Interpolation on manifolds using Bézier functions

Pierre-Yves Gousenbourger[1], P.-A. Absil[1], Benedikt Wirth[2] and Laurent Jacques[1].

[1] Université catholique de Louvain - ICTEAM Institute, B-1348 Louvain-la-Neuve, Belgium

[2] University of Münster - Institute for Numerical and Applied Mathematics, Einsteinstraße 62, D-48149 Münster, Germany

***Abstract* – Given a set of data points lying on a smooth manifold, we present methods to interpolate those with piecewise Bézier splines. The spline is composed of Bézier curves (resp. surfaces) patched together such that the spline is continuous and differentiable at any point of its domain. The spline is optimized such that its mean square acceleration is minimized when the manifold is the Euclidean space. We show examples on the sphere S^2 and on the special orthogonal group $SO(3)$.**

1 Introduction

Given a set of data points (p_k) in a manifold $\mathcal{M}$ associated to nodes $(k) \in \mathbb{Z}^s$ of a Cartesian grid in $\mathbb{R}^s$, we seek a $\mathcal{C}^1$ function $\mathfrak{B} : \mathbb{R}^s \to \mathcal{M}$ such that $\mathfrak{B}(k) = p_k$. In this paper, we consider the cases where $s \in \{1, 2\}$, *i.e.* curves and surfaces.

This interpolation problem is motivated by several applications, as more and more acquired data tends to be constrained to smooth manifolds. In Pennec *et al.* [8], diffusion tensor images are assumed to lie on the manifold of positive definite matrices. Bergmann and Weinmann [4] propose inpainting models for images whose color pixels lie on the 2-sphere.

Manifold modeling appears in many other different fields including image modeling and processing [9] or optimization [1]. Its advantages are that *(i)* problem solutions are searched on spaces of much lower dimensionality compared to the ambient domain, with a direct and positive impact on algorithm performances in computational time, memory and accuracy; *(ii)* complex objects are represented by vectors of small size; and *(iii)* close formulas can be found for problems on manifolds while complex iterative procedures are required on Euclidean spaces.

Interpolation on manifold does not appear much in the literature. Popiel and Noakes [10] propose a manifold version of Bézier curves based on the work on $\mathbb{R}^r$ of Farin [6]. Boumal *et al.* proposed optimization algorithms for curve fitting on manifolds with the toolbox Manopt [5]. More recently, Solomon *et al.* [11] developed a Wasserstein distance-based method to interpolate probability distributions evaluated on manifolds.

In this paper, we summarize different techniques to interpolate data points on manifolds by means of a $\mathcal{C}^1$ piecewise-cubic Bézier spline [7, 3, 2], as illustrated in Figure 1. In Sec. 2, we define Bézier splines on manifolds and give continuity and differentiability conditions. We show in Sec. 3 how these splines can be optimized to have a small global energy. We present numerical examples in Sec. 4.

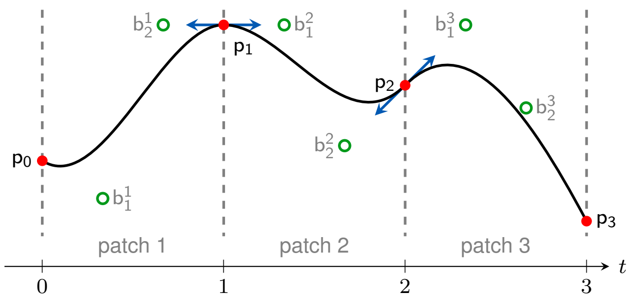

Figure 1: 1-D schematic representation of a three-pieces cubic Bézier spline with its continuity and differentiability constraints.

2 Bézier curves and surfaces

We first recall the definitions of Bézier curves and surfaces on the Euclidean space and we generalize them to manifolds. Then, we give conditions to achieve interpolation, continuity and derivability of the spline.

