Bézier Curves and C^2 Interpolation in Riemannian Symmetric Spaces

Chafik Samir[1,3(✉)] and Ines Adouani[2]

[1] CNRS LIMOS (UMR 6158),
University of Clermont Auvergne, Clermont-Ferrand, France
chafik.samir@uca.fr

[2] Higher Institute of Applied Sciences and Technology of Sousse,
University of Sousse, Sousse, Tunisia

[3] Institut de Mathématiques de Toulouse, Université Paul Sabatier, Toulouse, France

Abstract. We consider the problem of interpolating a finite set of observations at given time instant. In this paper, we introduce a new method to compute the optimal intermediate control points that define a C^2 interpolating Bézier curve. We prove this concept for interpolating data points belonging to a Riemannian symmetric spaces. The main property of the proposed method is that the control points minimize the mean square acceleration. Moreover, potential applications of fitting smooth paths on Riemannian manifold include applications in robotics, animations, graphics, and medical studies.

Keywords: Riemannian Bézier curves · Regression on Riemannian manifolds · Curve fitting · Mean square acceleration · Special orthogonal group

1 Introduction

The problem of constructing smooth interpolating curves in non-linear spaces, or manifolds plays an important role in a wide variety of applications. For instance, interpolation in the rotation group $SO(3)$ has immediate application not only in computer graphics and animation of 3D objects [1–3], but also in applications ranging from robot motion planning to machine vision [4–6]. Such applications encourage us to further search for some efficient methods to generate smooth interpolating curves on non-linear spaces.

Motivated by potential applications in engineering science and technology, our goal is to develop a new framework for generating C^2 Bézier curves on Riemannian manifolds that interpolate a given ordered set of points at specified time instants. While quite general, we will focus on a special class of Riemannian symmetric spaces. The task of constructing interpolating curve on $SO(n)$ has attracted the attention of several authors. One of the most widely cited approaches is the work of Shoemake [7] on $SO(3)$, who adopts a reparametrization of the rotation matrices based on unit quaternion representation. Shoemake's approach can essentially be viewed as a generalization of the de Casteljau's algorithm for Bézier curves to $SU(2)$ in which two elements

F. Nielsen and F. Barbaresco (Eds.): GSI 2019, LNCS 11712, pp. 589–598, 2019.
https://doi.org/10.1007/978-3-030-26980-7_61

of $SO(3)$ are interpolated by the geodesic that joins them. Although this algorithm seems computationally efficient, unfortunately the resulting curve depends on the choice of local system coordinates. A few years later, taking into account the Shoemake algorithm, a more careful geometric analysis of unit-quaternion-based method was introduced by Barr et al. [1], Hart et al. [8], Ge and Ravani [9], and Nielson et al. [10]. Despite the fact of producing an intrinsic curves, these approaches does not generalize to higher-dimensional manifold.

In this paper, we present a novel framework to treat the interpolation problem in the setting of Riemannian geometry and Bézier curve approach. We show that it makes sense to define a C^2 interpolating Bézier curve on Riemannian symmetric spaces as the result of a least squares minimization and a recursive algorithm. In particular, we will focus on a special class of Riemannian symmetric spaces: the special orthogonal group $SO(n)$. Indeed, working in such Riemannian manifold allows nice properties to solve the issues above. The key point to give explicit solution for the interpolation problem and ensures the C^2 differentiability condition at joint points is the use of global symmetries in these last points. In fact, we will first derive equations for control points of a C^2 Bézier curve on the Euclidean space $\mathbb{R}^m$. Then, building upon prior works [6,11], we use these equations to find the control points of a C^1 interpolating Bézier curve on Riemannian manifolds as a generalization of the Bézier based fitting in the Euclidean space and by means of methods of Riemannian geometry. These results are sufficient to give explicit formula for control points of the C^2 interpolating Bézier curve on $SO(n)$. The proposed method will be shown to enjoy a number of nice properties and the solution is unique in many common situations.

The rest of the paper is organized as follows. In Sect. 2, we present our new algorithm to construct a C^2 Bézier curve on the Euclidean space. This will help with the visualization of its main features and motivate its generalization on $SO(n)$. In Sect. 3, the generalization of our approach on the Lie group $SO(n)$ is prescribed. We conclude the paper with numerical examples and a conclusion.

