Linear-Projection Diffusion on Smooth Euclidean Submanifolds

Guy Wolf, Amir Averbuch*

School of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel

Abstract

To process massive high -dimensional datasets, we utilize the underlying assumption that data on manifold is approximately linear in sufficiently small patches (or neighborhoods of points) that are sampled with sufficient density from the manifold. Under this assumption, each patch can be represented (up to a small approximation error) by a tangent space of the manifold in its area and the tangential point of this tangent space.

We extend previously obtained results[1] for the finite construction of a linear-projection diffusion (LPD) super-kernel by exploring its properties when it becomes continuous. Specifically, its infinitesimal generator and the stochastic process defined by it are explored. We show that the resulting infinitesimal generator of this super-kernel converges to a natural extension of the original diffusion operator from scalar functions to vector fields. This operator is shown to be locally equivalent to a composition of linear projections between tangent spaces and the vector-Laplacians on them. We define a LPD process by using the LPD super-kernel as a transition operator while extending the process to be continuous. The obtained LPD process is demonstrated on a synthetic manifold.

Keywords: Diffusion maps, kernel method, manifold learning, stochastic processing, vector processing

1. Introduction

Massive high-dimensional datasets are very common nowadays in data analysis applications. Classical statistical methods cannot be applied to these datasets due to the "curse of dimensionality" phenomenon. Recent methods use manifolds to cope with this problem. Under this manifold assumption, a dataset is assumed to be sampled from a Euclidean submanifold with a relatively

*Corresponding author, Tel: +972-54-5694455, Fax: +972-3-6422020
Email address: amir@math.tau.ac.il (Amir Averbuch)

[1] M. Salhov, G. Wolf, and A. Averbuch. Patch-to-tensor embedding. *Applied and Computational Harmonic Analysis*, 2011.

small intrinsic dimension. The ambient high-dimensional Euclidean space of the manifold is defined by the original parameters of the dataset. These parameters are mapped via non-linear functions to low-dimensional coordinates of the manifold, which represent the independent factors that control the behaviors of the analyzed phenomenon.

Several methods have been suggested to provide a global coordinate system that represents the structure of an underlying manifold of a high-dimensional dataset. Kernel methods such as k-PCA [11, 14], Diffusion Maps [3] and geometric harmonics [4] have been used for this task. These methods are based on a kernel construction that introduces the notion of similarity, proximity, or affinity between data-points. Spectral analysis of this kernel is used to obtain an embedding of the data-points into an Euclidean space in a manner that preserves the qualities represented by the kernel.

Kernel methods extend two classical methods that uncover linear structures in datasets. These methods are the Principal Component Analysis (PCA) [8, 7] and the Multi-Dimensional Scaling (MDS) [5, 9]. The PCA method uses a covariance matrix between the parameters of the analyzed datasets and projects the data-points onto a space spanned by the most significant eigenvectors of this matrix. The MDS method uses the eigenvectors of a Gram matrix, which contains the inner-products between the points in the analyzed dataset, to define a mapping of data-points into an embedded space that preserves most of these inner-products. Both methods are equivalent. They represent data-points according to the directions in which most of the variance in the dataset is located.

Kernel methods aim at extending the essence of the MDS method by replacing the Gram matrix with a kernel matrix while preserving the qualities represented by it instead of the inner-products that are preserved by the MDS method. Some examples of these methods are LLE [12], Isomaps [17], Laplacian eigenmaps [1], Hessian eigenmaps [6], local tangent space alignment [18, 19] and diffusion maps [3]. These methods are also inspired from the theory of spectral graphs [2]. The defined kernel can be thought of as an adjacency matrix of a graph whose vertices are the points in the dataset. The analysis of the eigenvalues and the corresponding eigenvectors of this matrix can reveal many qualities and connections in the graph.

A fundamental, well-based, assumption of kernel methods in general, and diffusion maps in particular, is that, locally, the manifold is approximately linear in sufficiently small patches (or neighborhoods of points). Under this assumption, each patch can, in fact, be represented (up to a small approximation error) by a tangent space of the manifold in its area and the tangential point of this tangent space. Local PCA was suggested in [15, 16] to compute an approximation of suitable tangent spaces and their tangential points for patches that define neighborhoods of points that are sampled with sufficient density from the manifold.

Using the suggested representations, the relations between patches can be modeled by the usual affinity between tangential points and an operator that translates vectors from one tangent space to another. The structure of the ambient space was utilized in [13] to define linear-projection operators between tangent spaces and utilize them to construct a super-kernel that represents the affinity/similarity between patches. The structure of the underlying manifold was utilized for a similar purpose in [16] to define continuous parallel transport operators between tangent spaces and to define such a super-kernel by using discrete approximations of these operators. In fact, algorithmically, the approximations in [16] are achieved by orthogonalization of the linear-projection operators in [13]. Although these constructions differ by only a small modification of the construction algorithm, the resulting super-kernels have very different properties and different derived theories.

The construction of a super-kernel via orthogonal transformations between tangent spaces, which yield discrete approximations of the parallel transport operator on the underlying manifold, was utilized in [16] to define a vector diffusion map. The vector diffusion map is defined by the constructed super-kernel in a manner similar to the diffusion map, which is defined by the diffusion kernel. The infinitesimal generator of the super-kernel constructed in [16] converges to the connection-Laplacian on the manifold. A variation of this super-kernel was also utilized in [15] to encompass information about the orientability of the underlying manifold. When the manifold is orientable, the resulting orientable diffusion map gives an orientation over the manifold (in addition to the embedded coordinates). When it is not orientable, the double-cover of the manifold can be computed using this method.

In addition to defining the constructions of vector diffusion maps (VDM) and orientable diffusion maps (ODM), [16, 15] utilized a local-PCA process to approximate the tangent spaces that represent the analyzed patches on the manifold. The bounds of these approximation are thoroughly explored there and optimal values for the meta-parameters for this process are presented. In this paper, we assume that these tangent spaces can be approximated, e.g., by methods similar to the one described in [16, 15].

In this paper, we focus and extend the properties of linear-projection diffusion (LPD) super-kernels that were presented in [13]. These LPD super-kernels are a specific type of linear-projection super-kernels, whose spectra (i.e., eigenvalues) were shown to be non-negative. In case of LPD super-kernels, all the eigenvalues are between zero and one. This super-kernel was utilized in [13] to define an embedding of the patches on the manifold to a tensor space. The Frobenius distance between the coordinate matrices of the resulting tensors can be seen as an extension of the original diffusion distance, which was defined in [3]. This extension includes both the data about the proximities of tangential points in the diffusion process and the projections between the corresponding tangent spaces that represent the patches [13].

We extend the results obtained in [13] for finite constructions of LPD super-kernels by exploring its properties when it becomes continuous. We show that the resulting infinitesimal generator of this super-kernel converges to a natural extension of the original diffusion operator from scalar functions to vector fields. This operator is shown it be locally equivalent to a composition of linear projections between tangent spaces and the vector-Laplacians on them. We define a Linear-Projection Diffusion (LPD) process by using the LPD super-kernel as a transition operator while extending the process to be continuous.

The paper has the following structure. The manifold representation is defined in section 2. The original diffusion operator, the resulting diffusion map and a natural extension of the diffusion operator to work on vector-fields are presented in section 3. Section 4 describes the properties of the LPD diffusion operator. Specifically, its infinitesimal generator is explored in section 4.1 and the stochastic process defined by it is described in section 4.2. Finally, section 4.3 demonstrates the LPD process on a synthetic manifold.

2. Manifold representation

Let $\mathcal{M} \subseteq \mathbb{R}^m$ be a d-dimensional smooth Euclidean submanifold, which lies in the ambient space $\mathbb{R}^m$. For every point $x \in M$, the manifold $\mathcal{M}$ has a d-dimensional tangent space $T_x(\mathcal{M})$, which is a subspace of $\mathbb{R}^m$. Assume $o_x^1, \ldots, o_x^d \in T_x(\mathcal{M})$ form a d-dimensional orthonormal basis of $T_x(\mathcal{M})$. These d vectors can also be regarded as vectors in the ambient space $\mathbb{R}^m$, thus, we can represent them by using m coordinates by a basis of $\mathbb{R}^m$. Assume that $O_x \in \mathbb{R}^{m\times d}$ is a matrix whose columns are these vectors represented by the ambient coordinates

$$O_x \triangleq \begin{pmatrix} | & & | & & | \\ o_x^1 & \cdots & o_x^i & \cdots & o_x^d \\ | & & | & & | \end{pmatrix} \qquad x \in M. \tag{2.1}$$

We will assume from now on that vectors in $T_x(\mathcal{M})$ are expressed by their d coordinates according to the presented basis $o_x^1, \ldots, o_x^d$. For each vector $v \in T_x(\mathcal{M})$, the vector $\tilde{v} = O_x v \in \mathbb{R}^m$ is the same vector as v represented by m coordinates according to the basis of the ambient space. For each vector $u \in \mathbb{R}^m$ in the ambient space, the vector $u' = O_x^T u \in T_x(\mathcal{M})$ is the linear projection of u on the tangent space $T_x(\mathcal{M})$.

