

Poisson Manifold Reconstruction — Beyond Co-dimension One

M. Kohlbrenner¹  S. Lee²  M. Alexa¹  M. Kazhdan² 

¹TU Berlin, Germany ²Johns Hopkins University, USA

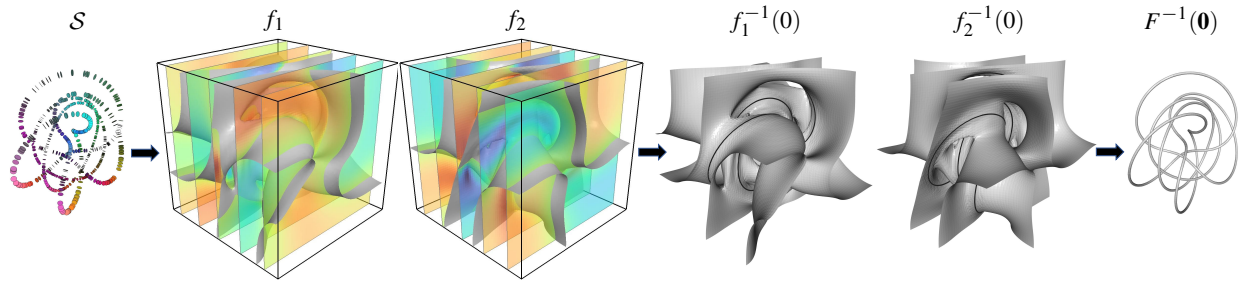


Figure 1: Co-dimension two reconstruction: Starting with an oriented point set, \mathcal{S} , we compute coordinate functions, f_1 and f_2 , whose exterior product of gradients matches the skew symmetric matrix characterizing the samples' normal space. The intersection of the zero level-sets of the coordinate functions, $F^{-1}(\mathbf{0}) = f_1^{-1}(0) \cap f_2^{-1}(0)$, gives the reconstructed manifold. (Connected components away from regions of high sampling density are removed.)

Abstract

Screened Poisson Surface Reconstruction creates 2D surfaces from sets of oriented points in 3D (and can be extended to co-dimension one surfaces in arbitrary dimensions). In this work we generalize the technique to manifolds of co-dimension larger than one. The reconstruction problem consists of finding a vector-valued function whose zero set approximates the input points. We argue that the right extension of screened Poisson Surface Reconstruction is based on exterior products: the orientation of the point samples is encoded as the exterior product of the local normal frame. The goal is to find a set of scalar functions such that the exterior product of their gradients matches the exterior products prescribed by the input points. We show that this setup reduces to the standard formulation for co-dimension 1, and leads to more challenging multi-quadratic optimization problems in higher co-dimension. We explicitly treat the case of co-dimension 2, i.e., curves in 3D and 2D surfaces in 4D. We show that the resulting bi-quadratic problem can be relaxed to a set of quadratic problems in two variables and that the solution can be made effective and efficient by leveraging a hierarchical approach.

CCS Concepts

• **Computing methodologies** → **Shape modeling**; • **Mathematics of computing** → **Nonlinear equations**; **Numerical analysis**;

Keywords: curve and surface reconstruction, sub-manifold reconstruction, exterior product, polynomial optimization

1. Introduction

A central task in geometry processing is the reconstruction of a surface from point samples. Most of the work in this area considers two-dimensional surfaces embedded in 3D. Among different ways to define a surface from point samples (see Section 2), level-set techniques and, in particular, (screened) Poisson Surface Reconstruction [KBH06, KH13] have proven to be particularly resilient to common sampling artifacts such as noise, outliers, and missing data. The reason for this robustness is that they treat the reconstruction problem globally (see Section 3 for details).

The aim of this work is to generalize the class of Poisson Surface

Reconstruction methods to oriented d -dimensional manifolds embedded in an n -dimensional space with *co-dimension* $\bar{d} = n - d > 1$. The input is a sampling \mathcal{S} of the manifold, consisting of locations of the samples and (oriented) frames spanning the \bar{d} -dimensional normal space. The output is a vector valued function $F : \mathbb{R}^n \rightarrow \mathbb{R}^{n-d}$ that represents the manifold as a level-set, i.e.

$$\mathcal{M}_{\mathcal{S}} = \{\mathbf{x} \in \mathbb{R}^n : F(\mathbf{x}) = \mathbf{0}\}. \quad (1)$$

As in screened Poisson Surface Reconstruction, we solve for the coordinate functions of F whose gradients span the normal frames prescribed at the sample points. This poses several challenges.

When $\bar{d} > 1$, there is a continuum of frames that can be used to represent the same oriented \bar{d} -dimensional normal space at a point. As the reconstructed manifold should only depend on the oriented normal space at the samples, we require a representation that is agnostic to the particular choice of frame.

For a similar reason, there can be a continuum of solutions F whose gradients span the normal space at the sample points. Often, this reflects the fact that the underlying optimization is non-convex making it hard to find solutions that avoid getting trapped in unwanted local minima.

Key contributions Our main contributions are choices in representation and optimization that lead to a practical extension of screened Poisson Surface Reconstruction to higher co-dimension, overcoming the mentioned challenges:

- We argue that the right mathematical tool to deal with the multiplicity of frames is *exterior algebra* (see Section 4 for a brief introduction): we encode the local normal space of the manifold as an exterior product — making the representation independent of the particular choice of frame.
- We show that using the exterior product (Section 5), the problem of solving for the coordinate functions of F reduces to minimizing a multi-quadratic energy. The continuous approach can be discretized using a finite basis, and we specifically develop a discretization for the case $\bar{d} = 2$ (Section 6).
- We propose a hierarchical approach for minimizing the energy that smooths out the energy landscape at coarser resolutions, keeping the optimization from getting trapped in local minima and consistently returning the same solution, regardless of initial guess (Section 7).

We demonstrate the efficacy of our approach in reconstructing curves in 3D and surfaces in 4D, analyzing run-time performance, convergence, and stability in the presence of noise (Section 8). We conclude by offering some discussion and proposing directions for future work (Section 9).

2. Related Work

For the vast field of curve reconstruction in the plane and surface reconstruction in 3D, we refer the reader to existing overview literature [OPP*21, BTS*14, Dey06]. We defer a discussion of reconstructing a surface as the level set of a signed scalar function, i.e., $\bar{d} = 1$ to the next section. We are unaware of work that extends this approach to the intersection of level-sets ($\bar{d} > 1$). Here we focus on reconstruction approaches that have been extended to manifolds in arbitrary dimension and co-dimension.

Subsets of simplicial complexes A classic approach to curve reconstruction in the plane is based on first triangulating the set of points and then selecting an appropriate subset of the edges [KSO04]. For nicely sampled curves one observes that the Voronoi cells of the points are elongated in direction normal to the curve [AB99]. While many techniques based on this observation in 2D fail to generalize to 3D or higher dimension, the general observation still holds and allows detecting the local dimension of the tangent space in higher dimension and co-

dimension [DGGZ02, ACSTD07]. Combining this idea with modifying the weights in a weighted Delaunay triangulation [Aur87, dGMMD14] to remove slivers [CDE*00] leads to a first algorithm with guarantees [CDR05]. By the authors own admission it is not practical “mainly because it requires a very dense and noise-free sample”, but also because of the high computational complexity.

An idea that keeps the complexity mostly dependant on the dimension of the manifold, and to a lesser extent the ambient space, is to compute the Delaunay triangulation in the tangent planes of the points [BF04]. While this could be used for reconstruction directly [Fre02], the combination with the above ideas on sliver-removal leads to a practical algorithm with guarantees [BG10]. Notably, it is only linear in ambient dimension (but quadratic in the number of samples).

All of the above require clean and dense samples. Some tolerance to noise can be achieved by appropriately selecting a subset of the samples, for example using topological persistence [CO08].

Projection operator / ridges of density function A popular approach in computer graphics and geometric modeling for defining a surface from surface samples is to use a projection operator: the operator is defined by repeatedly projecting any point in space onto a locally estimated tangent plane. The process can be carefully constructed so that it satisfies the properties of a projection [Lev04] and the surface is defined as the set of stationary points.

This idea has been used first for curve reconstruction from points [Lee00], and later for surfaces in 3D [ABCO*03]. It generalizes directly to arbitrary dimension and co-dimension [AA06, SL20]. For well-behaved manifolds and dense enough sampling the projection can be constructed so that it guarantees homeomorphic reconstruction [CC19]. Likewise, tolerance to some noise can be proven [FIK*18]. Relying on a projection operator and only locally establishing tangent- and/or normal-spaces alleviates the problem that not all manifolds admit globally smooth frames and even supports reconstruction of non-orientable manifolds. In contrast, globally smooth reconstruction potentially deals better with outliers and missing data, and the signed implicit function significantly simplifies iso-surfacing the zero-level set.

In machine learning, related surface definitions have been used [DG03], albeit from a different mathematical view-point: the point set is considered to be a discrete sample of a probability distribution and the manifold may then be defined as the *ridge* of an appropriately chosen density function [OE11, GPPVW14].

Parametric curves and surfaces A common theme in CAGD is fitting a spline curve or surface to given point data. This is done essentially by defining a distance between the parametric curve or surface to the samples and then optimizing the control points to minimize this distance measure. For curves this has been done in application domains using the idea of *active contour models* [CnRT*00, SKG15]. The idea has been extended to 3D [PL03, PLH02] and it seems that it could also be used in higher dimensions and co-dimension.

3. Background: Implicit Surface Reconstruction

There are different strategies for reconstructing a surface from point samples. We focus on methods that first approximate the input data with a level-set of a scalar function and then extract a typically piecewise linear approximation of the level-set as an explicit surface representation. One approach has been to represent the surface implicitly by a (possibly truncated) signed distance function [HDD*92, CL96, CBC*01, MPS08, NIH*11, CT11]. While this has advantages in some applications, for the reconstruction problem it has been observed that it is useful to ask only that the function has non-vanishing gradients close to the surface. This avoids spurious surface components and is the central idea of screened Poisson Surface Reconstruction [KH13].

Screened Poisson Surface Reconstruction Given a set of oriented points $\mathcal{S} \subset \mathbb{R}^3 \times \mathbb{R}^3$ where each sample $\mathbf{s} = (\mathbf{s.p}, \mathbf{s.n})$ is described by its position in \mathbb{R}^3 and the associated (possibly area-weighted) outward-facing normal, the screened Poisson Surface Reconstruction algorithm proceeds in three steps.

First, a sampling density $\rho : \mathbb{R}^3 \rightarrow \mathbb{R}^{\geq 0}$ is estimated and the oriented samples are extended to a target vector-field $V_{\mathcal{S}} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ by “splatting” the samples’ normals:

$$V_{\mathcal{S}}(\mathbf{q}) = \sum_{\mathbf{s} \in \mathcal{S}} \kappa_{\mathbf{s.p}}(\mathbf{q}) \cdot \mathbf{s.n} \quad (2)$$

with $\kappa_{\mathbf{p}}$ a splatting kernel (weighted inversely to density ρ).

Next, the implicit function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ is found by minimizing the combination of gradient-fitting and screening energies:

$$E(f) = \int_{\mathbb{R}^3} \|\nabla f - V_{\mathcal{S}}\|^2 + \alpha \sum_{\mathbf{s} \in \mathcal{S}} f(\mathbf{s.p})^2. \quad (3)$$

Finally, given the function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$, the reconstructed manifold $\mathcal{M}_{\mathcal{S}}$ is obtained by extracting the zero level-set:

$$\mathcal{M}_{\mathcal{S}} = \left\{ \mathbf{p} \in \mathbb{R}^3 : f(\mathbf{p}) = 0 \right\}. \quad (4)$$

Choosing a (finite) function basis, this reduces to solving a linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ where the system matrix \mathbf{A} encodes the discrete Laplacian combined with the mass matrix defined by the samples, the vector \mathbf{b} describes the divergence of $V_{\mathcal{S}}$, and the the solution vector \mathbf{x} gives the coefficients of the implicit function in the basis. Choosing a B-spline basis, the system is solved efficiently using a hierarchical solver [BHM00]. Iso-surfacing can be done using standard techniques [LC87, SW04, KKDH07] to obtain a triangle mesh approximating the zero level-set of f .