2 C^2 Interpolating Bézier Curves on $\mathbb{R}^m$

In this section, we first describe our approach on the Euclidean space $\mathbb{R}^m$. For simplicity we will assume that the time instants are $t_i = i$. In this work, we only use Bézier curves of degree 2 and 3 such that the segment joining p_0 and p_1, as well as the segment joining p_{N-1} and p_N are Bézier curves of order two, while all the other segments are Bézier curves of order three. Explicitly, the Bézier curve β_k of degree $k \in \{2, 3\}$ are expressed in $\mathbb{R}^m$ with a number of control points b_i, represented as their coefficients in the Bernstein basis polynomials by:

$$\beta_2(t; b_0, b_1, b_2) = b_0(1-t)^2 + 2b_1(1-t)t + b_2t^2,$$
$$\beta_3(t; b_0, b_1, b_2, b_3) = b_0(1-t)^3 + 3b_1t(1-t)^2 + 3b_2t^2(1-t) + b_3t^3.$$

Moreover, we assume that there exists two artificial control points $(\widehat{b_i^-}, \widehat{b_i^+})$ on the left and on the right hand side of the interpolation point p_i for $i = 1, ..., (N-1)$. Consequently, the Bézier curve β on $\mathbb{R}^m$ is given by:

$$\beta(t) = \begin{cases} \beta_2(t; p_0, \widehat{b}_1^-, p_1), & \text{if } t \in [0,1] \\ \beta_3(t-(i-1); p_{i-1}, \widehat{b}_{i-1}^+, \widehat{b}_i^-, p_i), & \text{if } t \in [i-1, i], i = 2, ..., N-1 \\ \beta_2(t-(N-1); p_{N-1}, \widehat{b}_{N-1}^+, p_N), & \text{if } t \in [N-1, N] \end{cases}$$

Then β is C^∞ on $[t_i, t_{i+1}]$, for $i = 0, ..N-1$. To ensure that β is C^1 at knots p_i, for $i = 1, ..N-1$, we shall make the following assumption:

$$\dot{\beta}_{k^i}(b_0^i, ..., b_{k^i}^i; t-i+1)|_{t=i} = \dot{\beta}_{k^{i+1}}(b_0^{i+1}, ..., b_{k^{i+1}}^{i+1}; t-i)|_{t=i} \qquad i = 0, ..., N-2. \tag{1}$$

This differentiability condition allows us to express $\widehat{b}_i^+$ in terms of $\widehat{b}_i^-$ as:

$$\widehat{b}_1^+ = \frac{5}{3}p_1 - \frac{2}{3}\widehat{b}_1^-, \tag{2}$$

$$\widehat{b}_i^+ = 2p_i - \widehat{b}_i^-, i = 2, ..., N-2 \tag{3}$$

$$\widehat{b}_{N-1}^+ = \frac{5}{2}p_{N-1} - \frac{3}{2}\widehat{b}_{N-1}^-, \tag{4}$$

We are left with the task of computing the control points $\widehat{b}_i^-$, for $i = 1, ..., N-1$, that generate the C^1 Bézier curve β. In [11], we have shown that solutions of the problem of minimization of the mean square acceleration of the Bézier curve β are exactly the control points of the curve:

$$\min_{\widehat{b}_1^-, ..., \widehat{b}_{N-1}^-} E(\widehat{b}_1^-, ..., \widehat{b}_{N-1}^-) := \min_{\widehat{b}_1^-, ..., \widehat{b}_{N-1}^-} \int_0^1 \|\ddot{\beta}_2^0(t; p_0, \widehat{b}_1^-, p_1)\|^2$$
$$+ \sum_{i=1}^{N-2} \int_0^1 \|\ddot{\beta}_3^i(t; p_i, \widehat{b}_i^-, \widehat{b}_{i+1}^-, p_{i+1})\|^2 + \int_0^1 \|\ddot{\beta}_2^{N-1}(t; p_{N-1}, \widehat{b}_{N-1}^-, p_N)\|^2 \tag{5}$$