3. Diffusion Maps

The original diffusion maps method [3, 10] can be used to analyze the geometry of the manifold $\mathcal{M}$. This method is based on defining an isotropic kernel K as

$$k(x, y) \triangleq e^{-\frac{\|x-y\|}{\varepsilon}}, \qquad x, y \in M,$$

where ε is a meta-parameter of the algorithm. This kernel represents the affinities between points on the manifold. Next, a degree is defined for each point $x \in \mathcal{M}$ as $q(x) \triangleq \sum_{y\in M} k(x, y)$. Kernel normalization with this degree produces a stochastic transition operator P that is defined as $Pf(x) = \int f(y)p(x, y)dy$ for every function $f : \mathcal{M} \to \mathbb{R}$, where

$$p(x, y) = \frac{k(x, y)}{q(x)} \qquad x, y \in M, \tag{3.1}$$

which defines a Markov process (i.e., a diffusion process) over the points on the manifold $\mathcal{M}$.

The diffusion maps method computes an embedding of data points on the manifold into an Euclidean space whose dimensionality is usually significantly lower than the original data dimensionality. This embedding is a result of spectral analysis of the diffusion kernel. Thus, it is preferable to work with a symmetric conjugate to P, which is denoted by A and its elements are

$$a(x,y) = \frac{k(x,y)}{\sqrt{q(x)q(y)}} = \sqrt{q(x)}p(x,y)\frac{1}{\sqrt{q(y)}} \qquad x,y \in M. \tag{3.2}$$

We will refer to A as the diffusion affinity kernel or as the symmetric diffusion kernel. The eigenvalues $1 = \sigma_0 \geq \sigma_1 \geq \ldots$ of A and their corresponding eigenvectors $\psi_0, \psi_1, \ldots$ are used to construct the desired map, which embeds each data-point $x \in M$ onto the point $\Psi(x) = (\sigma_i \psi_i(x))_{i=0}^{\delta}$ for a sufficiently small δ, which is the dimension of the embedded space that depends on the decay of the spectrum of A. This construction is also known as the Laplacian of the graph constructed by the diffusion kernel [2].

3.1. Extended diffusion operator

The original diffusion kernel operates on scalar functions. Its extension to vector fields, which are expressed in local coordinates, is not trivial, since the local coordinates vary from point to point. However, in global coordinates (i.e., the coordinates of the ambient space $\mathbb{R}^m$), a simple extension can be defined as

$$\bar{P}\vec{\bar{\nu}}(x) = \int \vec{\bar{\nu}}(x)p(x,y)dy \ , \tag{3.3}$$

where $\vec{\bar{\nu}} : \mathcal{M} \to \mathbb{R}^m$ is a vector field expressed in the global coordinates of the ambient space. The relation between a tangent vector field $\vec{\nu} : \mathcal{M} \to \mathbb{R}^d$, expressed in local coordinates and its corresponding global vector field $\vec{\bar{\nu}} : \mathcal{M} \to \mathbb{R}^m$ is given by $\vec{\bar{\nu}}(x) = O_x \vec{\nu}(x)$, $x \in \mathcal{M}$. It should be noted that the vector field $\bar{P}\vec{\bar{\nu}}$, which results from the extended diffusion operator, is not necessarily a tangent vector field, i.e., the resulting vectors may not be tangent to the manifold at their assigned points.

While the operator $\bar{P}$ might not be useful by itself (since its operation does not result in a tangent vector field), it does allow us to extend the infinitesimal generator of the diffusion kernel in a meaningful way. The infinitesimal generator of the diffusion kernel is defined by $\mathcal{L}(P) \triangleq \lim_{\varepsilon \to 0} \frac{I-P}{\varepsilon}$, and it is shown in [3] that if the manifold has a uniform density, it satisfies $\mathcal{L}(P) = \Delta$, where Δ is the Laplace-Beltrami operator. If the density of the manifold is not uniform, a simple correction to the diffusion kernel can be used to maintain the same result. Therefore, for every function $f : \mathcal{M} \to \mathbb{R}$,

$$\mathcal{L}(P)f(x) \triangleq \lim_{\varepsilon \to 0} \frac{f(x) - Pf(x)}{\varepsilon} = \Delta f(x).$$

An extension of the described infinitesimal generator can be defined such that every vector field $\vec{\tilde{\nu}} : \mathcal{M} \to \mathbb{R}^m$ can be expressed in global coordinates of the ambient space to be

$$\mathcal{L}(\bar{P})\vec{\tilde{\nu}}(x) \triangleq \lim_{\varepsilon \to 0} \frac{\vec{\tilde{\nu}}(x) - \bar{P}\vec{\tilde{\nu}}(x)}{\varepsilon}.$$

The vector field $\vec{\tilde{\nu}}$ can be defined by using m scalar functions that determine its (global) coordinates at any point on the manifold, i.e. $\vec{\tilde{\nu}}(x) = (\tilde{\nu}_1(x), \ldots, \tilde{\nu}_m(x))^T$. By using these functions, the extended infinitesimal generator takes the form

$$\mathcal{L}(\bar{P})\vec{\tilde{\nu}}(x) = \begin{pmatrix} \lim_{\varepsilon \to 0} \frac{\tilde{\nu}_1(x) - P\tilde{\nu}_1(x)}{\varepsilon} \\ \vdots \\ \lim_{\varepsilon \to 0} \frac{\tilde{\nu}_m(x) - P\tilde{\nu}_m(x)}{\varepsilon} \end{pmatrix} = \begin{pmatrix} \Delta\tilde{\nu}_1(x) \\ \vdots \\ \Delta\tilde{\nu}_m(x) \end{pmatrix}, \tag{3.4}$$

which resembles the vector Laplacian in Cartesian coordinates, where the Laplace-Beltrami operator replaces the standard Laplacian on each coordinate. In this paper, we will show a different extension for the diffusion operator that uses the linear-projection operator to maintain the tangency of the vector fields on which this extension operates. We will show that the infinitesimal generator of both extensions are equivalent.

4. Linear-projection diffusion

In this section, we extend the original diffusion operator (Eq. 3.1). This extended operator is introduced in Definition 4.1, which uses both the scalar values from Eq. 3.1, which can be seen as transition probabilities between points on the manifold, and the linear-projection matrices between tangent spaces of the manifold, which are defined using the basis matrices from Eq. 2.1.

Definition 4.1 (LPD operator)**.** *Let $\vec{\nu} : \mathcal{M} \to \mathbb{R}^d$ be a smooth tangent vector field on $\mathcal{M}$ that assigns for each $x \in \mathcal{M}$ a vector $\vec{\nu}(x) \in T_x(\mathcal{M})$ represented in the d local coordinates of $T_x(\mathcal{M})$. A Linear-Projection Diffusion (LPD) operator G operates on such vector fields in the following way:*

$$G\vec{\nu}(x) = \int G_{xy}\vec{\nu}(y)dy,$$

where $G_{xy} \triangleq p(x,y)O_x^T O_y$, $x, y \in \mathcal{M}$.

The LPD operator in Definition 4.1 operates on tangent vector fields expressed in local coordinates (of the tangent spaces) and it results in tangent vector fields as well. Proposition 4.1 shows that this operator is independent of the global coordinates of the ambient space, i.e., it does not change under a change of basis of the ambient space. This is intuitively reasonable since the linear projections, which are used to define it, depend only on the relations between tangent spaces (i.e., their local bases) of the manifold and the scalar components from the diffusion operator (Eq. 3.1) depend only on distances between points on the manifold (and not on the coordinates used to express these points).

Proposition 4.1. *The LPD operator G is independent of the coordinates of the ambient space.*

Proof. Every change of basis in the ambient space $\mathbb{R}^m$ is represented and defined by an orthogonal $m \times m$ matrix. Assume that B is such a matrix. Assume that O_x, $x \in \mathcal{M}$, are the matrices from Eq. 2.1 expressed in the original basis. Then, in the new basis (i.e., after a change was made), they are expressed by BO_x, $x \in \mathcal{M}$, thus, under the new basis we have

$$\begin{aligned} G_{xy} = p(x,y)(BO_x)^T(BO_y) &= p(x,y)O_x^T B^T B O_y \\ &= p(x,y)O_x^T O_y \qquad x, y \in \mathcal{M}, \end{aligned} \tag{4.1}$$

where the $d \times d$ matrices G_{xy} are used in Definition 4.1 to represent the LPD operator G. Therefore, the LPD operator G does not change under a change of basis of the ambient space. In fact, it is expressed by the same matrices G_{xy}, $x, y \in \mathcal{M}$, in every basis of the ambient space. $\square$

A symmetric Linear-Projection Diffusion (LPD) super-kernel was constructed in [13] for a finite dataset of points sampled from an Euclidean submanifold. In the finite case, this super-kernel was a block matrix, where each block was defined by the diffusion affinities (see Eq. 3.2 in *this* paper) and linear-projection matrices between the tangent spaces of the manifold. An extension of this construction to the continuous case is given by the symmetric conjugate $\hat{G}$ of the LPD operator. The continuous LPD super-kernel $\hat{G}$ is defined by its operation on the tangent vector field $\vec{\nu} : \mathcal{M} \to \mathbb{R}^d$ as $\hat{G}\vec{\nu}(x) = \int \hat{G}_{xy}\vec{\nu}(y)dy$, where

$$\begin{aligned} \hat{G}_{xy} &= a(x,y)O_x^T O_y = \sqrt{q(x)}p(x,y)O_x^T O_y \frac{1}{\sqrt{q(y)}} \\ &= \sqrt{q(x)}G_{xy}\frac{1}{\sqrt{q(y)}} \qquad x, y \in \mathcal{M}. \end{aligned} \tag{4.2}$$

Therefore, the relation between the LPD operator G and the LPD super-kernel $\hat{G}$ is similar to the one between the diffusion operator P (Eq. 3.1) and the diffusion affinity kernel A (Eq. 3.2).