4. Exterior Products

In co-dimension $\bar{d} = 1$, we can represent the d -dimensional tangent space at a point uniquely in terms of its outward-facing normal. In a similar way, for co-dimension $\bar{d} \geq 2$, we would like to represent the d -dimensional tangent space uniquely in terms of the \bar{d} -dimensional normals, without explicitly choosing a basis for the \bar{d} -dimensional normal space. Our desiderata are: (1) The representation should be linear, and (2) it should uniquely encode the oriented normal frame together with its measure (area, volume, etc.).

We begin by reviewing the necessary concepts for co-dimension 2, i.e., two vectors spanning the normal space, and generalize later. We focus our review on the case that the vector space is \mathbb{R}^n , with canonical basis $(\mathbf{e}_1, \dots, \mathbf{e}_n)$, with the i -th coefficient of \mathbf{e}_i equal to 1 and all other coefficients equal zero.

Exterior product of two vectors The right concept for our purpose is the *exterior product*. Given two vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$, the exterior product $\mathbf{u} \wedge \mathbf{v}$ represents the *vector area* of the parallelogram spanned by \mathbf{u} and \mathbf{v} in \mathbb{R}^n . It satisfies the following properties:

Anti-symmetry $\mathbf{u} \wedge \mathbf{v} = -\mathbf{v} \wedge \mathbf{u}$

(Bi-)linearity $(\alpha\mathbf{u} + \beta\mathbf{v}) \wedge \mathbf{w} = \alpha(\mathbf{u} \wedge \mathbf{w}) + \beta(\mathbf{v} \wedge \mathbf{w})$

It follows that exterior products can be represented in a vector space with basis $\mathbf{e}_i \wedge \mathbf{e}_j, i < j$. Let $\mathbf{u} = \sum_i u_i \mathbf{e}_i$ and $\mathbf{v} = \sum_j v_j \mathbf{e}_j$ we have:

$$\begin{aligned} \mathbf{u} \wedge \mathbf{v} &= \left(\sum_i u_i \mathbf{e}_i \right) \wedge \left(\sum_j v_j \mathbf{e}_j \right) \\ &= \sum_{i,j} u_i v_j (\mathbf{e}_i \wedge \mathbf{e}_j) \\ &= \sum_{i,j>i} (u_i v_j - u_j v_i) (\mathbf{e}_i \wedge \mathbf{e}_j). \end{aligned} \quad (5)$$

Geometrically, we can think of the coefficients $u_i v_j - u_j v_i$ in this basis as the (signed) areas of the shadows of the parallelogram on the planes spanned by $\mathbf{e}_i, \mathbf{e}_j$. An algebraic view leads to a convenient implementation: noting that the *outer product* $\mathbf{u}\mathbf{v}^T$ is bilinear and taking the difference

$$\mathbf{u} \wedge \mathbf{v} \simeq \mathbf{u}\mathbf{v}^T - \mathbf{v}\mathbf{u}^T \quad (6)$$

yields the anti-symmetry in the form of a skew-symmetric matrix.

Constructing the exterior product of $(\mu\mathbf{v} + \nu\mathbf{u})$ and $(\mu'\mathbf{v} + \nu'\mathbf{u})$ results in $(\mu\nu' - \nu\mu')(\mathbf{u}\mathbf{v}^T - \mathbf{v}\mathbf{u}^T)$. Noting that the scalar term is the determinant of $\begin{pmatrix} \mu & \nu \\ \mu' & \nu' \end{pmatrix}$, the matrix acting on the vectors \mathbf{u} and \mathbf{v} by right-multiplication, shows the invariance to the group of transformations that map the plane spanned by the frame onto itself and preserve the signed area — the matrix group $SL(2, \mathbb{R})$. As before, a basis can be constructed as $\mathbf{e}_i \mathbf{e}_j^T - \mathbf{e}_j \mathbf{e}_i^T$ with $i < j$, spanning all $\mathbb{R}^{n \times n}$ skew-symmetric matrices. The basis has $\binom{n}{2}$ elements.

As desired, we can easily perform computation on this space as the skew-symmetric matrices form a vector space. However, while every frame with signed area can be uniquely encoded as a skew-symmetric matrix, not every skew-symmetric matrix is the exterior product of two vectors. In fact, the non-zero exterior products are precisely the skew symmetric matrices with rank 2.

Linearization One of the challenges in working with the exterior product is that it is bi-linear in the input, making quadratic energies in the exterior product difficult to optimize. An obvious linearization results from fixing one of the two vectors and considering the energy as a function of the other one.

Exterior product spaces All of the concepts above extend naturally to co-dimension $\bar{d} > 2$. The geometric picture is that we encode the oriented vector measure of a parallelepiped spanned by \bar{d} vectors $\mathbf{v}_1, \dots, \mathbf{v}_{\bar{d}} \in \mathbb{R}^n$ as the signed measures of the shadows onto \bar{d} -dimensional linear subspaces spanned by \bar{d} -tuples of canonical

basis vectors. We denote this space as $\bigwedge^{\bar{d}} \mathbb{R}^n$. It is spanned by the basis $\mathbf{e}_{i_1} \wedge \dots \wedge \mathbf{e}_{i_{\bar{d}}}$, with $i_1 < \dots < i_{\bar{d}}$.

Anti-symmetry now means that if the positions of any two elements in the product are transposed, the sign changes. To explain what this entails we use a multi-dimensional index $\mathbf{i} = (i_1, \dots, i_{\bar{d}})$. A permutation π of the indices can be encoded as an orthogonal matrix in $\{0, 1\}^{\bar{d} \times \bar{d}}$ and $\text{sign} \pi$ is the determinant of this matrix. Intuitively, $\text{sign} \pi$ equals $+1$ if the number of transpositions in π is even, otherwise it is -1 . With this notation, the anti-symmetry is characterized by the property that for any permutation π , we have:

$$\mathbf{v}_1 \wedge \dots \wedge \mathbf{v}_{\bar{d}} = \text{sign}(\pi) (\mathbf{v}_{\pi(1)} \wedge \dots \wedge \mathbf{v}_{\pi(\bar{d})}). \quad (7)$$

The algebraic construction now requires tensor products and \bar{d} -th order tensors (generalizations of square matrices with more than two indices). The tensor product $\mathbf{v}_1 \otimes \dots \otimes \mathbf{v}_{\bar{d}}$ is multilinear and generates all possible \bar{d} -wise products of the entries of the vectors. We can represent the exterior product of \bar{d} vectors as an anti-symmetric \bar{d} -th order tensor, similar to the representation by skew symmetric matrices, as

$$\mathbf{v}_1 \wedge \dots \wedge \mathbf{v}_{\bar{d}} \equiv \sum_{\pi} \text{sign} \pi (\mathbf{v}_{\pi(1)} \otimes \dots \otimes \mathbf{v}_{\pi(\bar{d})}). \quad (8)$$

In practice, it suffices to store only the entries with multi-indices \mathbf{i} for which $i_1 < \dots < i_{\bar{d}}$. Entries with repeated index $i_j = i_k$ are necessarily zero because of anti-symmetry. This means it suffices to store $\binom{n}{\bar{d}}$ coefficients.

As in the two-dimensional case, the space of anti-symmetric \bar{d} -th order tensors uniquely encodes the \bar{d} -frames with signed measure, sometimes referred to as \bar{d} -blades, but also contains elements that have no corresponding \bar{d} -frame.

Exterior product fields

We remark that the above discussion extends directly to vector-fields on \mathbb{R}^n by applying the exterior product point-wise. For example, given vector-fields $\vec{v}_1, \dots, \vec{v}_{\bar{d}} : \mathbb{R}^n \rightarrow \mathbb{R}^n$, the exterior product defines an anti-symmetric tensor field on \mathbb{R}^n :

$$(\vec{v}_1 \wedge \dots \wedge \vec{v}_{\bar{d}})(\mathbf{p}) \equiv \vec{v}_1(\mathbf{p}) \wedge \dots \wedge \vec{v}_{\bar{d}}(\mathbf{p}).$$

5. Generalizing Screened Poisson Surface Reconstruction

In this section we describe the generalization of screened Poisson Surface Reconstruction (sPSR) to the reconstruction of co-dimension $\bar{d} \geq 2$ manifolds. For simplicity, we focus on the case $\bar{d} = 2$, though our discussion extends to higher co-dimension. Figure 1, gives a visualizations of our process.

To simplify notation, we overload the operator $\langle \cdot, \cdot \rangle$ (and correspondingly $\| \cdot \|^2$) so that for scalar-fields, vector-fields, and skew-symmetric matrix fields, it denotes the integral of the product, the dot-product, and the Frobenius-dot-product (respectively) over \mathbb{R}^n .

5.1. Overview of the approach

As in sPSR, we start with oriented samples $\mathcal{S} \subset \mathbb{R}^n \times (\mathbb{R}^n \times \mathbb{R}^n)$, where each sample is described by its position in n -dimensional

space and two vectors describing the (weighted) oriented normal space at that point. Our goal is to solve for a vector-valued function $F = (f_1, f_2) : \mathbb{R}^n \rightarrow \mathbb{R}^2$ with the property that at every sample $\mathbf{s} = (\mathbf{p}, \mathbf{n}_1, \mathbf{n}_2)$ the gradients $\nabla f_1|_{\mathbf{s}, \mathbf{p}}$ and $\nabla f_2|_{\mathbf{s}, \mathbf{p}}$ describe the same normal space as the vectors $\mathbf{s} \cdot \mathbf{n}_1$ and $\mathbf{s} \cdot \mathbf{n}_2$. In the language of exterior products, we want $\nabla f_1|_{\mathbf{s}, \mathbf{p}} \wedge \nabla f_2|_{\mathbf{s}, \mathbf{p}} = \mathbf{s} \cdot \mathbf{n}_1 \wedge \mathbf{s} \cdot \mathbf{n}_2$.

To achieve this, we proceed as in sPSR, estimating the sampling density $\rho : \mathbb{R}^n \rightarrow \mathbb{R}^{\geq 0}$ and extending the samples' normals to a target skew-symmetric matrix field: $V_S : \mathbb{R}^n \rightarrow \bigwedge^2 \mathbb{R}^n$ by "splatting" the samples' skew-symmetric matrices:

$$V_S(\mathbf{q}) = \sum_{\mathbf{s} \in \mathcal{S}} \kappa_{\mathbf{p}}(\mathbf{q}) \cdot \mathbf{s} \cdot \mathbf{n}_1 \wedge \mathbf{s} \cdot \mathbf{n}_2, \quad (9)$$

with $\kappa_{\mathbf{p}}$ a splatting kernel (weighted inversely to density ρ).

Then we solve for the function $F = (f_1, f_2)$ minimizing the combination of skew-symmetric-matrix-fitting and screening energies:

$$E(F) = \|\nabla f_1 \wedge \nabla f_2 - V_S\|^2 + \alpha \sum_{\mathbf{s} \in \mathcal{S}} (f_1(\mathbf{s}, \mathbf{p})^2 + f_2(\mathbf{s}, \mathbf{p})^2). \quad (10)$$

Finally, given the function $F : \mathbb{R}^n \rightarrow \mathbb{R}^2$, the reconstructed manifold \mathcal{M}_S is obtained by extracting the zero level-set:

$$\mathcal{M}_S = \{\mathbf{p} \in \mathbb{R}^n : F(\mathbf{p}) = \mathbf{0}\}. \quad (11)$$

Feasibility Before proceeding, it is natural to ask:

Given a d -dimensional manifold $\mathcal{M} \subset \mathbb{R}^n$, does there exist $F : \mathbb{R}^n \rightarrow \mathbb{R}^{n-d}$ such that $\mathcal{M} = F^{-1}(\mathbf{0})$, with the gradients of F linearly independent at $F^{-1}(\mathbf{0})$?

Clearly, for such an F to exist, the manifold must have a trivial normal bundle. For manifolds of dimension $d \geq 4$ this is known to be more restrictive than orientability (one such example being complex projective 2-space which, by Whitney's Embedding Theorem [GP10], can be embedded in \mathbb{R}^n for n large enough). For manifolds of dimension $d = 1$ and $d = 3$, [Kir89, Sec. VIII, Thm. 1] shows that this is equivalent to orientability, and the same appears to hold for $d = 2$. However, in the general case, there exist closed manifolds embedded in \mathbb{R}^n with trivial normal bundles that cannot be realized as level-sets of an implicit function [AK85].