It turns out that the optimal solution $Y = [\widehat{b}_1^-, ..., \widehat{b}_{N-1}^-]^T \in \mathbb{R}^{(N-1)\times m}$ of (5) is the unique solution of a tridiagonal linear system

$$Y = A^{-1}CP = DP \text{ with } \sum_{j=0}^{j=N+1} d_{ij} = 1. \tag{6}$$

where A is a tridiagonal sparse square matrix of size $(N-1) \times (N-1)$ with a dominant diagonal, C a matrix of size $(N-1) \times (N+1)$ and P the matrix of p_i's of size $(N+1) \times m$ given by:

$$A_{(1,1:2)} = [16\ 6] \tag{7}$$
$$A_{(2,1:3)} = [6\ 36\ 9] \tag{8}$$
$$A_{(i,i-1:i+1)} = [9\ 36\ 9], \tag{9}$$
$$A_{(n-1,n-2:n-1)} = [9\ 36] \tag{10}$$

$$C_{(1,1:2)} = [16\ 6] \tag{11}$$
$$C_{(2,2:3)} = [6\ 36\ 9] \tag{12}$$
$$C_{(i,i:i+1)} = [9\ 36\ 9],\ i = 3, ..., n-2 \tag{13}$$
$$C_{(n-1,n-1:n+1)} = [9\ 36] \tag{14}$$

Now, let us assume that β is C^1, so that (1) is met and the solution Y given by (6) is obtained. The additional C^2 condition for a C^1 curve is the equality of the second derivative at the joint point p_i, for $i = 1, ..., N-1$:

$$\ddot{\beta}_{k^i}(b_0^i, ..., b_{k^i}^i; t-i+1)|_{t=i} = \ddot{\beta}_{k^{i+1}}(b_0^{i+1}, ..., b_{k^{i+1}}^{i+1}; t-i)|_{t=i} \qquad i = 0, ..., N-2.$$

It is obvious that with this C^2 condition the position of the control points $\widehat{b}_i^-$ and $\widehat{b}_i^+$ that generate the curve β will be modified. Therefore, it is more convenient to use another notation. Let us denote by b_i^- and b_i^+ the new control points on the left and on the right hand side of the interpolation point p_i, for $i = 1, ..., N-1$. Computing the acceleration of β on respective intervals and taking into account that β is C^1, we shall replace b_1^+ by (2), b_i^+ by (3), and b_{N-1}^+ by (4). We deduce that:

$$b_2^- = \frac{1}{3}p_0 - \frac{1}{2}b_1^- + \frac{8}{3}p_1, \tag{15a}$$

$$b_{i+1}^- = b_{i-1}^+ + 4p_i - 4b_i^-, i = 2, ..., N-2 \tag{15b}$$

$$p_N = 2p_{N-1} + 2b_{N-1}^+ - 6b_{N-1}^- + 3b_{N-2}^+, \tag{15c}$$

We see at once that points that will be modified by the additional C^2 condition are $\widehat{b}_i^-$ and hence $\widehat{b}_i^+$, for $i = 2, ..., N-1$. The point $\widehat{b}_1^-$ remains invariant and consequently it will be the case for $\widehat{b}_1^+$. We thus get $b_1^- = \widehat{b}_1^-$, with $\widehat{b}_1^-$ is the first row of the matrix Y obtained as a solution of the optimization problem (5). However, the endpoint p_N is affected as we can deduce from Eq. (15c). Nevertheless, it follows that giving the control point b_1^- allows us to find all the other control points including b_2^- with Eq. (15a) and hence b_2^+ with (3), then b_{i+1}^- for $i = 2, ..., N-2$ with (15b) and therefore b_i^+, for $i = 3, ..., N-2$ with (3) and b_{N-1}^+ with (4).

3 C^2 Interpolating Bézier Curves On $SO(n)$

Our objective in this section is to work out concretely the extension of our approach used to find control points that define a C^2 Bézier curve in the Euclidean space to the Riemannian manifold $SO(n)$. In other words, given $R_0, ..., R_N$ a set of $(N+1)$ distinct points in $SO(n)$ and $0 = t_0 < t_1 < ... < t_N = N$ an increasing sequence of time instants, we present a conceptually simple framework to construct a C^2 Bézier curve $\gamma : [0, N] \rightarrow SO(n)$ such that $\gamma(t_k) = R_k,\ k = 0, ..., N$. For the most part of Riemannian manifolds, the generalization of our approach is not straightforward. For the case treated here, of the Lie group $SO(n)$, since it is a symmetric space and all the important geometric functions have nice, closed-form expressions, the problem of finding a C^2 Bézier curve that interpolates a given set of points in such space can be completely solved.