While the eigen-decompositions of operators that operate on a vector fields are not well-studied as the ones of operators that operate on scalar functions, in the finite case these operators become block matrices and their eigen-decompositions follow from these matrices. As a block matrix, the LPD super-kernel is positive semi-definite and all its eigenvalues are between zero and one. Since the LPD super-kernel is symmetric conjugate of the LPD operator, then, in the finite case, the spectrum of the LPD operator is also between zero and one. Therefore, for all practical purposes, the LPD operator, which is presented in this paper, can be treated as positive semi-definite with all of its eigenvalues less than or equal to one. The eigenvectors of the LPD operator and the LPD super-kernel are related via conjugation in a manner similar to the relation between the original diffusion operator and the affinity kernel. The reader is referred to [3] for more information about these relations in the original diffusion-maps case.

4.1. Infinitesimal generator

This section is devoted to the infinitesimal generator study of the LPD operator presented in Definition 4.1. Theorem 4.2 shows that this infinitesimal generator is equivalent to that of the extended diffusion operator presented in Section 3.1 (specifically, by Eq. 3.3). Corollary 4.3 uses this result to explain the resulting operator in terms of vector-Laplacian operators on the tangent spaces of the manifold.

Theorem 4.2. *Let $\bar{P}$ be an extended diffusion operator (Eq. 3.3), G be a LPD operator (Definition 4.1) and $\mathcal{L}(\bar{P})$ and $\mathcal{L}(G)$ be the infinitesimal generators of these operators. In addition, let $\vec{\nu}$ be a vector field expressed in the local coordinates of the tangent spaces and let $\vec{\bar{\nu}}$ be the same vector field expressed in global coordinates. Then,*

$$\mathcal{L}(\bar{P})\vec{\bar{\nu}}(x) = O_x \mathcal{L}(G)\vec{\nu}(x) \qquad x \in \mathcal{M},$$

where the matrices O_x are defined in Eq. 2.1, i.e., the infinitesimal generators of $\bar{P}$ and G are equivalent, where $\bar{P}$ and G operate in global and in local coordinates, respectively.

Proof. The infinitesimal generator of $\bar{P}$ is $\mathcal{L}(\bar{P}) = \lim_{\varepsilon\to 0}(I - \bar{P})/\varepsilon$. Let $x \in \mathcal{M}$ be an arbitrary point on the manifold, then

$$\mathcal{L}(\bar{P})\vec{\bar{\nu}}(x) = \lim_{\varepsilon\to 0} \frac{\vec{\bar{\nu}}(x) - \bar{P}\vec{\bar{\nu}}(x)}{\varepsilon}, \tag{4.3}$$

where, by definition, $\bar{P}\vec{\bar{\nu}}(x) = \int \vec{\bar{\nu}}(y)p(x,y)dy$.

Since the tangent space $T_x(\mathcal{M})$ is a subspace of the ambient space $\mathbb{R}^m$, every vector $\vec{\bar{\nu}}(y) \in \mathbb{R}^m$, $y \in \mathcal{M}$, can be expressed as the sum of a vector on the subspace $T_x(\mathcal{M})$ and by a orthogonal vector to it. In other words, for every $y \in \mathcal{M}$, we can define a vector $\vec{\bar{\nu}}_\top(y) \in T_x(\mathcal{M})$ (expressed in the global coordinates of the ambient space) and a vector $\vec{\bar{\nu}}_\perp(y) \perp T_x(\mathcal{M})$ such that $\vec{\bar{\nu}}(y) = \vec{\bar{\nu}}_\top(y) + \vec{\bar{\nu}}_\perp(y)$. Therefore, Eq. 4.3 can be rewritten as

$$\mathcal{L}(\bar{P})\vec{\bar{\nu}}(x) = \lim_{\varepsilon\to 0} \frac{\vec{\bar{\nu}}(x) - \int(\vec{\bar{\nu}}_\top(y) + \vec{\bar{\nu}}_\perp(y))p(x,y)dy}{\varepsilon}. \tag{4.4}$$

Since $T_x(\mathcal{M})$ is a d-dimensional subspace of $\mathbb{R}^m$, its basis $o_x^1, \ldots, o_x^d$ can be treated as an orthonormal set of d vectors in $\mathbb{R}^m$. As such, this set can be expanded with the $m-d$ additional orthonormal vectors $b_x^1, \ldots, b_x^{m-d} \perp T_x(\mathcal{M})$

to form a basis for $\mathbb{R}^m$. Every vector in $\mathbb{R}^m$ can be expressed by m coordinates $c_1, \dots, c_m \in \mathbb{R}$ where $c_1, \dots, c_d$ correspond to $o_x^1, \dots, o_x^d$ and $c_{d+1}, \dots, c_m$ correspond to $b_x^1, \dots, b_x^{m-d}$. Thus, a vector in $T_x(\mathcal{M})$ has $c_{d+1} = c_{d+2} \dots = c_m = 0$, while a vector orthogonal to $T_x(\mathcal{M})$ has $c_1 = c_2 \dots = c_d = 0$. From here on we will assume w.l.o.g. that these coordinates are the global coordinates used to express the vectors in the ambient space.

According to the presented coordinate system, the vectors of $\vec{\tilde{\nu}}$ take the form

$$\vec{\tilde{\nu}} = \begin{pmatrix} \tilde{\nu}_1(y) \\ \vdots \\ \tilde{\nu}_d(y) \\ \tilde{\nu}_{d+1}(y) \\ \vdots \\ \tilde{\nu}_m(y) \end{pmatrix} = \begin{pmatrix} \tilde{\nu}_1(y) \\ \vdots \\ \tilde{\nu}_d(y) \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \tilde{\nu}_{d+1}(y) \\ \vdots \\ \tilde{\nu}_m(y) \end{pmatrix} = \vec{\tilde{\nu}}_\top(y) + \vec{\tilde{\nu}}_\perp(y) \qquad y \in \mathcal{M},$$

where $\tilde{\nu}_1, \dots, \tilde{\nu}_m$ are the coordinate functions of the vector field $\vec{\tilde{\nu}}$ according to this system. Thus, we get

$$\int \vec{\tilde{\nu}}_\perp(y) p(x,y) dy = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \int \tilde{\nu}_{d+1}(y) p(x,y) dy \\ \vdots \\ \int \tilde{\nu}_m(y) p(x,y) dy \end{pmatrix}. \tag{4.5}$$

Let us examine one of the nonzero coordinates in the vector $\xi \in \{d{+}1, \dots, m\}$ (Eq. 4.5). The function $\tilde{\nu}_\xi$ is a scalar function. Thus, the results shown in [3] can be applied to it. Specifically, the integration $\int \tilde{\nu}_\xi(y) p(x,y) dy$ over the entire manifold can be approximated $\int_{\|y-x\|<\varepsilon} \tilde{\nu}_\xi(y) p(x,y) dy$ on an open ball of radius ε around x on the manifold (i.e., the distance $\|y - x\|$ is a geodesic distance). Also, for a small enough ε, all the points in this ball are in the same coordinate neighborhood, where their coordinates can be expressed by the d orthogonal geodesics $s_1, \dots, s_d$ that meet at x. Every point $y \in \mathcal{M}$ in this ball (i.e., $\|y - x\| < \varepsilon$ in terms of geodesic distances) can be represented by a vector $\vec{s^y} = (s_1^y, \dots, s_d^y)$ such that $\|\vec{s^y}\| < \varepsilon$. By using this representation, we can apply Taylor expansion in this ball to the function $\tilde{\nu}_\xi$ to get

$$\tilde{\nu}_\xi(y) = \tilde{\nu}_\xi(x) + \sum_{j=1}^{d} \frac{\partial \tilde{\nu}_\xi}{\partial s_j} s_j^y + \sum_{i=1}^{d} \sum_{j=1}^{d} \frac{\partial^2 \tilde{\nu}_\xi}{\partial s_i \partial s_j} s_i^y s_j^y + \dots \qquad \|s^y\| < \varepsilon, y \in \mathcal{M}. \tag{4.6}$$

Since $\vec{\tilde{\nu}}(x) \in T_x(\mathcal{M})$, the orthogonal component $\vec{\tilde{\nu}}_\perp(x)$ is zero, thus $\tilde{\nu}_\xi(x) = 0$ and this term is canceled in Eq. 4.6.

