.Y.: Manifold reconstruction in arbitrary dimensions using witness complexes. Discrete & Computational Geometry **42**(1), 37–70 (2009)

[9] Chazal, F., Cohen-Steiner, D., Mérigot, Q.: Geometric inference for probability measures. Foundations of Computational Mathematics **11**(6), 733–751 (2011)

[10] Cheng, S.W., Dey, T.K., Ramos, E.A.: Manifold reconstruction from point samples. In: SODA, vol. 5, pp. 1018–1027 (2005)

[11] Cohen-Or, D., Levin, D., Remez, O.: Progressive compression of arbitrary triangular meshes. Proceedings of Visualization '99, IEEE (1999)

[12] Coifman, R.R., Lafon, S., Lee, A.B., Maggioni, M., Nadler, B., Warner, F., Zucker, S.W.: Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. Proceedings of the National Academy of Sciences **102**(21), 7426–7431 (2005)

[13] Cox, T.F., Cox, M.A.: Multidimensional Scaling. Chapman and Hall, London (2000)

[14] Domingos, P.M.: A few useful things to know about machine learning. Commun. ACM **55**(10), 78–87 (2012)

[15] Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. IEEE Transactions on Image processing **15**(12), 3736–3745 (2006)

[16] Faigenbaum-Golovin, S., Shaus, A., Sober, B., Levin, D., Na'aman, N., Sass, B., Turkel, E., Piasetzky, E., Finkelstein, I.: Algorithmic handwriting analysis of judah's military correspondence sheds light on composition of biblical texts. Proceedings of the National Academy of Sciences **113**(17), 4664–4669 (2016)

[17] Federer, H.: Curvature measures. Transactions of the American Mathematical Society **93**(3), 418–491 (1959)

[18] Fefferman, C., Ivanov, S., Kurylev, Y., Lassas, M., Narayanan, H.: Fitting a putative manifold to noisy data. In: Conference on Learning Theory, pp. 688–720 (2018)

[19] Fisher, R.A.: The use of multiple measurements in taxonomic problems. Annals of Eugenics **7**(2), 179–188 (1936)

[20] He, X., Niyogi, P.: Locality preserving projections. In: Advances in Neural Information Processing Systems, pp. 153–160 (2004)

[21] Huang, H., Li, D., Zhang, H., Ascher, U., Cohen-Or, D.: Consolidation of unorganized point clouds for surface reconstruction. ACM Transactions on Graphics (TOG) **28**(5), 176 (2009)

[22] Huang, H., Wu, S., Gong, M., Cohen-Or, D., Ascher, U., Zhang, H.R.: Edge-aware point set resampling. ACM Transactions on Graphics (TOG) **32**(1), 9 (2013)

[23] Iyengar, S.S., Kouri, D.J., Parker, G.A., Hoffman, D.K.: Estimating bounds on the highest and lowest eigenvalues of any matrix. Theoretical Chemistry Accounts **103**(6), 507–517 (2000)

[24] Lee, J.D., Simchowitz, M., Jordan, M.I., Recht, B.: Gradient descent only converges to minimizers. In: Conference on Learning Theory, pp. 1246–1257 (2016)

[25] Levin, D.: The approximation power of moving least-squares. Mathematics of Computation **67**(224), 1517–1531 (1998)

[26] Levin, D.: Mesh-independent surface interpolation. In: Geometric Modeling for Scientific Visualization, pp. 37–49. Springer (2004)

[27] Levin, D.: Between moving least-squares and moving least-ℓ_1. BIT Numerical Mathematics **55**(3), 781–796 (2015)

[28] Lin, T., Zha, H.: Riemannian manifold learning. IEEE Transactions on Pattern Analysis and Machine Intelligence **30**(5), 796–809 (2008)

[29] Lipman, Y., Cohen-Or, D., Levin, D.: Error bounds and optimal neighborhoods for MLS approximation. In: Proceedings of the fourth Eurographics Symposium on Geometry Processing, pp. 71–80 (2006)

[30] Lipman, Y., Cohen-Or, D., Levin, D., Tal-Ezer, H.: Parameterization-free projection for geometry reconstruction. In: ACM Transactions on Graphics (TOG), vol. 26, p. 22. ACM (2007)

[31] Maggioni, M., Minsker, S., Strawn, N.: Multiscale dictionary learning: non-asymptotic bounds and robustness. The Journal of Machine Learning Research **17**(1), 43–93 (2016)

[32] Mahmoudi, M., Sapiro, G.: Fast image and video denoising via nonlocal means of similar neighborhoods. IEEE Signal Processing Letters **12**(12), 839–842 (2005)

[33] Nesterov, Y.: Lectures on Convex Optimization, Springer Optimization and Its Applications, vol. 137. Springer (2018)

[34] Niyogi, P., Smale, S., Weinberger, S.: Finding the homology of submanifolds with high confidence from random samples. Discrete & Computational Geometry **39**(1-3), 419–441 (2008)

[35] Pearson, K.: LIII. On lines and planes of closest fit to systems of points in space. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science **2**(11), 559–572 (1901)

[36] Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science **290**(5500), 2323–2326 (2000)

[37] Singer, A., Zhao, Z., Shkolnisky, Y., Hadani, R.: Viewing angle classification of cryo-electron microscopy images using eigenvectors. SIAM Journal on Imaging Sciences **4**(2), 723–759 (2011)

[38] Sober, B., Levin, D.: Manifold approximation by moving least-squares projection (MMLS). arXiv preprint arXiv:1606.07104 (2016)

[39] Starck, J.L., Candès, E.J., Donoho, D.L.: The curvelet transform for image denoising. IEEE Transactions on Image Processing **11**(6), 670–684 (2002)

[40] Su, Z.x., Li, Z.y., Cao, J.j., et al.: Curvature-aware simplification for point-sampled geometry. Journal of Zhejiang University SCIENCE C **12**(3), 184–194 (2011)

[41] Tenenbaum, J.B., De Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science **290**(5500), 2319–2323 (2000)

[42] Vardi, Y., Zhang, C.H.: The multivariate l1-median and associated data depth. Proceedings of the National Academy of Sciences **97**(4), 1423–1426 (2000)

[43] Woodruff, D.P., et al.: Sketching as a tool for numerical linear algebra. Foundations and Trends in Theoretical Computer Science **10**(1–2), 1–157 (2014)

[44] Yadav, S.K., Reitebuch, U., Skrodzki, M., Zimmermann, E., Polthier, K.: Constraint-based point set denoising using normal voting tensor and restricted quadratic error metrics. Computers & Graphics **74**, 234–243 (2018)