Curves and surfaces on the Euclidean space $\mathbb{R}^r$. Let $B_{k3}(t)$ be Bernstein polynomials and $\mathbf{b} = (b_0, \ldots, b_3) \subset \mathbb{R}^r$ be a set of *control points*. Cubic Bézier *curves* are functions $\beta_3(\cdot, \mathbf{b}) : [0, 1] \to \mathbb{R}^r$ of the form

$$\beta_3(t; \mathbf{b}) \mapsto \textstyle\sum_{i=0}^{3} b_i B_{i3}(t). \tag{1}$$

Cubic Bézier *surfaces* $\beta_3(\cdot, \mathbf{b}) : [0, 1]^2 \to \mathbb{R}^r$ are their bivariate extensions with $\mathbf{b} = (b_{ij})_{i,j=0,\ldots,3} \subset \mathbb{R}^r$

$$\beta_3(t_1, t_2; \mathbf{b}) = \textstyle\sum_{i,j=0}^{3} b_{ij} B_{i3}(t_1) B_{j3}(t_2), \tag{2}$$

Control points b_{ij} are interpolated when $i, j \in \{0, 1\}$.

Curves on a manifolds. We generalize (1) to a smooth, connected, finite-dimentional Riemannian manifold $\mathcal{M}$ (where $\mathcal{M} = \mathbb{R}^r$ is included) in two different ways. First, since Bernstein polynomials form a partition of unity, one remarks that the Bézier curves are a weighted average of the control points

$$\beta_3(\cdot; \mathbf{b}) : [0, 1] \to \mathcal{M}, t \mapsto \mathrm{av}[(b_i)_{i=0,\ldots,3}, (B_{i3}(t))_{i=0,\ldots,3}], \tag{3}$$

where $\mathrm{av}[(y_0, \ldots, y_n), (w_0, \ldots, w_n)]$ is the weighted geodesic average $x = \mathrm{argmin}_y \sum_{i=0}^{n} w_i \mathrm{d}^2(y_i, y)$ with the geodesic distance d. Necesseraly, when d is the Euclidean distance, definition (3) reduces to equation (1). This model is introduced in Absil *et al.* [2].

The second generalization of (1) to the manifold setting is based on the De Casteljau algorithm (see Farin [6] for details in $\mathbb{R}^r$) where the Euclidean straight line is replaced by geodesics. Popiel and Noakes [10] proposed a manifold version of the algorithm and Arnould *et al.* [3] applied it to shape analysis.

Surfaces on manifolds. Similarly, we generalize (2) to $\mathcal{M}$ in three different ways. First, Bézier surfaces can be interpreted as a one-parameter family of Bézier curves

$$\begin{aligned} \beta_3(t_1, t_2; \mathbf{b}) &= \textstyle\sum_{j=0}^{3} \left(\sum_{i=0}^{3} b_{ij} B_{i3}(t_1) \right) B_{j3}(t_2) \\ &= \beta_3(t_2; (\beta_3(t_1; \mathbf{b}_j))_{j=0,\ldots,3}), \end{aligned}$$

where $\mathbf{b}_j = (b_{ij})_{i=0,\ldots,3}$. That formulation allows an evaluation based on curves as stated above.

A second interpretation of surfaces extends equation (3) as Bézier surfaces are convex combinations of their control points

$$\beta_3(t_1, t_2; \mathbf{b}) = \mathrm{av}[(b_{ij})_{i,j=0,\ldots,3}, (B_{i3} B_{j3})_{i,j=0,\ldots,3}]. \tag{4}$$

Here again, the Euclidean Bézier surface is recovered with the classical Euclidean averaging.

The last generalization is also based on a geodesic extension of the De Casteljau algorithm (Farin [6]). All these methods are developed in Absil *et al.* [2].