We combine the above arguments to get

$$\int \tilde{\nu}_\xi(y)p(x,y)dy \approx \sum_{j=1}^{d} \frac{\partial \tilde{\nu}_\xi}{\partial_j s_j} \int s_j^y p(x,y)dy + \sum_{i=1}^{d}\sum_{j=1}^{d} \frac{\partial^2 \tilde{\nu}_\xi}{\partial s_i \partial s_j} \int s_i^y s_j^y p(x,y)dy,$$

and by canceling odd terms we get

$$\int \tilde{\nu}_\xi(y)p(x,y)dy \approx \sum_{i=1}^{d} \frac{\partial^2 \tilde{\nu}_\xi}{\partial s_i^2} \int (s_i^y)^2 p(x,y)dy.$$

According to [3], the approximation error of this calculation is of order ε^2, or higher powers of ε, for a small enough meta-parameter $1 > \varepsilon > 0$. In addition, since the integration can be taken to be within an open ball of radius ε, we can have $(s_i^y)^2 < \varepsilon^2$, thus, for a small $1 > \varepsilon > 0$,

$$\left| \frac{\int \tilde{\nu}_\xi(y)p(x,y)dy}{\varepsilon} \right| \leq \left| \frac{\varepsilon^2 \gamma}{\varepsilon} \right| = |\varepsilon\gamma| \,, \tag{4.7}$$

where γ is the sum $\sum_{i=1}^{d} \frac{\partial^2 \tilde{\nu}_\xi}{\partial_i s_i^2}$ which is a suitable constant coefficient for bounding the approximation error.

Combining Eqs. 4.5 and 4.7 we get

$$\left\| \frac{\int \vec{\tilde{\nu}}_\perp(y)p(x,y)dy}{\varepsilon} \right\| \leq \sqrt{m-d} \cdot |\varepsilon\gamma| \,.$$

Thus, when $\varepsilon \to 0$, the length of the vector $\int \vec{\tilde{\nu}}_\perp(y)p(x,y)dy/\varepsilon$ becomes zero. Therefore, by using Eq. 4.4 we get

$$\begin{aligned} \mathcal{L}(\bar{P})\vec{\tilde{\nu}}(x) &= \lim_{\varepsilon\to 0} \frac{\vec{\tilde{\nu}}(x) - \int \vec{\tilde{\nu}}_\top(y)p(x,y)dy}{\varepsilon} + \frac{\int \vec{\tilde{\nu}}_\perp(y)p(x,y)dy}{\varepsilon} \\ &= \lim_{\varepsilon\to 0} \frac{\vec{\tilde{\nu}}(x) - \int \vec{\tilde{\nu}}_\top(y)p(x,y)dy}{\varepsilon}. \end{aligned} \tag{4.8}$$

Finally, we notice that for $y \in \mathcal{M}$, the vector $\vec{\tilde{\nu}}_\top(y)$ is in fact the projection of $\vec{\tilde{\nu}}(y)$ on $T_x(\mathcal{M})$ expressed in the global coordinates of the ambient space, which is given by the matrix $O_x O_x^T$. Also, the relation between $\vec{\nu}$ and $\vec{\tilde{\nu}}$, which expresses the same vector field in local and global coordinates respectively, is given by $\vec{\tilde{\nu}}(y) = O_y \vec{\nu}(y)$ for every $y \in \mathcal{M}$. Therefore, we have

$$\begin{aligned} \mathcal{L}(\bar{P})\vec{\tilde{\nu}}(x) &= \lim_{\varepsilon\to 0} \frac{\vec{\tilde{\nu}}(x) - \int \vec{\tilde{\nu}}_\top(y)p(x,y)dy}{\varepsilon} \\ &= \lim_{\varepsilon\to 0} \frac{O_x\vec{\nu}(x) - \int O_x O_x^T O_y \vec{\nu}(y)p(x,y)dy}{\varepsilon} \\ &= O_x \lim_{\varepsilon\to 0} \frac{\vec{\nu}(x) - G\vec{\nu}(x)}{\varepsilon} = \mathcal{L}(G)\vec{\nu}(x), \end{aligned}$$

and since $x \in \mathcal{M}$ was chosen arbitrarily, the equality is satisfied at every point on the manifold and the theorem is proved. $\square$

Intuitively, the infinitesimal generator of an operator such as $\bar{P}$ or G considers the effects of the operator on the values of vector fields (i.e., vector-valued functions) in infinitesimal neighborhoods on the manifold. In the case of $\bar{P}$, the ambient directions of the vectors are not changed by the operator and the measured effects are determined by the scalar affinities (i.e., transition probabilities). The LPD operator G, however, also projects the vectors on the corresponding tangent spaces of the manifold so the vector field remains tangent to it. Thus, in the case of G, the measured effects are determined by both the scalar affinities (i.e., transition probabilities) and the curvatures of the manifold, which are intuitively manifested as the angles between its tangent spaces in the considered area. Thus, the difference between the two infinitesimal generators comes from the curvature component in the latter case. However, as the proof of Theorem 4.2 shows, when only an infinitesimal area is considered, the manifold converges to its locally-linear nature and the distances (i.e., angles) between the tangent spaces of the manifold diminish and converge to zero.

This result also gives an insight into the role of the scaling meta-parameter ε in the LPD construction. In scalar diffusion maps (and in the extended diffusion operator $\bar{P}$) it controls the sizes of the considered neighborhoods. In the LPD operator, it controls both these sizes and the effects of the curvatures that are taken in consideration. Smaller sizes of ε consider smaller neighborhoods and less affect of the curvatures, and when $\varepsilon \to 0$, neighborhoods converge to single points and the effects of the curvatures are canceled.

Theorem 4.2 shows that the LPD operator G maintains the same infinitesimal generator as the extended diffusion operator $\bar{P}$ while operating in local coordinates instead of global ones. This result shows that the LPD construction maintains, to some degree, the infinitesimal behavior (or nature) of the original and of the extended diffusion operators. In the scalar case, the infinitesimal generator of the diffusion operator can be expressed by Laplace operators (specifically, the graph Laplacian and the Laplace-Beltrami operator on manifolds). Corollary 4.3 utilizes the relation shown in Theorem 4.2 to provide an expression for the resulting infinitesimal generator using the vector-Laplacian, which extents the Laplacian from scalar functions to vector fields.

Corollary 4.3. *Let G be the LPD operator with the infinitesimal generator $\mathcal{L}(G)$. Let $\vec{\nu}$ be a tangent vector field expressed by the local coordinates of the tangent spaces of the manifold $\mathcal{M}$. Then,*

$$\mathcal{L}(G)\vec{\nu}(x) = \bar{\Delta}(\mathrm{proj}_x \vec{\nu})(x) \qquad x \in \mathcal{M},$$

where the operator proj_x projects a vector field on the tangent space $T_x(\mathcal{M})$, and $\bar{\Delta}$ is the vector-Laplacian on this tangent space.

Proof. Let $x \in \mathcal{M}$ be an arbitrary point on the manifold and let $\vec{\tilde{\nu}}$ expresses the tangent vector field $\vec{\nu}$ by the ambient coordinates resulting from expanding the basis $o_x^1, \ldots, o_x^d$ of the tangent space $T_x(\mathcal{M})$ with additional $m-d$ orthonormal vectors, as was explained in the proof of Theorem 4.2. Let $\tilde{\nu}_1, \ldots, \tilde{\nu}_m$ be the coordinate functions of $\vec{\tilde{\nu}}$, where the first d vectors correspond to $o_x^1, \ldots, o_x^d$ and the rest correspond to the other $m-d$ vectors, which are orthogonal to $T_x(\mathcal{M})$. The projection of the vector field on $T_x(\mathcal{M})$ can now be written as

$$\operatorname{proj}_x \vec{\nu}(y) = (\tilde{\nu}_1(y), \ldots, \tilde{\nu}_d(y))^T \qquad y \in \mathcal{M}. \tag{4.9}$$

According to Theorem 4.2, we have

$$O_x^T \mathcal{L}(\bar{P})\vec{\bar{\nu}}(x) = O_x^T O_x \mathcal{L}(G)\vec{\tilde{\nu}}(x) = \mathcal{L}(G)\vec{\tilde{\nu}}(x)$$

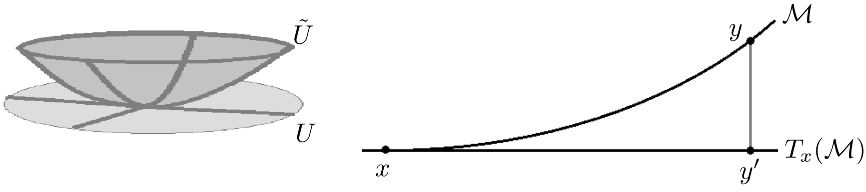
and by using Eq. 3.4 we get

$$\mathcal{L}(G)\vec{\tilde{\nu}}(x) = (\Delta\tilde{\nu}_1(x), \ldots, \Delta\tilde{\nu}_d(x))^T, \tag{4.10}$$

where Δ is regarded as the Laplace-Beltrami operator (at x) on the manifold or, equivalently, the Laplacian in a small open set on $T_x(\mathcal{M})$ around x whose points are related to those on the manifold via the exponential map. Using the second interpretation and by recalling Eq. 4.9, the right-hand side expression in Eq. 4.10 is in fact the vector-Laplacian (in Cartesian coordinates of $T_x(\mathcal{M})$) at x of the projection of the vector field $\vec{\nu}$ on the tangent space $T_x(\mathcal{M})$, as stated in the corollary. □