It is known that if \mathcal{M} has a trivial normal bundle, such an F can be constructed locally (i.e. in a tubular neighborhood) [Lee12, Ex. 10-18] and can be extended to a function $F : \mathbb{R}^n \rightarrow S^{n-d}$ (e.g. using the Pontryagin-Thom Collapse Map) [Mil65, Ch. 7, Thm. C]. Furthermore, Milnor's construction of a function $F : \mathbb{R}^n \rightarrow S^{n-d}$ can be transformed into a function $F : \mathbb{R}^n \rightarrow \mathbb{R}^{n-d}$ such that \mathcal{M} is a disconnected subset of $F^{-1}(\mathbf{0})$. (See Appendix A.) We leverage this in Section 5.4, using the local sampling density to select the connected components of $F^{-1}(\mathbf{0})$ comprising the reconstruction.

Finally, we note that while our approach assumes that the solution \mathcal{M} has a trivial normal bundle, the use of the exterior product to describe the normal space means we do not require a trivialization to be given. This is in contrast to [Lee12, Mil65] that use the framing to define the function in the tubular neighborhood.

5.2. Optimization

The challenge in minimizing the energy in Equation 10 is that the exterior product $\nabla f_1 \wedge \nabla f_2$ is *bi*-linear in the functions f_1 and f_2 , making the fitting energy term bi-quadratic. Thus, minimization requires solving a cubic system in f_1 and f_2 . We address this by alternately linearizing. Concretely, fixing f_i , let \mathcal{F}_i be the map taking scalar fields in \mathbb{R}^n to skew-symmetric matrix fields:

$$\mathcal{F}_1(f) = \nabla f_1 \wedge \nabla f \quad \text{and} \quad \mathcal{F}_2(f) = \nabla f \wedge \nabla f_2.$$

This map is linear, so for fixed f_2 the energy in f_1 is quadratic

$$E(f_1) = \|\mathcal{F}_2(f_1) - V_S\|^2 + \alpha \sum_{\mathbf{s} \in \mathcal{S}} \left(f_1(\mathbf{s}, \mathbf{p})^2 + f_2(\mathbf{s}, \mathbf{p})^2 \right)$$

and can be minimized by solving a linear system. Analogously, fixing f_1 gives a quadratic energy in f_2 .

Informal remark

We note that the linearization of the energy has an elegant geometric interpretation. Assume the idealized situation in which f_2 is the indicator function of some co-dimension 1 surface Ω passing through the co-dimension 2 manifold \mathcal{M}_S . Consider the contribution of points $\mathbf{q} \in \mathbb{R}^n$ to the energy $E(f_1)$.

$\mathbf{q} \notin \Omega$: In this case $V_S(\mathbf{q})$ vanishes because \mathbf{q} is away from the samples and $\nabla f_1 \wedge \nabla f_2$ vanishes regardless of the value of f_1 because f_2 is constant away from Ω . That is, points $\mathbf{q} \notin \Omega$ do not contribute to the energy, allowing us to replace the domain of integration in the energy:

$$E(f_1) = \int_{\Omega} \|\nabla f_1 \wedge \nabla f_2 - V_S\|^2.$$

$\mathbf{q} \in \Omega$: In this case consider the projection $\pi_{\mathbf{q}} : T_{\mathbf{q}}\mathbb{R}^n \rightarrow T_{\mathbf{q}}\Omega$, taking vectors in \mathbb{R}^n to their projection onto the tangent space of Ω at \mathbf{q} . Decomposing the gradient of f_1 as:

$$\nabla f_1 = \pi(\nabla f_1) + (\nabla f_1 - \pi(\nabla f_1))$$

and noting that the second term is parallel to ∇f_2 , we get:

$$E(f_1) = \int_{\Omega} \|\pi(\nabla f_1) \wedge \nabla f_2 - V_S\|^2.$$

Letting $\vec{N} : \Omega \rightarrow T\Omega$ be the (unique) tangent vector field on Ω such that $\vec{N} \wedge \nabla f_2 = V_S$, we can re-write the energy as:

$$E(f_1) = \int_{\Omega} \|\pi(\nabla f_1) \wedge \nabla f_2 - \vec{N} \wedge \nabla f_2\|^2 = \int_{\Omega} \|\pi(\nabla f_1) - \vec{N}\|^2.$$

Noting that \vec{N} vanishes for points $\mathbf{q} \notin \mathcal{M}_S$ (since V_S vanishes there) and, up to scale, \vec{N} is the normal to \mathcal{M}_S on Ω (since the norms of ∇f_2 and V_S should be constant on \mathcal{M}_S), the linearization seeks the function f_1 whose projected gradient is the vector field that vanishes away from \mathcal{M}_S and is equal to the normal of \mathcal{M}_S (on Ω) at \mathcal{M}_S .

That is, the linearization solves a generalized version of the Poisson Reconstruction problem in which the samples \mathcal{S} live on a manifold Ω , rather than in Euclidean space.

5.3. Regularization

Note that if we can find a solution $F : \mathbb{R}^n \rightarrow \mathbb{R}^2$ for which the energy vanishes, then for any $\mathbf{M} \in \text{SL}(2, \mathbb{R})$ the function $F \cdot \mathbf{M}$ will also have vanishing energy. This is because (1) the exterior products of gradients is invariant to linear re-combinations of the coordinate functions which preserve the skew-symmetric matrix and (2) the screening term will continue to vanish for all linear re-combinations.

A natural way to select between the multiplicity of solutions is to prefer those functions $F = (f_1, f_2)$ for which $(\nabla f_1, \nabla f_2)$ is an orthogonal frame with equal-length axes — $\langle \nabla f_1, \nabla f_2 \rangle = 0$ and $\|\nabla f_1\|^2 = \|\nabla f_2\|^2$. Incorporating this directly into the energy is challenging as it would add quartic terms into the optimization.

Instead, we propose incorporating the Dirichlet energy, often used as a smoothness regularizer, defining the modified energy as the sum of the energy in Equation 10 and a Dirichlet regularizer:

$$E^{\text{reg}}(F) = E(F) + \beta \cdot (\|\nabla f_1\|^2 + \|\nabla f_2\|^2). \quad (12)$$

As the Dirichlet term is quadratic in F , this does not make the system fundamentally more difficult to solve.

We note that while the addition of the Dirichlet energy does introduce a preference for a function $F = (f_1, f_2)$ whose coefficients' gradients are perpendicular and equal length, there remains an ambiguity due to the action of $\text{SO}(2)$, similar to the ambiguity when defining smooth frames along curves [BWR*08].

Intuition The idea behind this approach can be understood by considering the cross-product in \mathbb{R}^3 (formally the dual of the exterior product). Given a vector $\mathbf{n} \in \mathbb{R}^3$ consider asking for a pair of vectors $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{R}^3$ whose cross-product equals \mathbf{n} . There exist many such pairs of vectors and to narrow the solution space we consider the constrained optimization problem of minimizing

$$E(\mathbf{v}_1, \mathbf{v}_2) = \|\mathbf{v}_1\|^2 + \|\mathbf{v}_2\|^2 \quad \text{s.t.} \quad \mathbf{v}_1 \times \mathbf{v}_2 = \mathbf{n}.$$

Using the fact that $\|\mathbf{n}\| = \|\mathbf{v}_1\| \cdot \|\mathbf{v}_2\| \cdot \sin(\alpha)$ where α is the angle between \mathbf{v}_1 and \mathbf{v}_2 , we get:

$$E(\mathbf{v}_1, \mathbf{v}_2) = \|\mathbf{n}\| \cdot \left(\frac{\|\mathbf{v}_1\|^2}{\|\mathbf{n}\|} + \frac{\|\mathbf{n}\|}{\|\mathbf{v}_1\|^2 \sin^2(\alpha)} \right).$$

This is minimized when $\sin(\alpha) = 1$ and $\|\mathbf{v}_1\| = \|\mathbf{v}_2\| = \sqrt{\|\mathbf{n}\|}$. That is, when the vectors are orthogonal and equal length.

Relation to sPSR The Dirichlet energy term is not necessary in sPSR because there the gradient is explicitly constrained to match V_S . In contrast, for higher co-dimensions the constraint is formulated only on the exterior product of the gradients.

5.4. Trimming the level-set

One of the challenges for co-dimension $\bar{d} \geq 2$ is that the zero level-set of F may be a superset of the manifold \mathcal{M} , with spurious geometric components outside a tubular neighborhood of \mathcal{M} . (In the case of co-dimension $\bar{d} = 1$, sPSR solves for the indicator function of \mathcal{M} , which does not have this problem.)

Since the reconstruction process estimates the sampling density

$\rho : \mathbb{R}^n \rightarrow \mathbb{R}^{\geq 0}$, we address this problem by using ρ to trim the level-set, discarding connected components of the level-set over which the (maximum) sampling density is low.

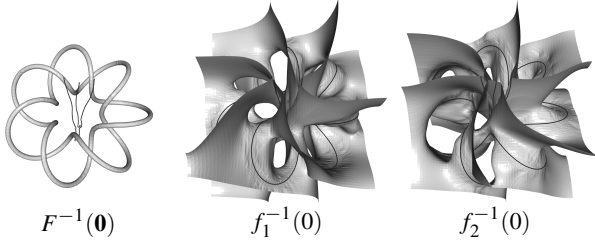


Figure 2: The zero level-set of the function reconstructed from samples of the (2,7) torus-knot (left column) and the zero level-sets of the individual coordinate functions (right two columns).

Figure 2 shows a visualization of this for the case of a (2,7) torus-knot. The zero level-set is shown on the left, visualized as the union of dilated curves drawn in gray (components for which some part is near a region of high sampling density) and curves drawn in black (components for which no part is near a region of high sampling density). Looking at the level-sets of f_1 and f_2 individually (right two columns) it is not surprising that the spurious curves drawn in black appear. The target curve is highly oscillatory so we expect the functions f_1 and f_2 to exhibit similar twisting to ensure that they intersect along the curve.

This challenge further motivates the need for f_1 and f_2 to have equal-length and orthogonal gradients. If the gradients are orthogonal and equal length, then for a point $\mathbf{p} \in \mathcal{M}$ on the manifold, any unit normal direction \mathbf{n} , and small ε , we have:

$$\|F(\mathbf{p} + \varepsilon \cdot \mathbf{n}) - F(\mathbf{p})\|^2 \approx \varepsilon^2 \cdot \|\nabla f_i|_{\mathbf{p}}\|^2.$$

This means that as we (locally) move off the manifold at \mathbf{p} , we can expect the deviation of F from zero to grow quickly and independently of the direction in which we move, so that the spurious geometry should not be too close to the target manifold.

5.5. General co-dimension

While this section focused on the particular case of $\bar{d} = 2$, the discussion generalizes directly to higher co-dimension.

Constructing the target anti-symmetric tensor field Given an oriented point set $\mathcal{S} \subset \mathbb{R}^n \times (\mathbb{R}^n \times \dots \times \mathbb{R}^n)$, we can construct the anti-symmetric tensor field $V_S : \mathbb{R}^n \rightarrow \wedge^{\bar{d}} \mathbb{R}^n$ by setting:

$$V_S(\mathbf{q}) = \sum_{\mathbf{s} \in \mathcal{S}} \kappa_{\mathbf{s}, \mathbf{p}(\mathbf{q})} \cdot \mathbf{s} \cdot \mathbf{n}_1 \wedge \dots \wedge \mathbf{s} \cdot \mathbf{n}_{\bar{d}}.$$

Defining the energy As above, the goal can be stated as solving for the function $F = (f_1, \dots, f_{\bar{d}}) : \mathbb{R}^n \rightarrow \mathbb{R}^{\bar{d}}$ minimizing:

$$E(F) = \|\nabla f_1 \wedge \dots \wedge \nabla f_{\bar{d}} - V_S\|^2 + \alpha \sum_{\mathbf{s} \in \mathcal{S}} \|F(\mathbf{s}, \mathbf{p})\|^2 + \dots$$

with the ellipses denoting the Dirichlet regularization energy.

Optimizing As above, this energy is multi-quadratic in the coordinate functions of F . Thus, it can be optimized by alternately fixing all but one coordinate function f_i , so that the energy becomes quadratic in f_i , reducing the optimization to a linear solve.

Complexity of the solve Noting that the anti-symmetric tensor field $\nabla f_1 \wedge \dots \wedge \nabla f_{\bar{d}}$ is multi-linear in the coordinate functions, the energy $E(F)$ is a polynomial of degree $2\bar{d}$ in the coordinate functions. Thus, finding the minimizer of F requires solving a system of polynomial equations of degree $2\bar{d} - 1$, so that the problem becomes more difficult as the co-dimension is increased.