Let us start by briefly sketch the differential structure of $SO(n)$. We illustrate this with the geometric toolbox described in Table 1. For more details concerning the differential geometry of $SO(n)$, see [12,13].

Table 1. Geometric toolbox for the Riemannian manifold $SO(n)$

Set	$SO(n) = \{R \in \mathbb{R}^{n\times n} \mid R^T R = I_n \text{ and } \det(R) = 1\}$
Tangent spaces	$T_R SO(n) = \{H \in \mathbb{R}^{n\times n} \mid RH^T + HR^T = 0\}$
Inner product	$<H_1, H_2>_R = \mathrm{trace}(H_1^T H_2)$
Exponential	$\mathrm{Exp}_R(H) = \mathrm{Exp}_I(R^T H) = R\exp(R^T H)$
Logarithm	$\mathrm{Log}_{R_1}(R_2) = R_1 \log(R_1^T R_2)$

The shortest geodesic arc joining R_1 to R_2 in $SO(n)$ can be parameterized explicitly by:

$$\alpha(t, R_1, R_2) = R_1 \exp(t \log(R_1^T R_2)),\ t \in [0,1]. \tag{16}$$

and we write:

$$\dot{\alpha}(t, R_1, R_2) := \frac{\partial}{\partial u}|_{u=t}\ \alpha(t, R_1, R_2).$$

Furthermore, for each $R_1 \in SO(n)$, there exists a symmetry

$$\varphi_{R_1} : SO(n) \longrightarrow SO(n),\ R_2 \longrightarrow R_1 R_2^T R_1$$

that reverses geodesics through R_1. It is easy to check that φ_{R_1} is an isometry and thus $SO(n)$ turns into a Riemannian symmetric space. For $R_1, R_2 \in SO(n)$, let us denote by $(d\mathrm{Exp}_{R_1})_H$ the derivative of Exp_{R_1} at $H \in T_{R_1}SO(n)$ and by $(d\varphi_{R_1})_{R_2}$ the derivative of the geodesic symmetry φ_{R_1} at R_2. Then, the following result can be easily proved and will be very important for the derivation of the results presented along this section.

Lemma 1. *Let $R_1 \in SO(n)$.*

(i) $(d\varphi_{R_1})_{R_2}^{-1} = (d\varphi_{R_1})_{\varphi_{R_1}(R_2)}$, *for all* $R_2 \in SO(n)$
(ii) $(dExp_{R_1})_H^{-1} = -(dExp_{R_1})_{-H} \circ (d\varphi_{R_1})_{Exp_{R_1}(H)}$, *for all* $H \in T_{R_1}SO(n)$

Let us now denote by $\gamma_k(t, V_0, ..., V_k)$ the Bézier curve of order $k \in \{2,3\}$ on $SO(n)$ with a number of control points V_i for $i = 0, ..., k$. Furthermore, similar to the Euclidean case, we will suppose that there exists two artificial control points $(\widehat{Z}_i^-, \widehat{Z}_i^+)$ on the left and on the right hand side of the interpolation point R_i for $i = 1, ..., (N-1)$. Hence, the Bézier Curve $\gamma : [0, N] \longrightarrow SO(n)$ is defined by:

$$\gamma(t) = \begin{cases} \gamma_2(t; R_0, \widehat{Z}_1^-, R_1), & \text{if } t \in [0,1] \\ \gamma_3(t-(i-1); R_{i-1}, \widehat{Z}_{i-1}^+, \widehat{Z}_i^-, R_i), & \text{if } t \in [i-1, i], i = 2, ..., N-1 \\ \gamma_2(t-(N-1); R_{N-1}, \widehat{Z}_{N-1}^+, R_N), & \text{if } t \in [N-1, N] \end{cases}$$

In order to obtain equations that govern the control points of the C^2 Bézier curve on $SO(n)$, one should begin to compute $(\widehat{Z}_i^-, \widehat{Z}_i^+)$, for $i = 1, ..., N-1$, control

Algorithm 1. Construction of the C^1 interpolating Bézier curve on $SO(n)$.