4.2. Stochastic diffusion process

In this section, we define the Brownian motion (BM) on a d-dimensional manifold in $\mathbb{R}^m$, $d \ll m$. Let $\tilde{u} : \mathbb{R} \to \mathcal{M} \subseteq \mathbb{R}^m$ be a stochastic process on the manifold $\mathcal{M}$, such that at time t_0 the process is at $x = \tilde{u}(t_0) \in \mathcal{M}$. The d-dimensional manifold is defined locally at each point $x \in \mathcal{M}$. Let $\tilde{U} \subset \mathcal{M}$ be a sufficiently small open set around x defined as $\tilde{U} = \{z \in \mathcal{M} | \, \|x - z\| < \zeta\}$ for a small ζ. We can choose ζ to be sufficiently small such that all the points in $\tilde{U}$ have the same coordinate neighborhood in $\mathcal{M}$, and furthermore we can set it so that the coordinates in $\tilde{U}$ are given by the bijective exponential map $\exp_x : U \to \tilde{U}$, where $U \subseteq T_x(\mathcal{M})$ is the projection of $\tilde{U}$ on the tangent space $T_x(\mathcal{M})$ (see Fig. 4.1). Let $\Delta t \in \mathbb{M}$ be sufficiently small such that almost surely $\tilde{u}(t) \in \tilde{U}$ for every $t \in (t_0 - \Delta t, t_0 + \Delta t)$. Therefore, the stochastic process can



(a) The set $\tilde{U} \subset \mathcal{M}$ and its projection $U \subset T_x(\mathcal{M})$ on the tangent space of the manifold $x \in \mathcal{M}$.

(b) The exponential map $\exp_x$ maps the points $y \in \tilde{U} \subset \mathcal{M}$ and $y' \in U \subset T_x(\mathcal{M})$ to each other.

Figure 4.1: An illustration of an open set $\tilde{U} \subseteq \mathcal{M}$ around $x \in \mathcal{M}$, its projection $U \subseteq T_x(\mathcal{M})$ on the tangent space $T_x(\mathcal{M})$, and the exponential map $\exp_x$, which maps each point $y \in \tilde{U}$ on the manifold to $y' \in U$ on the tangent space $T_x(\mathcal{M})$.

be expressed, in the time segment $\mathcal{T}_{\Delta t}(t_0) = (t_0 - \Delta t, t_0 + \Delta t)$, by the local coordinates of U, i.e., we define the process $u : \mathcal{T}_{\Delta t}(t_0) \to U$ such that for each $t \in \mathcal{T}_{\Delta t}(t_0)$, it satisfies $\exp_x(u(t)) = \tilde{u}(t)$.

We define the Brownian trajectories of the local process u (and thus its global version $\tilde{u}$) by

$$u(t_0 + \tau) \triangleq u(t_0) + \Delta u(\tau) \in T_x(\mathcal{M}) \qquad |\tau| < \Delta t, \tag{4.11}$$

where the transition vector $\Delta u \in T_x(\mathcal{M})$ is d-dimensional stochastic vector given by

$$\Delta u(\tau) \triangleq B \Delta w \qquad |\tau| < \Delta t, \tag{4.12}$$

where $\Delta w \sim \mathcal{N}(0, \tau I)$ is a vector of d i.i.d. normal zero-mean random variables with variance τ, and B is a $d \times d$ diffusion coefficients matrix. To stay on $T_x\mathcal{M}$, the vector $\Delta u(\tau)$ has to satisfy the orthogonality condition $\langle \Delta u(\tau), \vec{n}(x) \rangle = 0$ where $\vec{n}(x)$ is the m-dimensional unit normal to $T_x(\mathcal{M})$.

The global process $\tilde{u}$ can be discretized by setting a time unit $\bar{\tau} < \Delta t$ and expressing the transition probabilities from $x = \tilde{u}(t_0)$ to each possible $y = \tilde{u}(t_0 + \bar{\tau})$ by a probability distribution function $p_x : \mathcal{M} \to [0, 1]$. According to our choice of Δt, almost surely $\tilde{u}(t_0 + \bar{\tau}) \in \tilde{U}$ and therefore, since there is a bijection between U and $\tilde{U}$ (i.e., the exponential map $\exp_x$), a restriction of p_x to $\tilde{U}$ should yield the transition probabilities of the local process u. In fact, the row-stochastic diffusion operator P (Eq. 3.1), with a suitable meta-parameter ε, defines such probability distributions by setting $p_x(\cdot) = p(x, \cdot)$ [3, 10].

The processes u and $\tilde{u}$ represent transitions between points on the manifold. However, while the Brownian trajectories defined by $\tilde{u}$ give points on the manifold itself, the points on Brownian trajectories defined by u are just approximations that lie on the tangent $T_x(\mathcal{M})$ (see Fig. 4.1). The exponential map $\exp_x$ "raises" these approximations to lie on the manifold, thus providing the bijective relation between the local process u and the global process $\tilde{u}$. It was shown in [3] that in a sufficiently small neighborhood around x, all quantities concerning the diffusion operator in Eq. 3.1, and the resulting diffusion process, can be expressed in terms of the tangent space $T_x(\mathcal{M})$. This representation entails an infinitesimal approximation error that is canceled when the process becomes continuous in the limit $\bar{\tau} \to 0$ ($\varepsilon \to 0$ in terms of the diffusion operator). This result justifies our definition of the process $\tilde{u}$ via its local approximation u.

Further justification comes from examining the difference between the point $y = \tilde{u}(t_1)$ and its tangential approximation $y' = u(t_1)$, $t_1 = t_0 + \bar{\tau}$. Since $x, y' \in T_x(\mathcal{M})$, we have $y' - x \in T_x(\mathcal{M})$, and since $y - y' = (y - x) - (y' - x)$ we get that

$$y = y' + \rho \vec{n}(x), \tag{4.13}$$

where $\vec{n}(x) \in \mathbb{R}^m$ is the normal of $T_x(\mathcal{M})$ (as a subspace of $\mathbb{R}^m$). Since the difference is in a direction orthogonal to the tangent space $T_x(\mathcal{M})$, it is bounded by the distance between x and y, and by the angle θ between the tangent space $T_x(\mathcal{M})$ and the vector $y - x$, which goes from x to y. The distance between x

and y is bound by the radius ζ of $\tilde{U}$, which can be chosen to be as infinitesimally small (as long as Δt is set accordingly). Also, as y gets infinitesimally close to x, the angle between the vector $y - x$ and the tangent $T_x(\mathcal{M})$ vanishes, where the rate of the decrement is given by the curvature of the manifold $\mathcal{M}$ around x. Therefore, both error terms are canceled when the process becomes continuous (i.e., by taking $\zeta \to 0$, $\Delta t \to 0$ and $\bar{\tau} \to 0$).

Equation 4.13 has $d+1$ unknowns: the d local coordinates of $u(t_1)$ and ρ, which is the Euclidean distance from $y' \in T_x(\mathcal{M})$ to $y \in \mathcal{M}$ - see Fig. 4.1. To solve the system (4.13), we linearize it locally by setting $u(t_0 + \bar{\tau}) = u(t_0) + \Delta u$ and expanding everything to leading order in $\bar{\tau}$. We obtain

$$\tilde{u}(t_0) + \left.\frac{\partial \tilde{u}}{\partial u}\right|_{u=u(t_0)} \Delta u + \rho \vec{n}(x) = y + O(\Delta u^2) + O(\rho \Delta u),$$

and by using Eqs. 4.11 and 4.12, we get

$$\left.\frac{\partial \tilde{u}}{\partial u}\right|_{u=u(t_0)} \Delta u + \rho \vec{n}(x) = B\Delta w + O(\Delta u^2) + O(\rho \Delta u). \tag{4.14}$$

The system (4.14) consists of m linear equations for the $d+1$ unknowns Δu and ρ. The term $\rho \vec{n}(x)$ can be dropped, because $\rho \ll |\Delta \tilde{u}(t_1)|$. For $m = d+1$ the Euler scheme for the BM on $\mathcal{M}$ becomes

$$u(t + \bar{\tau}) = u(t_0) + \left[\frac{\partial \tilde{u}}{\partial u}\right]^{-1} B\Delta w. \tag{4.15}$$

In the limit $\bar{\tau} \to 0$, $\tilde{u}(t_1)$ converges in the usual way to a continuous trajectory on $\mathcal{M}$. The PDF of $\tilde{u}(t_1)$ satisfies the Laplace-Beltrami equation [3, 10] on $\mathcal{M}$.