The case of $\bar{d} = 1$ For sPSR, the co-dimension is $\bar{d} = 1$ so that $\wedge^1 \mathbb{R}^n \approx \mathbb{R}^n$ and the exterior product acts as the identity. In this case the above expressions for the target vector-field and the energy reduce to those in Equations 2 and 3, demonstrating that the proposed approach generalizes sPSR. In particular, we have $2\bar{d} - 1 = 1$, corroborating the established result that sPSR reduces surface reconstruction to the solution of a *linear* system of equations. We also note that in the case of co-dimension one, orientable manifolds necessarily have trivial normal bundles so the requirement that the normal bundle be trivial does not introduce additional constraints.

6. Discretization and Solution

We next proceed to a discussion of discretization. We focus on the general problem in this section and discuss our implementation using the B-spline basis in Section 7.

6.1. Basis and operators

Basis

To discretize the energy in Equation 12 we need to define a discrete space of real-valued functions \mathcal{F} (for representing the solution coefficients f_1 and f_2) and a discrete space of skew-symmetric matrix fields \mathcal{W} (for representing the constraints defined by V_S). We do this by choosing a basis $\{\phi_i\}_{i=1}^N$ to span \mathcal{F} and then define the basis spanning \mathcal{W} in terms of the $\{\phi_i\}$.

In particular, given a basis $\{\phi_i\}_{i=1}^N$ of (weakly differentiable) functions, we define the skew-symmetric matrix fields:

$$\omega_{(i,j)} = \nabla \phi_i \wedge \nabla \phi_j.$$

By anti-symmetry, $\omega_{(i,j)} = -\omega_{(j,i)}$ so that we need only consider functions $\omega_{(i,j)}$ with $i < j$. Furthermore, we only need to consider indices (i,j) for which the supports of ϕ_i and ϕ_j overlap (otherwise $\omega_{(i,j)}$ is everywhere zero). Thus, we define:

$$\mathcal{I} = \{(i,j) \mid 1 \leq i < j \leq N, \text{supp}(\phi_i) \cap \text{supp}(\phi_j) \neq \emptyset\}$$

and take $\{\omega_i\}_{i \in \mathcal{I}}$ to be the basis spanning \mathcal{W} .

The advantage of this formulation is that the exterior product of the gradients of two functions in \mathcal{F} will reside in \mathcal{W} :

$$\nabla f_1 \wedge \nabla f_2 \in \mathcal{W}, \quad \forall f_1, f_2 \in \mathcal{F}.$$

For simplicity, we will refer to $\mathbf{f} = (f_1, \dots, f_N)^T \in \mathbb{R}^N$ equivalently as the coefficients of the function $\sum_i f_i \cdot \phi_i$ and as the function itself. Similarly we will refer to $\mathbf{w} \in \mathbb{R}^{|\mathcal{I}|}$ as both the coefficients of a skew-symmetric matrix field and the field itself.

Mass and stiffness matrices

Given the bases $\{\phi_i\}_{i=1}^N$ and $\{\omega_i\}_{i \in \mathcal{I}}$ we obtain the mass and stiffness matrices, $\mathbf{M}^{\mathcal{F}}, \mathbf{S}^{\mathcal{F}} \in \mathbb{R}^{N \times N}$ and $\mathbf{M}^{\mathcal{W}} \in \mathbb{R}^{|\mathcal{I}| \times |\mathcal{I}|}$, by integrating the products of basis functions (and their gradients):

$$\mathbf{M}_{ij}^{\mathcal{F}} = \langle \phi_i, \phi_j \rangle, \quad \mathbf{S}_{ij}^{\mathcal{F}} = \langle \nabla \phi_i, \nabla \phi_j \rangle, \quad \text{and} \quad \mathbf{M}_{ij}^{\mathcal{W}} = \langle \omega_i, \omega_j \rangle.$$

Additionally, given the samples \mathcal{S} , we set $\mathring{\mathbf{M}}^{\mathcal{F}} \in \mathbb{R}^{N \times N}$ to be the mass matrix defined by the samples:

$$\mathring{\mathbf{M}}_{ij}^{\mathcal{F}} = \sum_{\mathbf{s} \in \mathcal{S}} \phi_i(\mathbf{s}, \mathbf{p}) \cdot \phi_j(\mathbf{s}, \mathbf{p}).$$

The exterior product

Given $\mathbf{F} = (\mathbf{f}_1, \mathbf{f}_2) \in \mathbb{R}^{N \times 2}$, we denote by $\mathbf{W}_{\mathbf{F}} \in \mathbb{R}^{|\mathcal{I}|}$ the exterior product of the gradients of \mathbf{f}_1 and \mathbf{f}_2 :

$$(\mathbf{W}_{\mathbf{F}})_{(i,j)} = (\mathbf{f}_1)_i \cdot (\mathbf{f}_2)_j - (\mathbf{f}_1)_j \cdot (\mathbf{f}_2)_i.$$

Lastly, we denote by $[\mathbf{F}]_1 \in \mathbb{R}^{|\mathcal{I}| \times N}$ (resp. $[\mathbf{F}]_2 \in \mathbb{R}^{|\mathcal{I}| \times N}$) the matrix taking a function $\mathbf{f} \in \mathbb{R}^N$ and returning the exterior product of the gradient of \mathbf{f}_1 with the gradient of \mathbf{f} (resp. the gradient of \mathbf{f} with the gradient of \mathbf{f}_2):

$$([\mathbf{F}]_i)_{ij} = \begin{cases} (-1)^{i+0} \cdot (\mathbf{f}_i)_k & \text{if } \mathbf{i} = (j, k), \text{ for some } j < k \leq N \\ (-1)^{i+1} \cdot (\mathbf{f}_i)_k & \text{if } \mathbf{i} = (k, j), \text{ for some } 1 \leq k < j \\ 0 & \text{otherwise.} \end{cases}$$

The sign $(-1)^i$ is required to account for the position of the gradient of \mathbf{f} within the exterior product.

Prolongation

Next, We consider the case of nesting real-valued function spaces, $\widehat{\mathcal{F}} \subset \mathcal{F}$, with bases $\{\hat{\phi}_1, \dots, \hat{\phi}_{\hat{N}}\}$ and $\{\phi_1, \dots, \phi_N\}$, for which the nesting is given in terms of a prolongation matrix $\mathbf{P}^{\mathcal{F}} \in \mathbb{R}^{N \times \hat{N}}$. We show that despite the transition from a linear to a multilinear system, the classical Galerkin approach carries over in a straightforward manner (when solving in a strictly coarse-to-fine fashion).

As above, the basis for scalar fields $\{\hat{\phi}_i\}_{i=1}^{\hat{N}}$ defines a basis for skew-symmetric matrix fields $\{\hat{\omega}_i\}_{i \in \widehat{\mathcal{I}}}$. Furthermore, due to the bilinear relationship between the bases, the associated space of skew-symmetric matrix fields $\widehat{\mathcal{W}}$ is contained in \mathcal{W} , with the nesting relationship given by the prolongation matrix $\mathbf{P}^{\mathcal{W}} \in \mathbb{R}^{|\mathcal{I}| \times |\widehat{\mathcal{I}}|}$:

$$\mathbf{P}_{ij}^{\mathcal{W}} = \mathbf{P}_{i_1 j_1}^{\mathcal{F}} \cdot \mathbf{P}_{i_2 j_2}^{\mathcal{F}}, \quad \text{with } \mathbf{i} = (i_1, i_2), \mathbf{j} = (j_1, j_2).$$

Formally, this leverages the fact that for any linear map $P: \widehat{\mathcal{F}} \rightarrow \mathcal{F}$ there is a canonical multilinear map $\otimes^k P: \otimes^k \widehat{\mathcal{F}} \rightarrow \otimes^k \mathcal{F}$.

In what follows, we will overload notation, dropping the superscripts \mathcal{F} and \mathcal{W} in the mass, stiffness, and prolongation matrices when the distinction between \mathcal{F} and \mathcal{W} is contextually clear.

Observation For all $\widehat{\mathbf{F}} = (\hat{\mathbf{f}}_1, \hat{\mathbf{f}}_2) \in \mathbb{R}^{\hat{N} \times 2}$, prolonging and taking the exterior product of the gradients is equivalent to taking the exterior product of the gradients and then prolonging:

$$\mathbf{P} \cdot \mathbf{W}_{\widehat{\mathbf{F}}} = \mathbf{W}_{\mathbf{P} \cdot \widehat{\mathbf{F}}}. \quad (13)$$

6.2. Discretized energy

Given the target vector-field V_S the problem of solving for the coefficients function $\mathbf{f}_1, \mathbf{f}_2 \in \mathbb{R}^N$ becomes the problem of minimizing:

$$E(\mathbf{F}) = \mathbf{W}_{\mathbf{F}}^{\top} \cdot \mathbf{M} \cdot \mathbf{W}_{\mathbf{F}} - 2 \cdot \mathbf{W}_{\mathbf{F}}^{\top} \cdot \mathbf{b} + \|V_S\|^2 + \text{Tr} \left(\mathbf{F}^{\top} \cdot (\alpha \cdot \mathring{\mathbf{M}} + \beta \cdot \mathbf{S}) \cdot \mathbf{F} \right), \quad (14)$$

where $\mathbf{b} \in \mathbb{R}^{|\mathcal{I}|}$ is the dual representation of the target skew-symmetric matrix field:

$$\mathbf{b}_i = \langle \omega_i, V_S \rangle.$$

We note that although $\|V_S\|^2$ is not represented in our discretization (the target skew-symmetric field V_S need not reside in \mathcal{W}), its value does not affect the optimization since it is independent of \mathbf{F} .

Hierarchical system Given nesting function spaces $\widehat{\mathcal{F}} \subset \mathcal{F}$ we define the energy at the coarser resolution in the usual way — prolonging to the finer resolution and evaluating Equation 14:

$$E(\widehat{\mathbf{F}}) \equiv E(\mathbf{P} \cdot \widehat{\mathbf{F}}).$$

Leveraging the commutativity of the prolongation with the exterior derivative (Equation 13) and the fact that the multilinear mass matrix still satisfies the Galerkin property — that the coarse mass matrix is $\hat{\mathbf{M}} = \mathbf{P}^{\top} \cdot \mathbf{M} \cdot \mathbf{P}$ — it follows that:

$$E(\widehat{\mathbf{F}}) = \mathbf{W}_{\widehat{\mathbf{F}}}^{\top} \cdot \mathbf{M} \cdot \mathbf{W}_{\widehat{\mathbf{F}}} - 2 \cdot \mathbf{W}_{\widehat{\mathbf{F}}}^{\top} \cdot (\mathbf{P}^{\top} \cdot \mathbf{b}) + \|V_S\|^2 + \text{Tr} \left(\widehat{\mathbf{F}}^{\top} \cdot (\alpha \cdot \mathring{\mathbf{M}} + \beta \cdot \mathbf{S}) \cdot \widehat{\mathbf{F}} \right).$$

Comparing to Equation 14, we see that this is the energy we would have computed at the coarse resolution if we defined the coarse constraint vector as the restriction of the finer one, $\hat{\mathbf{b}} = \mathbf{P}^{\top} \cdot \mathbf{b}$. That is, the Galerkin formulation designed for the linear setting extends naturally to the multi-linear setting.

6.3. Relaxing the system

We leverage the fact that fixing \mathbf{f}_1 (resp. \mathbf{f}_2) the energy becomes quadratic in \mathbf{f}_2 (resp. \mathbf{f}_1). For example, fixing \mathbf{f}_1 the energy in \mathbf{f}_2 , up to a constant term, becomes:

$$E(\mathbf{f}_2) = \mathbf{f}_2^{\top} \cdot \left([\mathbf{F}]_1^{\top} \cdot \mathbf{M} \cdot [\mathbf{F}]_1 + \alpha \cdot \mathring{\mathbf{M}} + \beta \cdot \mathbf{S} \right) \cdot \mathbf{f}_2 - 2 \cdot \mathbf{f}_2^{\top} \cdot [\mathbf{F}]_1^{\top} \cdot \mathbf{b},$$

giving the optimal \mathbf{f}_2 as the solution:

$$\left([\mathbf{F}]_1^{\top} \cdot \mathbf{M} \cdot [\mathbf{F}]_1 + \alpha \cdot \mathring{\mathbf{M}} + \beta \cdot \mathbf{S} \right) \cdot \mathbf{f}_2 = [\mathbf{F}]_1^{\top} \cdot \mathbf{b}.$$

Algorithm 1 summarizes our approach for refining the estimate of the solution $\mathbf{F} = (\mathbf{f}_1, \mathbf{f}_2)$.