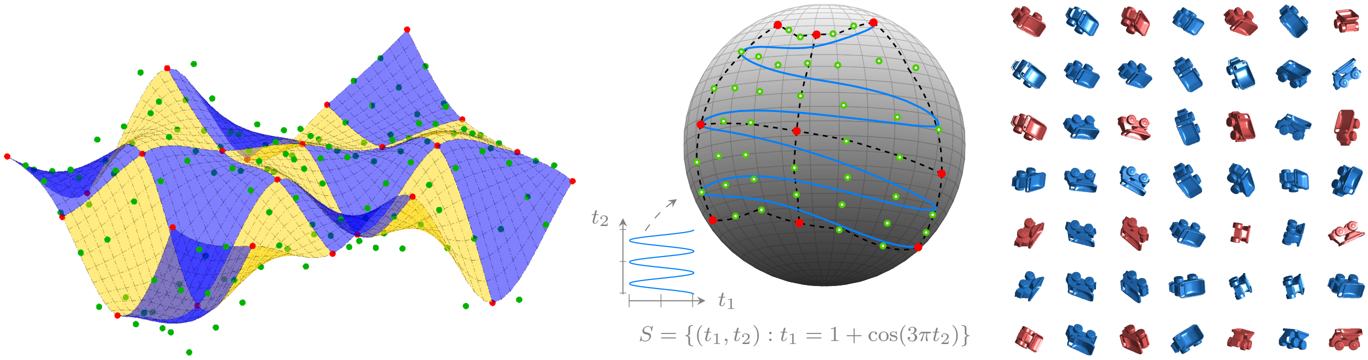

Figure 2: Differentiable piecewise-cubic Bézier surfaces. *Left:* $\mathcal{M} = \mathbb{R}^3$. The Bézier spline is composed of different Bézier surfaces (blue and yellow). It interpolates the (red) data points and is parametrized by the (green) control points. *Middle:* $\mathcal{M} = S^2$. Data points (red) are interpolated on the 2-sphere by a cubic Bézier spline. The (blue) curve is a smooth 1D-path on the surface. *Right:* $\mathcal{M} = SO(3)$. Data points (red) are orientations of a rigid body, represented as points on the special orthogonal group. The (blue) surface interpolates them.

Smooth splines on manifolds. A Bézier spline corresponds to several Bézier curves $(\beta_3^m)_{m=0,\dots,M}$ (or surfaces $(\beta_3^{mn})_{m,n\in\{0,\dots,M\}\times\{0,\dots,N\}}$) patched together (see Figure 1 for an example). For curves, the spline is continuous and interpolates the data points if $b_3^m = b_0^{m+1} = p_{m+1}$ for $m = 1,\dots,M-1$. Surfaces are continuously patched if their control points are the same at the shared border. Data points p_{mn} are interpolated if $b_{00}^{m,n} = p_{m,n}$

Differentiability of curves is achieved by constraining p_m to be in the middle of the geodesic between b_2^m and b_1^{m+1}. For surface, an equivalent bidimensional constraint exists on the Euclidean space, but its manifold version does not hold for surfaces: Absil *et al.* [2] hence introduced a modified definition of the Bézier surfaces such that $\mathcal{C}^1$ splines can be computed on any Riemannian manifold. Setting $\mathbf{b} = (b_{ij})_{i,j\in\mathcal{I}}$ and $\mathcal{I} = \{-1, 1, 2, 4\}$, one redefines

$$\beta_3(t_1, t_2; \mathbf{b}) = \text{av}\left[\mathbf{b}, (w_i(t_1)w_j(t_2))_{i,j\in\mathcal{I}}\right] \tag{5}$$

with slightly modified weights

$$w_i(t) = \begin{cases} \frac{1}{2}B_{03}(t) & \text{if } i = -1, \\ B_{13}(t) + \frac{1}{2}B_{03}(t) & \text{if } i = 1, \\ B_{23}(t) + \frac{1}{2}B_{33}(t) & \text{if } i = 2, \\ \frac{1}{2}B_{33}(t) & \text{if } i = 4. \end{cases}$$

3 Optimal splines

In Section 2, we showed that Bézier splines offer the possibility to interpolate a set of data points. In this section, we show how the control points of the Bézier curves (resp. surfaces) can be optimally chosen to obtain a "good-looking" spline.

To do so, we first optimize the control points of the spline in $\mathcal{M} = \mathbb{R}^r$ such that (i) the energy of the spline is minimum (*i.e.* the sum among all patches of the mean square acceleration of Bézier functions $\int_{[0,1]^s} \|\ddot{\beta}_3^m(\mathbf{t}, \mathbf{b})\|^2 \mathrm{d}\mathbf{t}$) and (ii) the spline is continuous, differentiable and interpolates the data points. The solution of this optimization problem reduces to a linear system. We then generalize this solution to manifolds.

In Gousenbourger *et al.* [7], the control points minimize

$$f[\mathbf{b}^m] = \hat{F}[\beta_2^0] + \textstyle\sum_{m=1}^{M-2} \hat{F}[\beta_3^m] + \hat{F}[\beta_2^{M-1}].$$

The points b_2^m, p_m and b_1^{m+1}, $m = 1,\dots,M-1$, are aligned on a geodesic whose direction is arbitrarily fixed. The length of these geodesics, however, are not specified and can be optimized.