In addition to the processes defined by u and $\tilde{u}$, which govern the movement from one point to another on the manifold $\mathcal{M}$, we define the vector functions $\vec{v} : \mathcal{T}_{\Delta t}(t_0) \to T_x(\mathcal{M})$ and $\vec{\tilde{v}} : \mathbb{R} \to \mathbb{R}^d$ that define the propagation of a vector along the route determined by the diffusion process. Let $\vec{v}_x = \vec{v}(t_0) \in T_x(\mathcal{M})$ be a tangent vector attached to the diffusion process at x in time t_0. In the discrete case, when the diffusion advances from time t_0 to time $t_1 = t_0 + \bar{\tau}$, it goes from $x = u(t_0)$ (since the tangential point x is on both $\mathcal{M}$ and $T_x(\mathcal{M})$) to $y' = u(t_1)$. Since this step is done entirely in the tangent space $T_x(\mathcal{M})$, we can propagate the vector $\vec{v}(t_0)$ to $\vec{v}(t_1) = \vec{v}(t_0)$ without change, and thus we attach the same vector $\vec{v}_x = \vec{v}_y \in T_x(\mathcal{M})$ to the point $y' \in T_x(\mathcal{M})$. However, when we move back to the manifold using the exponential map to get $y = \tilde{u}(t_1) = \exp_x(y') \in \mathcal{M}$, this vector cannot be directly propagated as-is to $y \in \mathcal{M}$ since $\vec{v}_y \notin T_y(\mathcal{M})$ (unless the manifold is flat). To deal with this problem, we use the linear projection operator $O_y^T O_x$ and define $\vec{\tilde{v}}_y = \vec{\tilde{v}}(t_1) = O_y^T O_x \vec{v}(t_1)$. Thus, at time t_1, the vector $\vec{\tilde{v}}(t_1)$ consists of d-coordinates that represent the closest vector in $T_y(\mathcal{M})$ to $\vec{v}(t_1)$.

The linear projection, which is used to transform the vector $\vec{v}_y$ to $\vec{\tilde{v}}_y$, does not preserve the length of the vector. In fact, the resulting vector becomes shorter. Eventually, at $t \to \infty$, the vectors, which were propagated by this discrete process, will converge to 0. However, this is only a property of the discretization and not of the continuous case. Since $\vec{\tilde{v}}_y$ is the projection of $\vec{v}_y$ on $T_y(\mathcal{M})$, then $\left\|\vec{\tilde{v}}_y\right\| = \|\vec{v}_y\| \cos\theta$, where θ is the angle between $\vec{v}_y$ and $T_y(\mathcal{M})$. Also, θ is bounded from above by the angle between the tangent spaces $T_x(\mathcal{M})$ and $T_y(\mathcal{M})$. Therefore, smaller angles between tangent spaces yield less decrement of the length by the projection. In the continuous case, we can take y to be infinitesimally close to x. Therefore, the angle between their tangent spaces $T_x(\mathcal{M})$ and $T_y(\mathcal{M})$ gets infinitesimally small (where the rate of the decrement is given by the curvature of the manifold $\mathcal{M}$), thus, $\theta \to 0$ and $\left\|\vec{\tilde{v}}_y\right\| \to \|\vec{v}_y\|$.

The discussion presented in this section is summarized in Proposition 4.4. The proof is straight forward from this discussion. The transitions performed by the process described in this section and in Proposition 4.4, are illustrated in Fig. 4.2.

Proposition 4.4. *Let G be a LPD operator with a sufficiently small ε such that if $x, y \in \mathcal{M}$ are not in the same neighborhood then $p(x, y) \approx 0$ with infinitesimal approximation error. The operator G is a transition operator of a discrete stochastic process that propagates vectors along the manifold. Each $d \times d$ block $G_{xy} = p(x, y)O_x^T O_y$ describes a transition from the tangent vector $v_x \in T_x(\mathcal{M})$ based at $x \in \mathcal{M}$ to the vector $v_y \in T_y(\mathcal{M})$ based at $y \in \mathcal{M}$. This discrete transition is done by the following steps:*

1. *A destination point $y \in \mathcal{M}$ is randomly chosen with probability $p(x, y)$;*
2. *The direction and the length of the transition are represented by a vector $\vec{u}_{x\to y} \in T_x(\mathcal{M})$ from x to the projection of y on $T_x(\mathcal{M})$;*

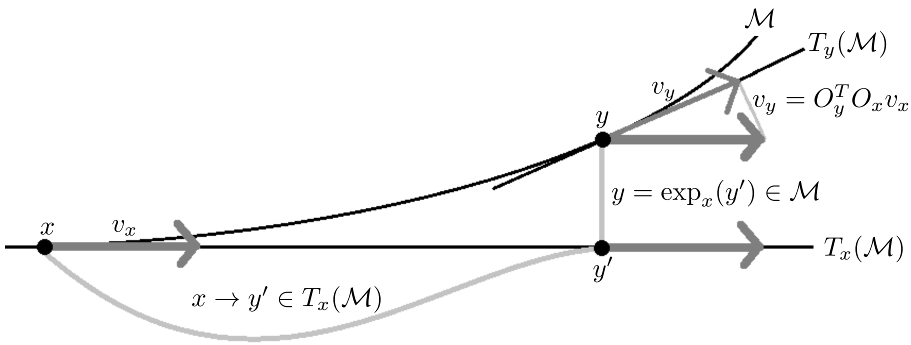


Figure 4.2: The "jump" of the LPD discrete process goes from time t_0 to time $t_1 = t_0 + \bar{\tau}$. The jump starts with a vector $v_x \in T_x(\mathcal{M})$ that is attached to the manifold at $x \in \mathcal{M}$. First, a point $y' \in T_x(\mathcal{M})$ is chosen according to the transition probabilities of the diffusion operator P (Eq. 3.1). Then, the exponential map is used to translate this point to a point $y \in \mathcal{M}$ on the manifold. Finally, the vector $v_x \in T_x(\mathcal{M})$ is projected to $v_y \in T_y(\mathcal{M})$ and attached to the manifold at $y \in \mathcal{M}$.

3. *The vector* $\vec{u}_{x\to y}$ *and the exponential map around* x *are used to perform the transition to*

$$y = \exp_x(x + \vec{u}_{x\to y}) = x + \vec{u}_{x\to y} + \rho\vec{n}(x) ,$$

where $\vec{n}(x)$ *is the normal of* $T_x(\mathcal{M})$ *in* $\mathbb{R}^m$ *and* $\rho \in \mathbb{M}$ *is the distance of the projection from the manifold;*
4. *The vector* v_x *(treated as a column vector) is projected on* $T_y(\mathcal{M})$ *to get* $v_y^T = v_x^T O_x^T O_y$*, thus*

$$v_y = v_x - \eta\vec{n}(y) ,$$

where $\vec{n}(y)$ *is the normal of* $T_y(\mathcal{M})$ *in* $\mathbb{R}^m$ *and* $\eta \in \mathbb{M}$ *is determined by the length of* v_x *and the angle it makes with* $T_y(\mathcal{M})$*;*
5. *The transition ends with the achieved tangent vector* $v_y \in T_y(\mathcal{M})$ *at* $y \in \mathcal{M}$*.*

As the process becomes continuous, $\varepsilon \to 0$ *and so does* $\rho \to 0$ *and* $\eta \to 0$*, and the process remains on the manifold.*

4.3. Linear-projection diffusion process demonstration

To demonstrate the stochastic process described in section 4.2, we implemented it on a two-dimensional paraboloid lying in a three dimensional Euclidean ambient space. We sampled 8101 points from the paraboloid defined

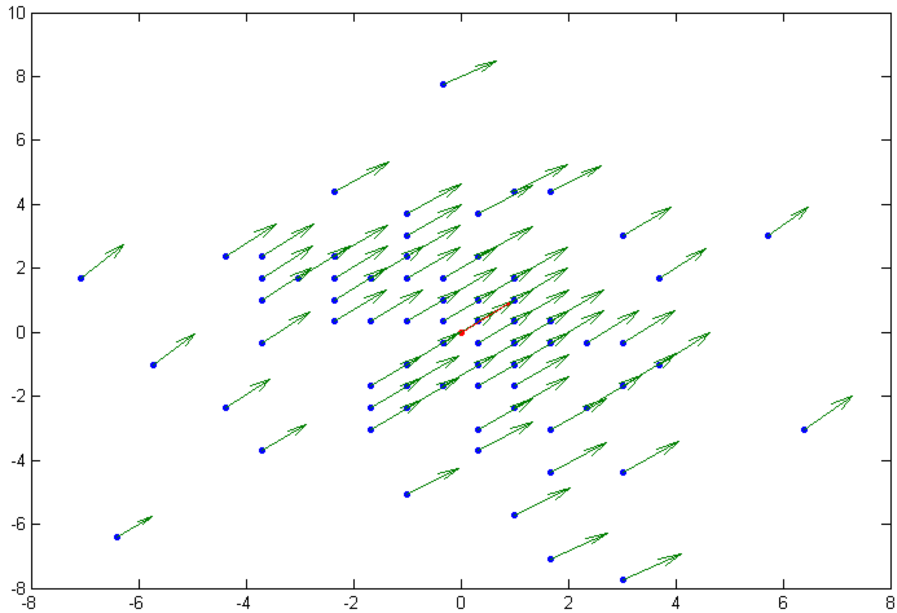


Figure 4.3: The results from performing 100 independent iterations of a single transition of the local process defined by u and $\vec{v}$ (see Section 4.2) from $x \in \mathcal{M}$ (in red), with a tangent vector $(1, 1)$ in local coordinates of the tangent space $T_x(\mathcal{M})$. The starting point x is marked in red and the destinations of the transitions are marked in blue.