Algorithm 1: Relax

Require: $\mathbf{F} \in \mathbb{R}^{N \times 2}$; $\alpha, \beta \in \mathbb{R}$; $\mathring{\mathbf{M}}, \mathbf{S} \in \mathbb{R}^{N \times N}$; $\mathbf{M} \in \mathbb{R}^{|\mathcal{I}| \times |\mathcal{I}|}$; $\mathbf{b} \in \mathbb{R}^{|\mathcal{I}|}$

- 1: $\mathbf{f}_1 \leftarrow \left([\mathbf{F}]_2^{\top} \cdot \mathbf{M} \cdot [\mathbf{F}]_2 + \alpha \cdot \mathring{\mathbf{M}} + \beta \cdot \mathbf{S} \right)^{-1} \cdot [\mathbf{F}]_2^{\top} \cdot \mathbf{b}$
- 2: $\mathbf{f}_2 \leftarrow \left([\mathbf{F}]_1^{\top} \cdot \mathbf{M} \cdot [\mathbf{F}]_1 + \alpha \cdot \mathring{\mathbf{M}} + \beta \cdot \mathbf{S} \right)^{-1} \cdot [\mathbf{F}]_1^{\top} \cdot \mathbf{b}$

Hierarchical relaxation

Given nesting spaces of scalar fields $\mathcal{F}_0 \subset \dots \subset \mathcal{F}_L$ with associated prolongation matrices, we define corresponding nesting spaces of skew-symmetric matrix fields $\mathcal{W}_0 \subset \dots \subset \mathcal{W}_L$ with their associated prolongation matrices. Then, we relax the system in a coarse-to-fine manner, first by restricting the dual constraints \mathbf{b} to the coarsest resolution, then proceeding as in the upstroke of the standard (linear) multigrid algorithm [BHM00], relaxing at the current level and then up-sampling the estimated solution to serve as an initial guess for the next finer level, proceeding until the system is relaxed at the finest level. Algorithm 2 summarizes our approach for finding the minimizer of the energy.

Algorithm 2: RelaxHierarchical

Require: $\alpha, \beta \in \mathbb{R}; K, L \in \mathbb{N} \ S \subset \mathbb{R}^n \times (\mathbb{R}^n \times \mathbb{R}^n)$

- 1: $V_S \leftarrow \text{ConstructField}(S)$
- 2: $\mathbf{b}_L \leftarrow \text{Dual}(V_S)$
- 3: **for** $\ell \leftarrow L - 1 \dots 0$:
- 4: $\mathbf{P}_\ell^{\mathcal{W}} \leftarrow \text{SkewSymmetricProlongation}(2^\ell)$
- 5: $\mathbf{b}_\ell \leftarrow (\mathbf{P}_\ell^{\mathcal{W}})^\top \cdot \mathbf{b}_{\ell+1}$
- 6: $\mathbf{F}_0 \leftarrow \text{RandomInitialize}(2^0)$
- 7: **for** $\ell \leftarrow 0 \dots L$:
- 8: $\mathbf{S}_\ell^{\mathcal{F}} \leftarrow \text{ScalarStiffness}(2^\ell)$
- 9: $\hat{\mathbf{M}}_\ell^{\mathcal{F}} \leftarrow \text{ScalarSampleMass}(2^\ell, S)$
- 10: $\mathbf{M}_\ell^{\mathcal{W}} \leftarrow \text{SkewSymmetricMass}(2^\ell)$
- 11: **for** $k \leftarrow 1 \dots K$:
- 12: $\mathbf{F}_\ell \leftarrow \text{Relax}(\mathbf{F}_\ell, \alpha, \beta, \hat{\mathbf{M}}_\ell^{\mathcal{F}}, \mathbf{S}_\ell^{\mathcal{F}}, \mathbf{M}_\ell^{\mathcal{W}}, \mathbf{b}_\ell)$
- 13: **if** $\ell < L$:
- 14: $\mathbf{P}_\ell^{\mathcal{F}} \leftarrow \text{ScalarProlongation}(2^\ell)$
- 15: $\mathbf{F}_{\ell+1} \leftarrow \mathbf{P}_\ell^{\mathcal{F}} \cdot \mathbf{F}_\ell$
- 16: **Return** \mathbf{F}_L

As the energy is bi-quadratic, the implementation of a full multi-grid solver (i.e. fine-to-coarse, in addition to coarse-to-fine) is more challenging. We show in the Supplemental that this too can be done, allowing for the implementation of standard multi-cycle V-, W-, and F-cycle solvers.

7. Implementation

7.1. Choosing a basis

We discretize the space of scalar fields using the first-order B-spline basis indexed over the corners of a regular grid. We choose this basis for two reasons.

1. The regularity of the grid facilitates indexing and enables all of the linear operators (with the exception of the mass matrix $\hat{\mathbf{M}}$ defined by the samples) to be defined using a stencil, reducing the cost of system set-up.
2. The basis supports a hierarchical representation, with the space of scalar-fields defined over a grid of resolution $R \times \dots \times R$ nesting within the space of scalar fields defined over a grid of resolution $2R \times \dots \times 2R$. Denoting by $\mathbf{i} = (i_1, \dots, i_d)$ (resp. $\mathbf{j} = (j_1, \dots, j_d)$) the coordinates of a corner of a $2R \times \dots \times 2R$

(resp. $R \times \dots \times R$) grid, the prolongation matrix is defined as:

$$\mathbf{P}_{ij} = \prod_{l=1}^d \pi(i_l, j_l) \quad \text{with} \quad \pi(i, j) = \begin{cases} 1 & \text{if } |2j - i| = 0 \\ \frac{1}{2} & \text{if } |2j - i| = 1 \\ 0 & \text{otherwise.} \end{cases}$$

7.2. Defining the dual constraints

Given the samples, we define the target skew-symmetric matrix field V_S by first multi-linearly splatting the samples' skew-symmetric matrices into a grid. That is, for each $\mathbf{s} \in \mathcal{S}$ we find the grid cell containing \mathbf{s}, \mathbf{p} and distribute the value $\mathbf{s}, \mathbf{n}_1 \wedge \mathbf{s}, \mathbf{n}_2$ to the corners of the cell using the standard multi-linear interpolation weights. Then we perform two passes of smoothing, replacing the value at each corner with the average of the values in its one-ring. Finally, we set V_S to be the piecewise-constant skew-symmetric matrix field whose value within a cell is the average of the values at the cell's corners. Thus, considering each cell in turn, the computation of the dual constraints $\mathbf{b} \in \mathbb{R}^{|\mathcal{Z}|}$ reduces to the integration of multi-quadratic polynomials (the dot-product of ω_i with a constant skew-symmetric matrix) over the grid cells.

7.3. Accounting for non-uniform sampling

The splatting of samples to define V_S is done taking into account sampling density [KBH06]. We compute a piecewise-constant density estimator ρ as above: For each sample $\mathbf{s} \in \mathcal{S}$ splatting a value of 1 into the corners of the grid, performing two-passes of one-ring averaging, and setting the value in a cell to the average of the corners' values. In constructing the target skew-symmetric matrix field, we rescale $\mathbf{s}, \mathbf{n}_1 \wedge \mathbf{s}, \mathbf{n}_2$ to have Frobenius norm equal to $1/\rho(\mathbf{s}, \mathbf{p})$ — an estimate of the d -dimensional volume of sample \mathbf{s} — before splatting the exterior product into the grid.

7.4. Solving the system

We solve for \mathbf{F} using the hierarchical approach described in Algorithm 2. In Line 12 we invoke Algorithm 1 to alternately update the coordinate functions within a level. A naïve implementation would use a direct solver. We could leverage the hierarchy, using a standard V-cycle solver [BHM00] to efficiently approximate the solutions of the linear systems. However, we have observed that even this is not necessary as we already have a good initial guess for \mathbf{F}_ℓ obtained by prolonging the estimated solution from level $\ell - 1$ (Line 15). Thus, for a given level $0 \leq \ell \leq L$, the relaxation in Algorithm 1 can be done by performing a small number of Gauss-Seidel iterations. (We accelerate the computation by using multi-coloring to parallelize the relaxations.)

7.5. Extracting the level-set

Extracting the level-set from the regular grid is more complex than the standard Marching Cubes case [LC87, KKDH07] because the basis functions are multi-linear within a grid cell. We resolve this by using a piecewise-linear approximation — partitioning grid cells into simplices, sampling the original function at the vertices of the simplices, and linearly interpolating within a simplex. Our recursive strategy is an implementation of the generalization of an algorithm for 2-surfaces in 4 dimensions [WB96]. It is possible to

resolve the recursion [Min03]; and in higher dimension it would be beneficial to use triangulations with few simplices [BKW21]. We found that our approach was sufficiently efficient to explore the results with negligible computation times.

Simplicial decomposition

To partition each grid cell we proceed by induction on cell dimension. For dimensions $k = 0$ and $k = 1$ nothing needs to be done since the k -dimensional cells of the grid are already k -dimensional simplices. Starting with dimension $k = 2$, we introduce an additional vertex at the center of each k -dimensional cell, and connect the previously obtained $(k - 1)$ -dimensional boundary simplices to the center. For example, with $k = 2$ we add the center of each 2-dimensional face to the grid and then triangulate each 2D face by connecting its boundary edges/simplices to the face center, obtaining four triangles. Similarly, for each 3D cube, we introduce the center and then iterate over each 2D face, connecting the four previously obtained triangles to the cube's center to get 24 tets.

Marching simplices

Having a piecewise-linear representation of the functions over a simplicial complex we obtain the level-set by inducting on the dimension of the simplices.

Base case $k = \bar{d}$: We iterate over all k -simplices (triangles) in the simplicial complex, embed each in in $\mathbb{R}^{\bar{d}}$ using the functions' values at the vertices, and test if the embedded simplex contains the origin. If it does, we introduce an iso-vertex at the location on the simplex that maps to the origin.

Induction step: For each $(k + 1)$ -dimensional simplex, we iterate over its k -dimensional faces, merging the associated $(k - \bar{d})$ -dimensional level-set simplices into $(k - \bar{d} + 1)$ -dimensional level-set simplices. As the intersection of the level-set with a simplex is convex, merging can be done by choosing a vertex on a $(k - \bar{d})$ -dimensional level-set simplex and connecting it to all other $(k - \bar{d})$ -dimensional level-set simplices that do not contain that vertex.

8. Results and Evaluation

We now assess the empirical behavior of our solver. We start by considering results when reconstructing curves in 3D and surfaces in 4D. Next, we analyze the convergence of the solver. We conclude by examining the robustness in the presence of non-uniform sampling and noise. Source code implementing the reconstruction of co-dimension two manifolds has been made publicly available at <https://github.com/mkazhdan/ExteriorPoissonRecon>.

With the exceptions of the curve in Figure 1, point samples are obtained by uniformly randomly sampling parameter space (with 1024 samples for curves and 1024×1024 samples for surfaces) and visualizations show the entire zero level-set, including connected components over which the sampling density is low (in black). Curves are reconstructed at a resolution of $64 \times 64 \times 64$. Surfaces are reconstructed at a resolution of $32 \times 32 \times 32 \times 32$. Connected components are removed when all points on the component have a sampling density of less than two samples per cell.

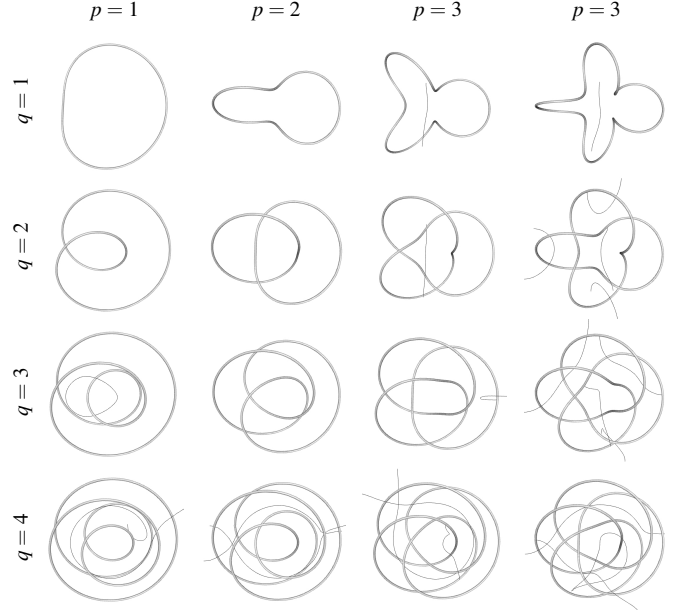


Figure 3: Reconstructed (p, q) torus-knots for varying values of integers p and q . (When p and q are not relatively prime, the curve consists of $\gcd(p, q)$ separate components. For example, the $(2, 2)$, $(3, 3)$, and $(4, 4)$ torus-knots are composed of multiple rotations of the $(1, 1)$ torus-knot.)