Input: $N \geq 3$, $R = [R_0, ..., R_N]^T$ a matrix of size $n(N+1) \times n$ containing the $(N+1)$ interpolation points on $SO(n)$.
Output: $\widehat{Z}$ and $\tilde{R}$.

for $i = 1 : N-1$ **do**
  Calculate $Q = [Q_0^i, ..., Q_N^i]^T$ a matrix of size $n(N+1) \times n$ containing the $(N+1)$ interpolation points on $T_{R_i}SO(n)$:
  for $k = 0 : N$ **do**
    $Q_k^i = \mathrm{Log}_{R_i}(R_k) = R_i \log(R_i^T R_k)$
    Calculate $X_i = [(B_1^i)^-, ..., (B_{N-1}^i)^-]^T$ a matrix of size $n(N-1) \times n$ containing the $(N-1)$ control points of the C^2 Bézier curve β_i on $T_{R_i}SO(n)$, and $\tilde{Q} = [\tilde{Q}_0^i, ..., \tilde{Q}_N^i]^T$ a matrix of size $n(N+1) \times n$ containing the new interpolation points on $T_{R_i}SO(n)$ using the prescribed method on section 2.
    Calculate control point $\widehat{Z}_i^-$ with $\widehat{Z}_i^- = \mathrm{Exp}_{R_i}((B_i^i)^-)$
    Calculate the new interpolation points $\tilde{R}_k = \mathrm{Exp}_{R_i}(\tilde{Q}_k^i)$.
  end for
end for
return $\widehat{Z}$ and $\tilde{R}$,

points of the Bézier curve γ that ensure the C^1 differentiability condition of γ at knots R_i on $SO(n)$. To do this, our main idea is to treat the fitting problem on the tangent space $T_{R_i}SO(n)$ at a point $R_i \in SO(n)$ as for the Euclidean case. Consequently, for each $i = 1, ..., N-1$, we would like to transfer the data $R_0, ..., R_N$ in each tangent space $T_{R_i}SO(n)$ using Riemannian logarithmic map. The mapped data are then given by $Q = (Q_0^i, ..., Q_N^i)$ with $Q_k^i = \mathrm{Log}_{R_i}(R_k)$ for $k = 0, ..., N$. Applying our approach used to define a C^2 Bézier curve on the Euclidean space $\mathbb{R}^m$ in each tangent space $T_{R_i}SO(n)$, for $i = 1, ..., N-1$, provides a natural and intrinsic method to compute control points $(\widehat{Z}_i^-, \widehat{Z}_i^+)$ of the desired C^1 Bézier curve γ on $SO(n)$.

Theorem 1. *Let $R_0, ..., R_N$ be a finite sequence of distinct points in the special orthogonal group $SO(n)$ with $R_i^T R_k$, $i \neq k$, sufficiently close to I_n. For each $i = 1, ..., N-1$, $Q = (Q_0^i, ..., Q_N^i)$ are the corresponding mapped data in the tangent space $T_{R_i}SO(n)$ at R_i defined by $Q_k^i = Log_{R_i}(R_k)$ for $k = 0, ..., N$. Set $t_0 = 0 < ... < t_N = N$ a sequence of time instants. Then, there exists a unique matrix $X_i = [(B_1^1)^-, ..., (B_{N-1}^1)^-]^T \in \mathbb{R}^{n(N-1)\times n}$ containing the $(N-1)$ control points that generate the C^2 Bézier curve β_i, in each tangent space $T_{R_i}SO(n)$ and a matrix $\tilde{Q} = [\tilde{Q}_0^i, ..., \tilde{Q}_N^i]^T$ of size $n(N+1) \times n$ containing the new $(N+1)$ interpolation points in each tangent space $T_{R_i}SO(n)$.*

Proposition 1. *Under the same hypotheses of Theorem 1, there exists a unique matrix $Z = [\widehat{Z}_1^-, ..., \widehat{Z}_{N-1}^-]^T \in \mathbb{R}^{n(N-1)\times n}$, containing the $(N-1)$ control points that generate the Bézier curve γ interpolating the points R_i at t_i on $SO(n)$, for $i = 0, ..., N$. The rows of $\widehat{Z}$ are given by:*

$$\widehat{Z}_i^- = Exp_{R_i}(\tilde{x}_i), \; i = 1, ..., N-1. \tag{17}$$

where $\tilde{x}_i$, represent the row i of X_i in $T_{R_i}SO(n)$, for $i = 1, .., N-1$. Moreover, the new $(N+1)$ interpolation points in $SO(n)$ are given by:

$$\tilde{R}_k = Exp_{R_i}(\tilde{Q}_k^i),\ k = 0, ..., N;\ i = 1, ..., N-1. \tag{18}$$

Algorithm 1 provides a detailed exposition of the steps of the proof of Theorem 1 and Proposition 1.