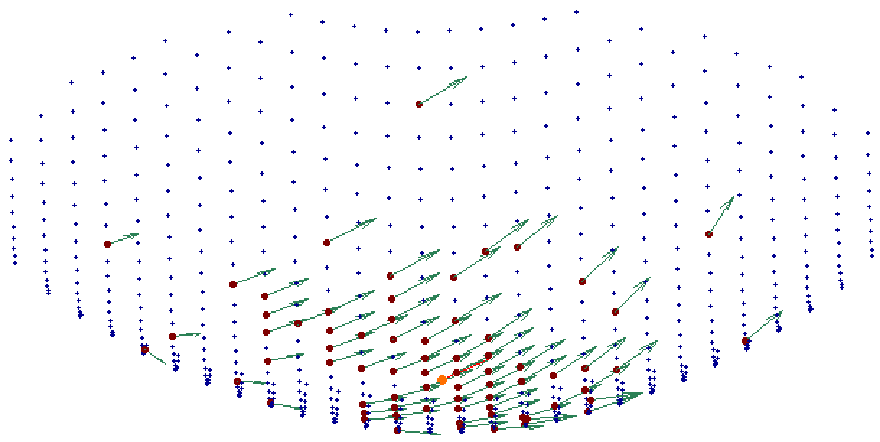
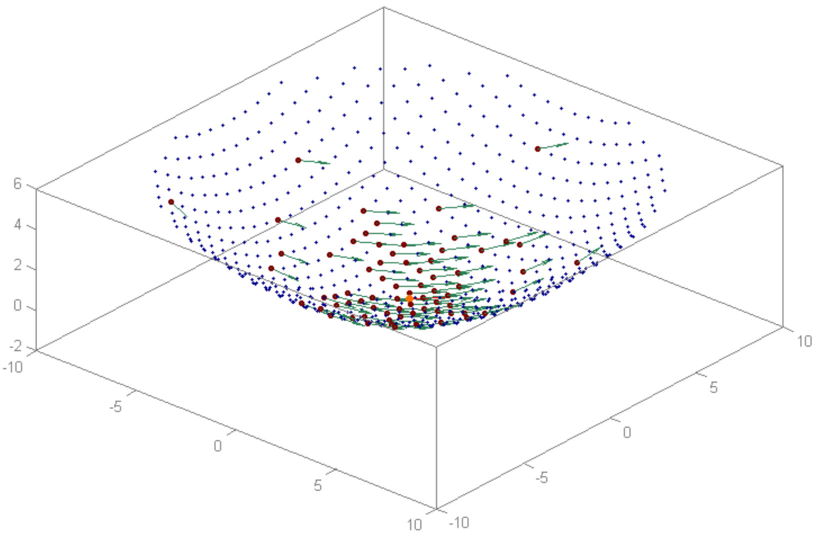


Figure 4.4: The results from performing 100 independent iterations of a single transition of the LPD process defined by $\tilde{u}$ and $\vec{\tilde{v}}$ (see Section 4.2) starting at $x \in \mathcal{M}$ with a tangent vector $(1, 1)$ in local coordinates of the tangent space $T_x(\mathcal{M})$. The points in the area around x on the paraboloid $\mathcal{M}$ are presented. The starting point x is marked in orange, the destinations of the transitions are marked in red, and other points in this area are marked in blue.

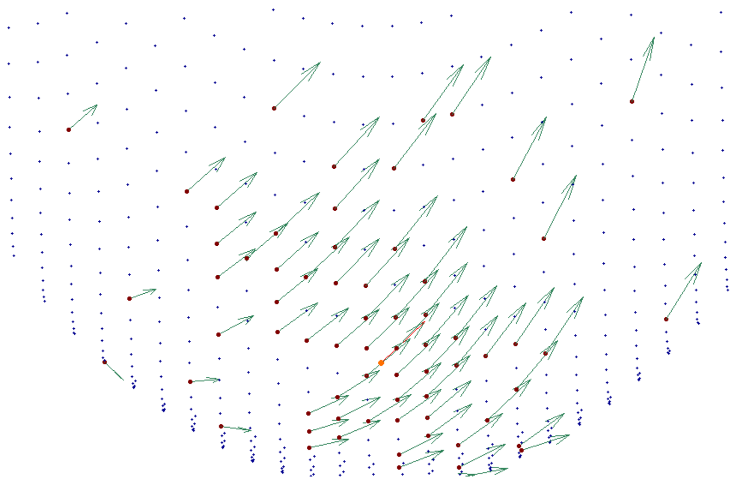
by the equation $z_3 = (z_1/4)^2 + (z_2/4)^2$ for $z = (z_1, z_2, z_3) \in \mathbb{R}^3$. We will refer to this paraboloid from now on as the manifold and denote it $\mathcal{M}$. Assume $x = (0, 0, 0) \in \mathcal{M}$ and the vector $(1, 1, 0)$ is tangent to the paraboloid $\mathcal{M}$ at the origin x. We will demonstrate the local and the global processes defined in Section 4.2 by propagating this vector from x using the stochastic transitions of these processes along the points that were sampled from the paraboloid.

In this case, the parametrization of the manifold is known, therefore there is no need to approximate the (known) tangent spaces of the manifold. We set the basis of the tangent space at x to be the vectors $(1, 0, 0)$ and $(0, 1, 0)$. The bases at every other point were set via parallel transport, which is computed using the known parameterization of the paraboloid, of the basis at x to each of the 8100 other sampled point. Once the bases of the tangent spaces were calculated we constructed the matrices O_y for every samples $y \in \mathcal{M}$ using Eq. 2.1. Finally, we computed the diffusion operator (Eq. 3.1) and constructed the LPD operator from Definition 4.1. We use the constructed operator to perform the LPD process transitions and propagate the vector we set at x on the resulting trajectories. The stochastic nature of a single transition will be demonstrated first, and then the resulting trajectories will be demonstrated.

In order to show the stochastic nature of a single transition, we performed 100 iterations that perform a single transition of the LPD process from x. The LPD transition, which was explained in details in section 4.2, consists of two main phases. First, a transition of the local process u is performed on the tangent space $T_x(\mathcal{M})$. Then, the resulting point and its vector are projected on the manifold to show the transition of the global process $\tilde{u}$. The results from

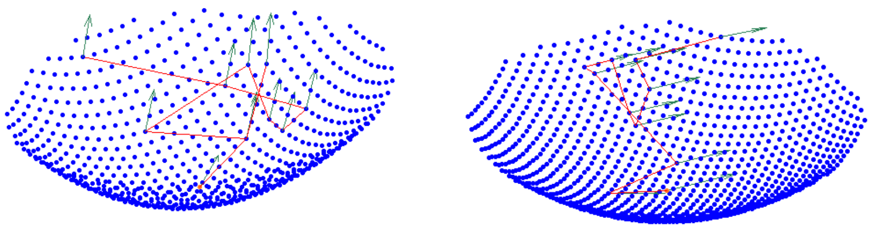


(a) The area around x as seen in the ambient space.



(b) The area around x is magnified here to see the directions of the tangent vectors

Figure 4.5: Two additional perspectives of the transitions shown in Fig. 4.4

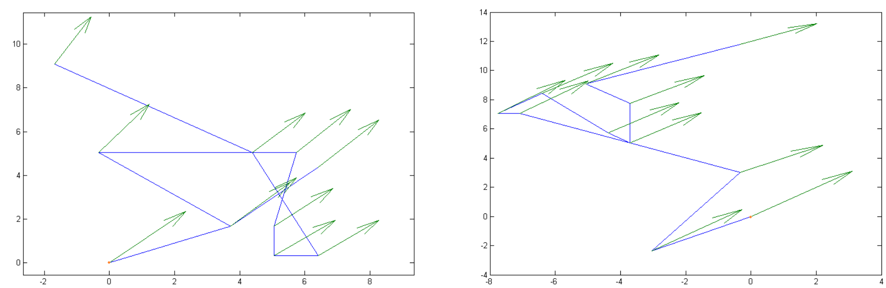


(a) (b)

Figure 4.6: Two independent trajectories of the LPD process defined by $\tilde{u}$ and $\vec{\tilde{v}}$ (see Section 4.2) starting at $x \in \mathcal{M}$ with the tangent vector $(1, 1)$ in local coordinates of the tangent space $T_x(\mathcal{M})$. The starting point x is marked in orange.