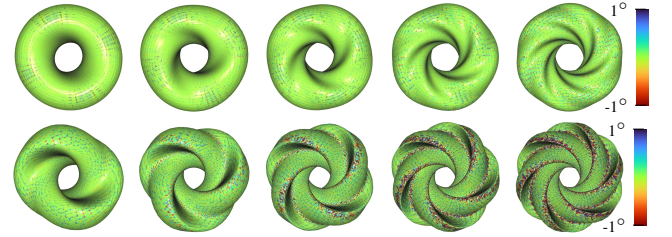


Figure 4: Reconstructed Hopf tori, with samples obtained using the parameterization from [Lau18] and visualized via stereographic projection, with different number of lobes and different frequency pre-image curves on the sphere. The tori are rendered with per-vertex Gaussian curvature (measured as the angular deficit in 4D divided by the vertex area).

8.1. Curve and surface reconstruction

Visualizations of various (p, q) torus-knots are shown in Figure 3 and several Hopf tori are visualized (after stereographic projection from the 3-sphere into Euclidean 3-space) in Figure 4.

For the curves, our method correctly reconstructs the geometry even when the curves consist of multiple disconnected components ($\gcd(p, q) > 1$) and are knotted ($p, q \neq 1$). While the spurious level-set components are generated for the more complex knots there are far from the input samples and are easily trimmed off.

For the tori we visualize the Gaussian curvature, computed per vertex in 4D as the angular deficit divided by the vertex area. We expect the curvature to be zero because the target surfaces are

	$d = 3$		$d = 4$	
	Time	Memory	Time	Memory
$R = 16$	0.4 (s)	14 (MB)	93.4 (s)	748 (MB)
$R = 32$	2.5 (s)	64 (MB)	1300.4 (s)	5522 (MB)
$R = 64$	21.5 (s)	420 (MB)	23,361.3 (s)	60,698 (MB)
$R = 128$	188.5 (s)	3331 (MB)	—	—
$R = 256$	1725.3 (s)	25,809 (MB)	—	—

Table 1: Run-time and peak memory usage for reconstructing curves in 3D and surfaces in 4D as a function of resolution, R . Results were obtained using an Intel Core i7 processor, running at 2.80 GHz, with 64 GB of RAM.

intrinsically flat. In general, we find this corroborated in our visualizations, though the tori exhibit more curvature variation as the number of lobes and the frequency of the pre-image curve on the sphere is increased, likely due to the low reconstruction resolution.

Computational complexity

Using a hierarchical solver with a fixed number of relaxations per level, the cost of the solve is linear in the number of non-zero entries in the system matrix. As we use a first-order B-spline basis to represent scalar functions, the number of non-zeros is proportional to the number of basis functions. Thus, using a resolution of R in d dimensions, the complexity (both temporal and spatial) of reconstructing the geometry is $O(R^d)$. Table 1 shows the empirical performance for reconstructions in 3D and 4D at different resolutions, which matches the theoretical complexity.

8.2. Solver performance

Unlike sPSR, the case of co-dimension larger than one requires solving a *multi*-quadratic system of equations. We examine the convergence of the solver by considering its behavior with respect to different optimization schemes and its stability with respect to choice of initial guess.

Convergence

We reduce the minimization of a multi-quadratic energy to the solution of linear systems by alternately locking one coordinate function and solving for the other. As the energy is not quadratic, the updating of the coordinate functions requires multiple passes (parameter K in Algorithm 2).

While a naïve approach would only iterate at the finest resolution, we found this to be ineffective for two reasons. First, as with general iterative Jacobi methods, convergence tends to be slow, with the local updates requiring many iterations for the constraints at one point to affect the solution at a point far away. More significantly, since the energy we minimize is not convex, naïve iteration can lead to the solution being trapped at *local* minima.

We argue that both problems can be addressed by leveraging the hierarchical structure of the B-spline basis. As with traditional multigrid, this improves convergence because iterations at coarser resolutions have more global effect. More significantly, the down-sampling of the system to coarser resolutions smooths out the energy landscape, allowing the solver to find a good solution without

getting trapped in local minima with narrow basins of convergence. (While this does not ensure we find the global minimum, it encourages the solution to be stable. A property we validate below.)

As the solution at a coarser level provides a good guess for the solution at the next finer level, we argue that the full power of a linear solver is not necessary and the finer solution can be obtained with a few passes of Gauss-Seidel relaxation. To this end we compare three implementations.

- **Single-solve:** *Solve* for the coordinate functions at the finest level of the hierarchy.
- **Multi-solve:** Processing the levels of the hierarchy in a coarse-to-fine manner, *solve* for the coordinate functions and prolong them to the next level.
- **Multi-relax:** Processing the levels of the hierarchy in a coarse-to-fine manner, *relax* the coordinate functions and prolong them to the next level.

The first two implementations require solving linear systems. As a direct solver would be too expensive, we take advantage of the multi-resolution hierarchy, using a standard V-cycle solver instead.

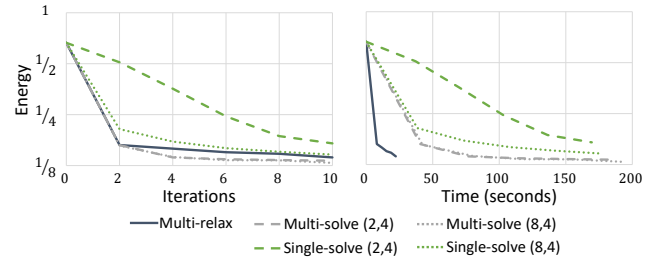


Figure 5: Energy as a function of alternating iteration passes (left) and time (right) for reconstructions of the (8,9) torus-knot at resolution $64 \times 64 \times 64$ with different solvers.

Figure 5 compares the performance of the different solvers, plotting the energy of the system as a function of the number of alternating relaxation passes and time when reconstructing the (8,9) torus-knot. For the **Single-solve** and **Multi-solve** implementations results are shown for V-cycle solvers using either 2 or 8 cycles, with 4 Gauss-Seidel relaxations per level.

Examining the non-hierarchical solution (**Single-solve**) we find that only alternating at the finest resolution does not converge efficiently and tends to converge to a solution with a larger energy (i.e. gets trapped at a local minimum). In contrast, both hierarchical solutions converge at roughly the same rate. As expected, relaxing the linearized problem by using a V-cycle solver (**Multi-solve**) converges in slightly fewer iterations than only relaxing the linearized problem within a single level (**Multi-relax**).

We also note that the hierarchical implementation of **Multi-solve** results in a good initial guess for the V-cycle solver so that convergence does not require a large number of cycles. In contrast, **Single-solve**, for which there is no initial guess, converges better as the number of V-cycles is increased.

Considering the energy as a function of time (right), we find a significant speed-up when using the **Multi-relax** implementation.

While there is an overhead to using a V-cycle solver, we believe that the increased computational cost is mostly due to the fact that the construction of the linear systems in Algorithm 1 requires multiplying and summing large matrices — a problem exacerbated by the difficulty of parallelizing these operations for sparse matrices.

We also explored gradient-descent and (quasi-)Newton methods. However, we found the behavior to be similar to that of the **Single-solve** approach — the approach required many iterations and would only converge to a local minimum that did not, in general, provide a good reconstruction. It is only by integrating a hierarchical approach that we were able to consistently obtain high-quality results.

Stability

As the energy is not convex, the solution we obtain may depend on the choice of the random initialization (Line 6, Algorithm 2). To assess this, we reconstructed the (8,9) torus-knot using twenty different initial guesses and computed the Hausdorff distance between each pair of reconstructed geometries. The distance never exceeded 0.11 voxels, indicating that all the iterations converged to the *same* solution. (The maximum distance was even smaller for the Clifford torus, likely due to the simplicity of the geometry.)

8.3. Robustness

We evaluate the behavior of our solver with respect to non-uniform sampling and different types of noise.

Non-uniform sampling

Throughout the paper, results are shown with samples generated by uniformly randomly sampling parameter space. To evaluate the performance of our reconstruction algorithm in the presence of non-uniform sampling we partitioned the embedding space with a hyperplane. Then for a given candidate sample (uniformly randomly sampled in parameter space), we randomly decide to keep the sample with a probability depending on which half of the hyperplane the sample belongs to.

Figure 6 shows a visualizations of the point samples generated for a curve consisting of two linked circles (top) and the associated reconstructions (bottom) for different ratios of samples on the two sides of the hyperplane. Even with a significant change in sampling density across the hyperplane our method correctly reconstructs the geometry of the curve, without generating spurious geometry.

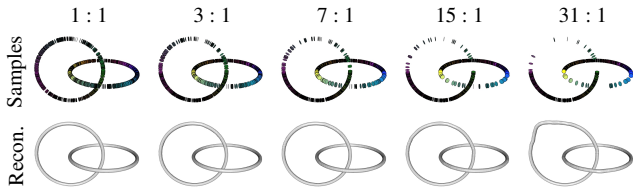


Figure 6: Reconstructions of the two linked circles with non-uniform sampling. Despite the abrupt change in sampling density, our approach correctly reconstructs the geometry of the curve, without introducing additional geometry.

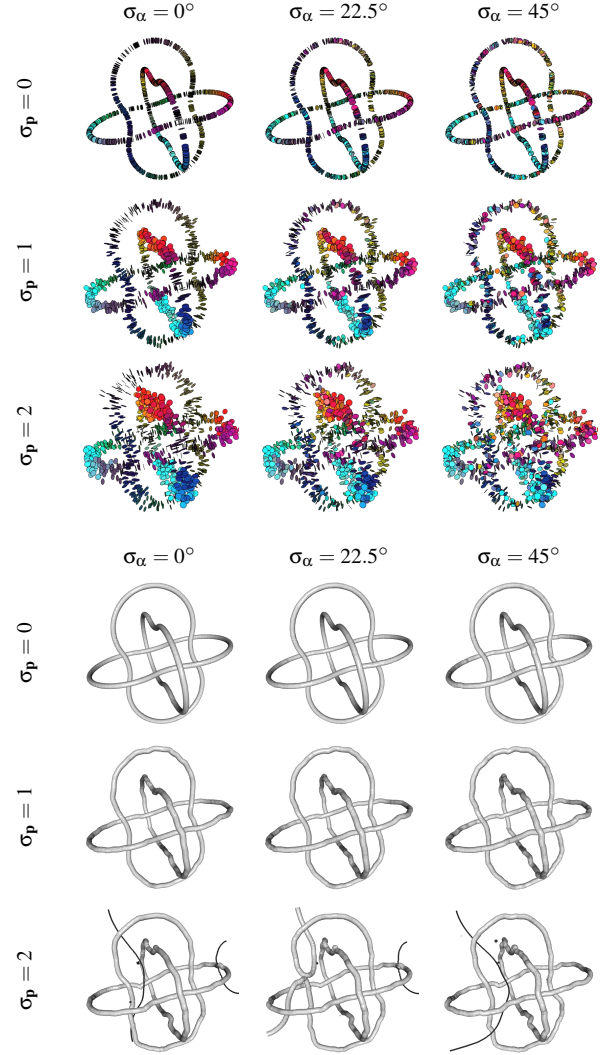


Figure 7: Reconstructions of the Borromean rings in the presence of noise (Top: samples, Bottom: reconstructions), for different deviations of position (σ_p) and normal orientation (σ_α). Positional displacements are in units of grid cell width. Note that for $\sigma_p = 1$ and $\sigma_\alpha = 22.5^\circ$, the ends of one of the rings do not close off and instead exit the bounding cube.

Positional and directional noise

To assess the robustness of our method in the presence of positional noise, we displaced a sample's position by a random 3D offset drawn from a normal distribution with standard deviation σ_p (in units of grid cell width). For directional noise, we rotated the sample's two normals using (separate) random rotations with angles drawn from a normal distribution with standard deviation σ_α .