In Arnould *et al.* [3], the constraint are relaxed and rewritten in terms of the control points only. In other words, on $\mathbb{R}^r$, we set $b_1^m = 2p_m - b_2^{m-1}$. After optimization, b_2^{m-1} appears to be a weighted sum of the data points. Due to translation invariance, we generalize the result to manifolds using the inverse exponential map. The exponential map and its inverse are operators of differential geometry. Examples of those are available in [1].

In Absil *et al.* [2], we propose a method minimizing the energy of a bivariate manifold valued Bézier spline

$$f[\mathbf{b}^{mn}] = \textstyle\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} \hat{F}[\beta_3^{mn}].$$

In $\mathbb{R}^r$, the optimal control points can be expressed as affine combinations of the data points because the problem is invariant to translations. On manifolds, we express the problem on a product of (Euclidean) tangents spaces using a technique close to the one proposed in Arnould *et al.* [3].

4 Examples

On Figure 2, we show results on *(left)* the Euclidean space, *(middle)* the sphere and *(right)* the space of orthogonal orientations $SO(3)$ also named special orthogonal group. All results represent a Bézier surface computed by geodesic averaging of the control points. The control points are generated with the method proposed in Absil *et al.* [2]. On the special orthogonal group $SO(3$ for instance (Figure 2, right) we are able to smoothly interpolate different orientations of the truck (red) by a Bézier surface (blue). The continuity and derivability of the curve is easy to see on the left and middle figures.

5 Conclusion

We summarized different interpolative methods on manifolds. These methods are generalizations of piecewise-Bézier splines in the Euclidean space to general manifolds. We showed that data points are interpolated continuously and differentiably with a minimal knowledge of the geometry of the manifold. We also showed a way to choose the control points to obtain a good-looking (*i.e.* as flat as possible) spline.

References

[1] P.-A. Absil, R. Mahony and R. Sepulchre. "Optimization algorithms on Matrix Manifolds". Princeton University Press, Princeton, NJ, 2008.

[2] P.-A. Absil, P.-Y. Gousenbourger, P. Striewski, B. Wirth. "Differentiable piecewise-Bézier surfaces on Riemannian manifolds", Technical report UCL-INMA-2015.10-v1, Université catholique de Louvain, 2016. Accepted with minor changes in SIIMS (SIAM).

[3] A. Arnould, P.-Y. Gousenbourger, C. Samir, P.-A. Absil, M. Canis. "Fitting Smooth Paths on Riemannian Manifolds: Endometrial Surface Reconstruction and Preoperative MRI-Based Navigation", In F.Nielsen and F.Barbaresco, editors, GSI2015, Springer International Publishing, 491–498, 2015.

[4] R. Bergmann and A Weinmann. "A Second Order TV-type Approach for Inpainting and Denoising Higher Dimensional Combined Cyclic and Vector Space Data", arXiv 1501.02684v1, 2015.

[5] N. Boumal, B. Mishra, P.-A. Absil and R Sepulchre. "Manopt, a Matlab toolbox for optimization on manifolds", Journal of Machine Learning Research, 15, 1455–1459, 2014.

[6] G.E. Farin, "Curves and Surfaces for CAGD", Morgan Kaufmann editor, Academic Press, fifth edition, 2002.

[7] P.-Y. Gousenbourger, C. Samir and P.-A. Absil. "Piecewise-Bézier $\mathcal{C}^1$ interpolation on Riemannian manifolds with application to 2D shape morphing", International Conference on Pattern Recognition (ICPR), 4086–4091, 2014.

[8] X. Pennec, P. Fillard and N Ayache. "A Riemannian framework for tensor computing", International Journal of Computer Vision, **66**(1):41–66, 2006.

[9] G. Peyré. "Manifold models for signals and images", Computer Vision and Image Understanding, **113**(2):249–260, 2009.

[10] T. Popiel and L. Noakes, "Bézier curves and C^2 interpolation in Riemannian manifolds", "J. Approx. Theory, **148**(2):111–127, 2007.

[11] J. Solomon, F. De Goes, G. Peyré, *et al.* "Convolutional wasserstein distances", ACM Transactions on Graphics, Association for Computing Machinery, **34**(4), 66:1–66:11, 2015.