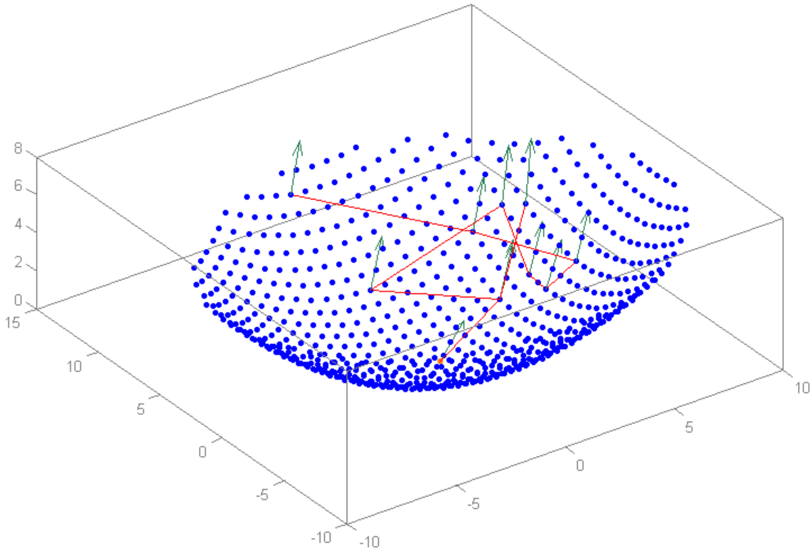
the iterations of the local transitions of u are presented in Fig. 4.3. The ones from the global LPD transitions of $\tilde{u}$ are presented in Figs. 4.4 and 4.5. These results demonstrate the locality of the transition, as well as its similarity to Brownian transitions over points on the manifold. Also, the vectors attached to the points on the manifold have similar magnitudes and directions while still remaining tangent to the manifold.

After a single transition was demonstrated, we perform several iterations that generate a trajectory of the LPD process over the manifold. We will demonstrate two trajectories that were generated by this process. For this demonstration, we will only show the first 10 transitions of each trajectory. Figure 4.6 shows these two trajectories on the manifold. Additional perspectives of the first trajectory are shown in Fig. 4.8 and the second one in Fig. 4.9. The vectors are propagated over the diffusion trajectory and they maintain similar directions and magnitude while remaining tangent to the manifold at their corresponding points. To see this more clearly, we projected each of the trajectories on the initial tangent space $T_x(\mathcal{M})$ at the starting point x. The projected trajectories are shown in Fig. 4.7.

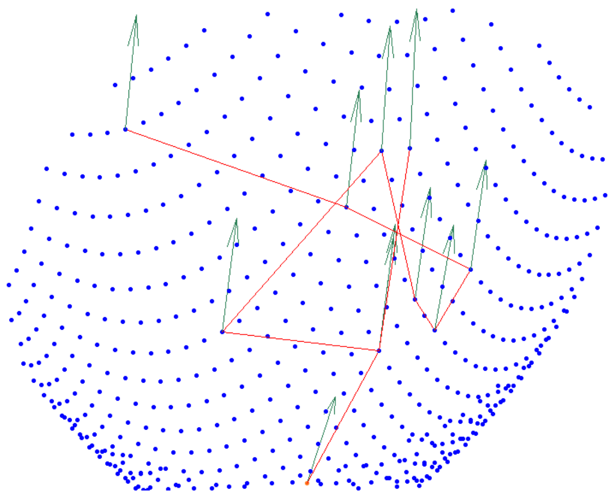


(a) (b)

Figure 4.7: The projection of the trajectories in Fig. 4.6 on the tangent space $T_x(\mathcal{M})$ at the starting point $x \in \mathcal{M}$ of these trajectories. The starting point x is marked in orange.

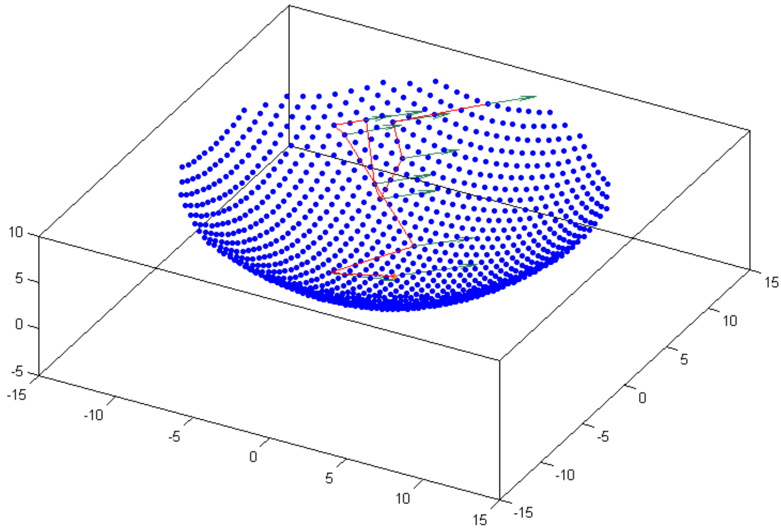


(a) The trajectory on the paraboloid as seen in the ambient space.

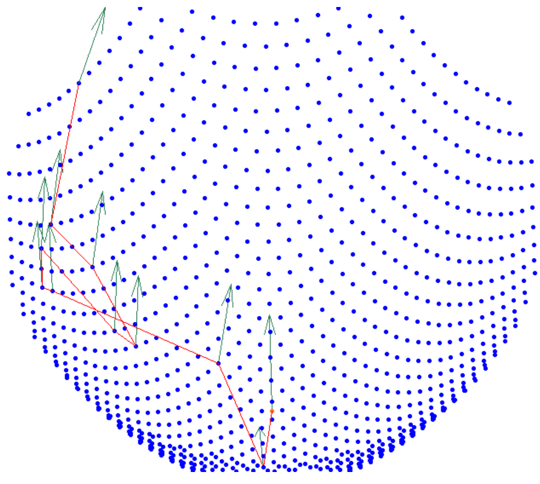


(b) The area containing the trajectory is magnified here to show the propagation of the tangent vectors more clearly.

Figure 4.8: Additional perspectives of the trajectory shown in Fig. 4.6(a).



(a) The trajectory on the paraboloid as seen in the ambient space.



(b) The area containing the trajectory is magnified here to see the propagation of the tangent vectors more clearly.

Figure 4.9: Additional perspectives of the trajectory shown in Fig. 4.6(b).

The LPD operator in [13], which generates the demonstrated stochastic process, was utilized for two data-analysis tasks. Specifically, it was utilized for classification of breast tissue impedance measurements to detect cancerous growth and for image segmentation. The latter application showed that various time-scales of the diffusion process provide different resolutions of the segmentation based on color shades and light levels. We refer the reader to [13] for more information on the implementation and on the applicative results of the LPD-based data analysis.

5. Conclusion

The paper enhances the properties of the linear-projection diffusion (LPD) super-kernels in [13] two-folds:

1. We showed that the infinitesimal generator of the LPD super-kernel converges to a natural extension of the original diffusion operator from scalar functions to vector fields. This operator was shown to be locally equivalent to a composition of linear projections between tangent spaces and the vector-Laplacians on them.
2. We introduced the stochastic process defined by the LPD super-kernels and demonstrated it on a synthetic manifold.

Future research plans include: utilization of the presented LPD super-kernels methodology to provide out-of-sample extension, adapting large kernel-based methods to computing environments with limited resources by applying the patch-based methodologies that were described in [13] and in this paper while processing real massive datasets.

Acknowledgments

This research was partially supported by the Israel Science Foundation (Grant No. 1041/10). The first author was also supported by the Eshkol Fellowship from the Israeli Ministry of Science & Technology.

References

[1] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.

[2] F. Chung. *Spectral Graph Theory*, volume 92. CBMS-AMS, May 1997.

[3] R.R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006. Diffusion Maps and Wavelets.

[4] R.R. Coifman and S. Lafon. Geometric harmonics: A novel tool for multiscale out-of-sample extension of empirical functions. *Applied and Computational Harmonic Analysis*, 21(1):31–52, 2006. Diffusion Maps and Wavelets.

[5] T. Cox and M. Cox. *Multidimensional Scaling.* Chapman and Hall, London, UK, 1994.

[6] D.L. Donoho and C. Grimes. Hessian eigenmaps: New locally linear embedding techniques for high dimensional data. *Proceedings of the National Academy of Sciences of the United States of America*, 100:5591–5596, May 2003.

[7] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24, 1933.

[8] I.T. Jolliffe. *Principal Component Analysis.* Springer, New York, NY, 1986.

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

[10] S. Lafon. *Diffusion Maps and Geometric Harmonics.* PhD thesis, Yale University, May 2004.

[11] S. Mika, , B. Schölkopf, A. Smola, K.-R. Müller, M. Scholz, and G. Rätsch. Kernel pca and de-noising in feature spaces. In *Proceedings of the 1998 conference on Advances in neural information processing systems II*, pages 536–542, Cambridge, MA, USA, 1999. MIT Press.

[12] S.T. Roweis and L.K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, December 2000.

[13] M. Salhov, G. Wolf, and A. Averbuch. Patch-to-tensor embedding. *Applied and Computational Harmonic Analysis*, 2011.

[14] B. Schölkopf, A.J. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, July 1998.

[15] A. Singer and H.-t. Wu. Orientability and diffusion maps. *Applied and Computational Harmonic Analysis*, 2010. In press, to appear.

[16] A. Singer and H.-t. Wu. Vector diffusion maps and the connection laplacian. *Arxiv preprint arXiv:1102.0075*, 2011.

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

[18] G. Yang, X. Xu, and J. Zhang. Manifold alignment via local tangent space alignment. *International Conference on Computer Science and Software Engineering*, December 2008.

[19] Z. Zhang and H. Zha. Principal manifolds and nonlinear dimension reduction via local tangent space alignment. *Technical Report CSE-02-019, Department of Computer Science and Engineering, Pennsylvania State University*, 2002.