Figure 7 shows reconstructions of the Borromean rings when positional and angular noise was added to the input samples while

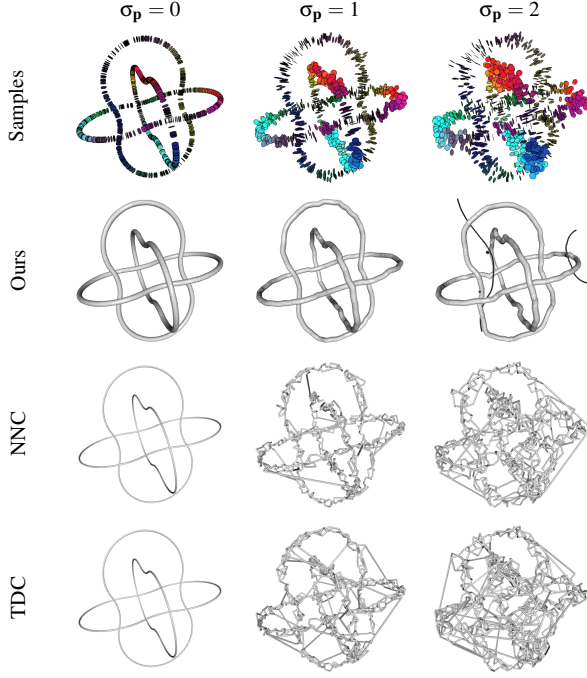


Figure 8: Reconstructions of the Borromean rings in the presence of positional noise (σ_p), comparing our approach to NN-Crust (NNC) and Tangential Delaunay Complexes (TDC). Positional displacements are in units of grid cell width.

Figure 8 compares our results to the results of Tangential Delaunay Complexes [BG10] and NN-Crust [DK99]. (Both earlier methods only use positional information.) Our results parallel those of earlier works on co-dimension 1 reconstruction (e.g. [Kaz05]), with the output remaining stable in the presence of directional noise, even as the angular deviation goes to 45° , and exhibit more robustness to positional noise than earlier methods.

A similar resilience to positional noise is observed when reconstructing surfaces in 4D. From left to right, the inset shows the stereographic projections of 10,000 noisy samples from a 3-lobed Hopf-torus, the computed Tangential Delaunay Complex, and our reconstruction at resolution $16 \times 16 \times 16 \times 16$. The standard deviation of the positional noise corresponds to approximately half a voxel width. Because it is interpolatory, the Tangential Delaunay complex reproduces the noise, outputting a non-manifold mesh for which almost 80% of the (projected) faces self-intersect. In contrast, our method generates a smooth and manifold mesh without any self-intersection. (As stereographic projection is not isometric, the samples are not uniform when mapped 3D.)

Spot noise

Finally, we evaluate the robustness of our method in the presence of spot noise by randomly synthesizing samples within the samples' bounding box. Figure 9 shows the results of one such experiment

for the reconstruction of the two linked circles. As the figure shows, our approach correctly reconstructs the geometry even as the number of noisy samples matches the number of curve samples, with spurious geometric components generated away from the samples removed using the density estimator. Only in the cases of extreme noise (right two columns) is spurious geometry not trimmed off, due to high sampling density away from the curve.

9. Discussion

We presented an extension of screened Poisson Surface Reconstruction, supporting the reconstruction of manifolds in general dimension and co-dimension. In doing so, we have shown that sPSR can be understood as the simplest instance of a broader class of variational problems that reduce manifold reconstruction to the solution of a polynomial system of equations, with polynomial degree growing with co-dimension. We have demonstrated that the approach works in co-dimension $\bar{d} = 2$, robustly reconstructing complex curves in 3D and surfaces in 4D.

As our approach is to solve for a \bar{d} -valued map whose zero-level-set is the desired manifold, our approach is restricted to reconstructing those manifolds with trivial normal bundles. However, knowing that a manifold *has* a trivial normal bundle does not specify what that trivialization is. Our approach takes in an anti-symmetric tensor at each sample point, describing the orientation of the manifold without prescribing the trivialization itself. As such, in solving for the implicit function, we implicitly select a trivialization of the normal bundle. In this context, one can view the incorporation of the Dirichlet energy as biasing the reconstruction towards the selection of a smooth trivialization.

Limitations: Trimming

As discussed in Section 5.1, given a closed manifold with trivial normal bundle, we are only guaranteed the existence of an implicit function $F : \mathbb{R}^n \rightarrow \mathbb{R}^{n-d}$ containing the manifold \mathcal{M} as a connected subset of the zero level-set. The implicit function F may have additional level-set components away from the target manifold, necessitating trimming.

The trimming value should depend on the (maximum) expected sampling density, which itself depends on the number of samples and the resolution of the grid. In our experiments we assume that (for each connected component of the manifold) there are at least two samples falling within the same grid cell and set the parameter to 2. This correctly removes all spurious geometry for results in Figures 1-8 and most of the results in Figure 9.

Limitations: Dirichlet Regularization

While empirically we have found that the regularizer in Equation 12 improves the quality of reconstruction, in principle it could preclude the generation of a good solution. This could be the case when the manifold exhibits complex twisting, requiring the coordinate functions of F to change quickly as well — a property in opposition to Dirichlet regularization.

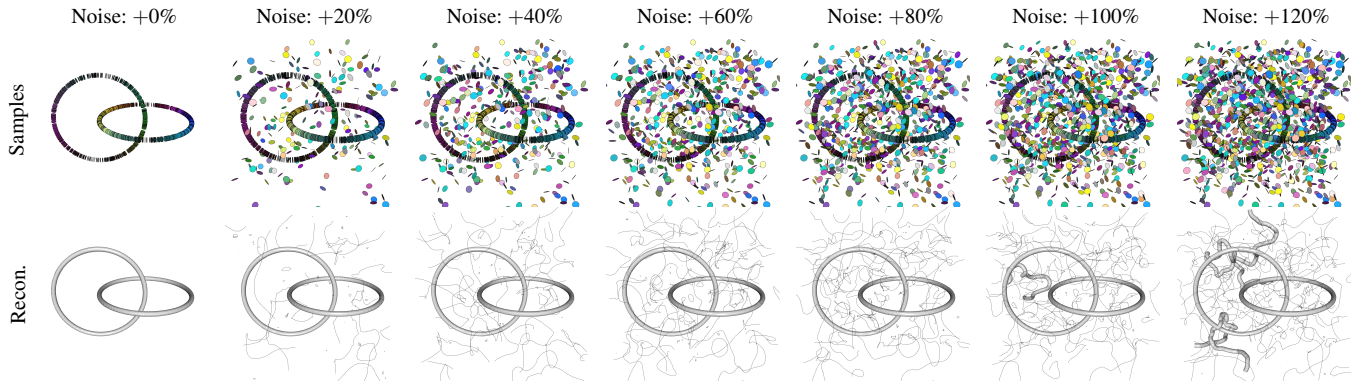
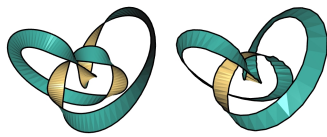


Figure 9: Reconstructions of the two linked circles using 1024 samples in the presence of added random noise. Reconstruction starts to break down when the size of the added random noise reaches the input sample size.

The inset shows an example of this for the trefoil knot, comparing the winding induced by correct coordinate functions [Sta12] on the left (the zero level-set equals the knot) with the winding induced by coordinate functions obtained using our approach on the right (the zero level-set contains spurious geometry that was trimmed off). Visualizing the gradient of one of the coordinate functions around the knot as a ribbon, we see that the correct solution introduces more torsion (with the ribbon winding three times around the curve) than our Dirichlet-regularized solution (for which the ribbon only winds around once).



Future work

We would like to continue exploring both theoretical and practical aspects of this work.

From the theoretical side, we would like to better understand the types of manifolds we can reconstruct and the energies we should incorporate to better guide us in finding “good” answers within the family of possible solutions. For example, we would like to consider replacing the global Dirichlet regularizer that is integrated over the entire domain with one that is only evaluated at the samples — encouraging the selection of a smooth normal frame for the manifold without otherwise constraining the implicit function. We would also like to consider incorporating the *quadratic* energy $\sum_i \|V_S \wedge \nabla f_i\|^2$ into our optimization. Though it does not encourage the gradients of the coordinate functions to span the normal space, it does encourage the gradients of the individual coordinate functions to lie in the normal space.

From the performance side, it would be natural to incorporate some of the machinery already established for sPSR. For example, using an octree discretization we should be able to reduce the complexity from $O(R^n)$ to $O(R^d)$.

Acknowledgements

The authors would like to express their sincere gratitude to Jordan Ellenberg, David Kazhdan, Jake Solomon, and Boris Springborn

for their repeated guidance and feedback. We would also like to thank reviewers for encouraging us to formalize the conditions on the target manifold implied by our approach.

This work was funded, in part, by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy–The Berlin Mathematics Research Center MATH+ (EXC-2046/1, project ID: 390685689); and by the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (Grant agreement No. 101055448, ERC Advanced Grant EMERGE).

References

- [AA06] ADAMSON A., ALEXA M.: Point-sampled cell complexes. *ACM Trans. Graph.* 25, 3 (jul 2006), 671–680. doi:10.1145/1141911.1141940. 2
- [AB99] AMENTA N., BERN M.: Surface reconstruction by Voronoi filtering. *Discrete and Computational Geometry* 22 (1999), 481–504. 2
- [ABCO*03] 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 (2003), 3–15. doi:10.1109/TVCG.2003.1175093. 2
- [ACSTD07] ALLIEZ P., COHEN-STEINER D., TONG Y., DESBRUN M.: Voronoi-based variational reconstruction of unoriented point sets. *Symposium on Geometry processing* (2007), 39–48. 2
- [AK85] AKBULUT S., KING H.: Submanifolds and homology of non-singular real algebraic varieties. *American Journal of Mathematics* 107 (1985), 45–83. 4
- [Aur87] AURENHAMMER F.: Power diagrams: properties, algorithms and applications. *SIAM Journal on Computing* 16, 1 (1987), 78–96. 2
- [BF04] BOISSONNAT J.-D., FLÖTTO J.: A coordinate system associated with points scattered on a surface. *Computer-Aided Design* 36, 2 (2004), 161–174. Solid Modeling and Applications. doi:10.1016/S0010-4485(03)00059-9. 2
- [BG10] BOISSONNAT J.-D., GHOSH A.: Manifold reconstruction using tangential delaunay complexes. In *Proceedings of the Twenty-Sixth Annual Symposium on Computational Geometry* (New York, NY, USA, 2010), SoCG ’10, Association for Computing Machinery, p. 324–333. doi:10.1145/1810959.1811013. 2, 12
- [BHM00] BRIGGS W. L., HENSON V. E., MCCORMICK S. F.: *A Multi-grid Tutorial*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2000. 3, 8

- [BKW21] BOISSONNAT J.-D., KACHANOVICH S., WINTRAECKEN M.: Tracing Isomanifolds in \mathbb{R}^d in Time Polynomial in d Using Coxeter-Freudenthal-Kuhn Triangulations. In *37th International Symposium on Computational Geometry (SoCG 2021)* (Dagstuhl, Germany, 2021), Buchin K., Colin de Verdière E., (Eds.), vol. 189 of *Leibniz International Proceedings in Informatics (LIPIcs)*, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, pp. 17:1–17:16. doi:10.4230/LIPIcs.SoCG.2021.17. 9
- [BTS*14] BERGER M., TAGLIASACCHI A., SEVERSKY L. M., ALLIEZ P., LEVINE J. A., SHARF A., SILVA C. T.: State of the Art in Surface Reconstruction from Point Clouds. In *Eurographics 2014 - State of the Art Reports* (2014), Lefebvre S., Spagnuolo M., (Eds.), The Eurographics Association, pp. 161–185. doi:10.2312/egst.20141040. 2
- [BWR*08] BERGOU M., WARDETZKY M., ROBINSON S., AUDOLY B., GRINSPUN E.: Discrete elastic rods. In *ACM SIGGRAPH 2008 Papers* (2008), SIGGRAPH '08, Association for Computing Machinery. 5
- [CBC*01] CARR J., BEATSON R., CHERRIE H., MITCHELL T., FRIGHT W., MCCALLUM B., EVANS T.: Reconstruction and representation of 3D objects with radial basis functions. *ACM SIGGRAPH* (2001), 67–76. 3
- [CC19] CHENG S.-W., CHIU M.-K.: Implicit manifold reconstruction. *Discrete & Computational Geometry* 62, 3 (Oct 2019), 700–742. doi:10.1007/s00454-019-00095-w. 2
- [CDE*00] CHENG S.-W., DEY T. K., EDELSBRUNNER H., FACELLO M. A., TENG S.-H.: Sliver exudation. *J. ACM* 47, 5 (Sept. 2000), 883–904. doi:10.1145/355483.355487. 2
- [CDR05] CHENG S.-W., DEY T. K., RAMOS E. A.: Manifold reconstruction from point samples. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms* (USA, 2005), SODA '05, Society for Industrial and Applied Mathematics, p. 1018–1027. 2
- [CL96] CURLESS B., LEVOY M.: A volumetric method for building complex models from range images. *ACM SIGGRAPH* (1996), 303–312. 3
- [CnRT*00] CAÑERO C., RADEVA P., TOLEDO R., VILLANUEVA J., MAURI J.: 3d curve reconstruction by biplane snakes. In *Proceedings of the International Conference on Pattern Recognition* (USA, 2000), vol. 4 of *ICPR '00*, IEEE Computer Society, pp. 563–566. doi:10.1109/ICPR.2000.902982. 2
- [CO08] CHAZAL F., OUDOT S. Y.: Towards persistence-based reconstruction in euclidean spaces. In *Proceedings of the Twenty-Fourth Annual Symposium on Computational Geometry* (New York, NY, USA, 2008), SCG '08, Association for Computing Machinery, p. 232–241. doi:10.1145/1377676.1377719. 2
- [CT11] CALAKLI F., TAUBIN G.: SSD: Smooth signed distance surface reconstruction. *Computer Graphics Forum* 30, 7 (2011), 1993–2002. 3
- [Dey06] DEY T. K.: Curve and surface reconstruction: Algorithms with mathematical analysis (cambridge monographs on applied and computational mathematics), 2006. 2
- [DG03] DONOHO D. L., GRIMES C.: Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences* 100, 10 (2003), 5591–5596. doi:10.1073/pnas.1031596100. 2
- [DGGZ02] DEY T. K., GIESEN J., GOSWAMI S., ZHAO W.: Shape dimension and approximation from samples. In *Proceedings of the Thirteenth Annual ACM-SIAM Symposium on Discrete Algorithms* (USA, 2002), SODA '02, Society for Industrial and Applied Mathematics, p. 772–780. 2
- [dGMM14] DE GOES F., MEMARI P., MULLEN P., DESBRUN M.: Weighted triangulations for geometry processing. *ACM Trans. Graph.* 33, 3 (June 2014). URL: <https://doi.org/10.1145/2602143>, doi:10.1145/2602143. 2
- [DK99] DEY T. K., KUMAR P.: A simple provable algorithm for curve reconstruction. In *Proceedings of the Tenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 17-19 January 1999, Baltimore, Maryland, USA (1999), Tarjan R. E., Warnow T. J., (Eds.), ACM/SIAM, pp. 893–894. URL: <http://dl.acm.org/citation.cfm?id=314500.315073>. 12
- [FIK*18] FEFFERMAN C., IVANOV S., KURYLEV Y., LASSAS M., NARAYANAN H.: Fitting a putative manifold to noisy data. In *Proceedings of the 31st Conference On Learning Theory* (06–09 Jul 2018), Bubeck S., Perchet V., Rigollet P., (Eds.), vol. 75 of *Proceedings of Machine Learning Research*, PMLR, pp. 688–720. URL: <https://proceedings.mlr.press/v75/fefferman18a.html>. 2
- [Fre02] FREEDMAN D.: Efficient simplicial reconstructions of manifolds from their samples. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24, 10 (2002), 1349–1357. doi:10.1109/TPAMI.2002.1039206. 2
- [GP10] GUILLEMIN V., POLLACK A.: *Differential topology*. AMS Chelsea Publishing, 2010. 4
- [GPPVW14] GENOVESE C. R., PERONE-PACIFICO M., VERDINELLI I., WASSERMAN L.: Nonparametric ridge estimation. *The Annals of Statistics* 42, 4 (2014), 1511–1545. URL: <https://doi.org/10.1214/14-AOS1218>, doi:10.1214/14-AOS1218. 2
- [HDD*92] HOPPE H., DEROSE T., DUCHAMP T., McDONALD J., STUETZLE W.: Surface reconstruction from unorganized points. *ACM SIGGRAPH* (1992), 71–78. 3
- [Kaz05] KAZHDAN M.: Reconstruction of Solid Models from Oriented Point Sets. In *Eurographics Symposium on Geometry Processing 2005* (2005), Desbrun M., Pottmann H., (Eds.), The Eurographics Association. doi:10.2312/SGP/SGP05/073-082. 12
- [KBH06] KAZHDAN M., BOLITHO M., HOPPE H.: Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2006), SGP '06, Eurographics Association, pp. 61–70. 1, 8
- [KH13] KAZHDAN M., HOPPE H.: Screened poisson surface reconstruction. *ACM Trans. Graph.* 32, 3 (jul 2013). doi:10.1145/2487228.2487237. 1, 3
- [Kir89] KIRBY R. C.: *The Topology of 4-Manifolds*. Springer Berlin, Heidelberg, 1989. doi:10.1007/BFb0089031. 4
- [KKDH07] KAZHDAN M., KLEIN A., DALAL K., HOPPE H.: Unconstrained isosurface extraction on arbitrary octrees. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing* (Goslar, DEU, 2007), SGP '07, Eurographics Association, p. 125–133. 3, 8
- [KSO04] KOLLURI R., SHEWCHUK J., O'BRIEN J.: Spectral surface reconstruction from noise point clouds. *Symposium on Geometry Processing* (2004), 11–21. 2
- [Lau18] LAURENT S.: Parametric hopf torus, 2018. URL: <https://laustep.github.io/stlahblog/posts/HopfTorusParametric.html>. 9
- [LC87] LORENSEN W. E., CLINE H. E.: Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1987), SIGGRAPH '87, ACM, pp. 163–169. doi:10.1145/37401.37422. 3, 8
- [Lee00] LEE I.-K.: Curve reconstruction from unorganized points. *Computer Aided Geometric Design* 17, 2 (2000), 161–177. doi:10.1016/S0167-8396(99)00044-8. 2
- [Lee12] LEE J. M.: *Introduction to Smooth Manifolds*. Springer New York, NY, 2012. doi:10.1007/978-1-4419-9982-5. 4
- [Lev04] LEVIN D.: Mesh-independent surface interpolation. In *Geometric Modeling for Scientific Visualization* (Berlin, Heidelberg, 2004), Brunnert G., Hamann B., Müller H., Linsen L., (Eds.), Springer Berlin Heidelberg, pp. 37–49. 2
- [Mil65] MILNOR J.: *Topology from the Differentiable Viewpoint*. University Press of Virginia, 1965. 4, 15

- [Min03] MIN C.: Simplicial isosurfacing in arbitrary dimension and codimension. *Journal of Computational Physics* 190, 1 (2003), 295–310. doi:[https://doi.org/10.1016/S0021-9991\(03\)00275-4](https://doi.org/10.1016/S0021-9991(03)00275-4). 9
- [MPS08] MANSON J., PETROVA G., SCHAEFER S.: Streaming surface reconstruction using wavelets. *Symposium on Geometry Processing* (2008), 1411–1420. 3
- [NIH*11] NEWCOMBE R. A., IZADI S., HILLIGES O., MOLYNEAUX D., KIM D., DAVISON A. J., KOHLI P., SHOTTON J., HODGES S., FITZGIBBON A.: Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality* (USA, 2011), ISMAR '11, IEEE Computer Society, p. 127–136. 3
- [OE11] OZERTEM U., ERDOGMUS D.: Locally defined principal curves and surfaces. *Journal of Machine Learning Research* 12, 34 (2011), 1249–1286. URL: <http://jmlr.org/papers/v12/ozertem11a.html>. 2
- [OPP*21] OHRHALLINGER S., PEETHAMBARAN J., PARAKKAT A. D., DEY T. K., MUTHUGANAPATHY R.: 2d points curve reconstruction survey and benchmark. *Computer Graphics Forum* 40, 2 (2021), 611–632. doi:<https://doi.org/10.1111/cgf.142659>. 2
- [PL03] POTTMANN H., LEOPOLDSER S.: A concept for parametric surface fitting which avoids the parametrization problem. *Computer Aided Geometric Design* 20, 6 (2003), 343–362. doi:[10.1016/S0167-8396\(03\)00078-5](https://doi.org/10.1016/S0167-8396(03)00078-5). 2
- [PLH02] POTTMANN H., LEOPOLDSER S., HOFER M.: Approximation with active b-spline curves and surfaces. In *10th Pacific Conference on Computer Graphics and Applications, 2002. Proceedings.* (2002), pp. 8–25. doi:[10.1109/PCCGA.2002.1167835](https://doi.org/10.1109/PCCGA.2002.1167835). 2
- [SKG15] SAINI D., KUMAR S., GULATI T. R.: Reconstruction of free-form space curves using nurbs-snakes and a quadratic programming approach. *Computer Aided Geometric Design* 33 (2015), 30–45. doi:[10.1016/j.cagd.2015.01.001](https://doi.org/10.1016/j.cagd.2015.01.001). 2
- [SL20] SOBER B., LEVIN D.: Manifold approximation by moving least-squares projection (mmls). *Constructive Approximation* 52, 3 (2020), 433–478. doi:[10.1007/s00365-019-09489-8](https://doi.org/10.1007/s00365-019-09489-8). 2
- [Sta12] STACKEXCHANGE: Trefoil knot as an algebraic curve, 2012. URL: <https://math.stackexchange.com/questions/148916/trefoil-knot-as-an-algebraic-curve>. 13
- [SW04] SCHAEFER S., WARREN J.: Dual marching cubes: primal contouring of dual grids. In *12th Pacific Conference on Computer Graphics and Applications, 2004. PG 2004. Proceedings.* (2004), pp. 70–76. doi:[10.1109/PCCGA.2004.1348336](https://doi.org/10.1109/PCCGA.2004.1348336). 3
- [WB96] WEIGLE C., BANKS D. C.: Complex-valued contour meshing. In *Proceedings of Seventh Annual IEEE Visualization '96* (1996), pp. 173–180. doi:[10.1109/VISUAL.1996.568103](https://doi.org/10.1109/VISUAL.1996.568103). 8

Since $R(\infty) = R(\mathbf{0}) = \mathbf{0}$ and $\pi(\mathbf{0}) = \mathbf{0}$, the zero level-set of F is:

$$F^{-1}(\mathbf{0}) = \tilde{F}^{-1}(\mathbf{0}) \cup \tilde{F}^{-1}(\infty).$$

Since \tilde{F} is continuous, it follows that $\tilde{F}^{-1}(\infty)$ is disconnected from $\tilde{F}^{-1}(\mathbf{0}) = \mathcal{M}$ as desired.

Appendix A: Extending Milnor's Construction

In [Mil65, Ch. 7, Thm. C], Milnor shows that if $\mathcal{M} \subset \mathbb{R}^n$ is a closed, $(n-d)$ -dimensional, embedded manifold with trivial normal bundle, it is possible to construct a continuous implicit function $\tilde{F} : \mathbb{R}^n \rightarrow S^{n-d}$ such that $\tilde{F}^{-1}(\mathbf{0}) = \mathcal{M}$. It is straight-forward to turn such a function into a function $F : \mathbb{R}^n \rightarrow \mathbb{R}^{n-d}$ containing \mathcal{M} as a disconnected subset of $F^{-1}(\mathbf{0})$. (With some abuse of notation, we use “ $\mathbf{0}$ ” to denote both the South pole in S^{n-d} and the origin in \mathbb{R}^{n-d} , and we use “ ∞ ” to denote the North pole in S^{n-d} .)

To see this, set $F = \pi \circ R \circ \tilde{F}$ with $\pi : S^{n-d} \rightarrow \mathbb{R}^{n-d}$ the stereographic projection and $R : S^{n-d} \rightarrow S^{n-d}$ the map that fixes the lower hemisphere and reflects the upper hemisphere into the lower hemisphere — $R(x_1, \dots, x_{n-d}, x_{n-d+1}) = (x_1, \dots, x_{n-d}, -|x_{n-d+1}|)$.