Corollary 1. *The Bézier path $\gamma : [0, 1] \to SO(n)$ is C^1 on $SO(n)$.*

Proof. The following result may be proved in much the same way as Corollary 3.3. in [11].

We are now in a position to formulate the main theorem of this section, which contains the counterpart of the equations derived in the last section that generate control points of a C^2 Bézier curve on $\mathbb{R}^m$. Let us assume that γ is C^1, so that the solution $\widehat{Z}$ is obtained. Let us denote by Z_i^- and Z_i^+ the new control points on the left and on the right side of the interpolation point $\tilde{R}_i$ that generate the C^2 Bézier curve γ on $SO(n)$. The key point to find the control points Z_i^-, for $i = 1, ..., N-1$ is similar to the Euclidean case. That is, we might know Z_1^- (and therefore Z_1^+ by the C^1 differentiability condition on $SO(n)$) and wish to define iteratively Z_i^- for $i = 2, ..., N-1$ (and obviously Z_i^+ in much the same way as Z_1^+).

Algorithm 2. Construction of the C^2 interpolating Bézier curve on $SO(n)$.

Input: $N \geq 3$, $\tilde{R} = [\tilde{R}_0, ..., \tilde{R}_N]^T$ a matrix of size $n(N+1) \times n$ containing the $(N+1)$ interpolation points on $SO(n)$.

Output: Z.

Calculate $\widehat{Z} = [\widehat{Z}_1^-, ..., \widehat{Z}_{N-1}^-]^T$ using Algorithm 1.

Set $Z_1^- = \widehat{Z}_1^-$.

Calculate control point Z_1^+:

$Z_1^+ = \mathrm{Exp}_{\tilde{R}_1}(-\frac{2}{3}\mathrm{Exp}_{\tilde{R}_1}^{-1}(Z_1^-))$

Calculate control point Z_2^-:

$Z_2^- = \mathrm{Exp}_{Z_1^+}\left(\frac{1}{3}\left((d\varphi_{\tilde{R}_1})_{Z_1^-}\left(\dot{\alpha}(1, \tilde{R}_0, Z_1^-)\right) - 4\dot{\alpha}(0, Z_1^-, \tilde{R}_1)\right)\right)$

for $i = 2 : N-2$ **do** do

  $Z_i^+ = \mathrm{Exp}_{\tilde{R}_i}(-\mathrm{Exp}_{\tilde{R}_1}^{-1}(Z_i^-))$

  $Z_{i+1}^- = \mathrm{Exp}_{Z_i^+}\left(\left((d\varphi_{\tilde{R}_i})_{Z_i^-}\left(\dot{\alpha}(1, Z_{i-1}^+, Z_i^-)\right) - 2\dot{\alpha}(0, Z_i^-, \tilde{R}_i)\right)\right)$

end for

Calculate control point Z_{N-1}^+:

$Z_{N-1}^+ = \mathrm{Exp}_{\tilde{R}_{N-1}}(-\frac{2}{3}\mathrm{Exp}_{\tilde{R}_{N-1}}^{-1}(Z_{N-1}^-))$

return Z,

Theorem 2. *Let $\tilde{R}_0, ..., \tilde{R}_N$ be a set of distinct points in the special orthogonal group $SO(n)$ given by Eq. (18) and $\alpha(t)$ the shortest geodesic arc joining control points of the curve γ on $SO(n)$ given by Eq. (16). Let $X_1 = [(B_1^1)^-, ..., (B_{N-1}^1)^-]^T$ be the matrix of size $n(N-1) \times n$ containing the control points of the C^2 Bézier curve β_1 in $T_{R_1}SO(n)$. Then, there exists a unique matrix $Z = [Z_1^-, ..., Z_{N-1}^-]^T \in \mathbb{R}^{n(N-1)\times n}$, containing the $(N-1)$ control points that generate the C^2 Bézier curve γ interpolating the points $\tilde{R}_i$ at t_i on $SO(n)$, for $i = 0, ..., N$. The rows of Z are given by:*

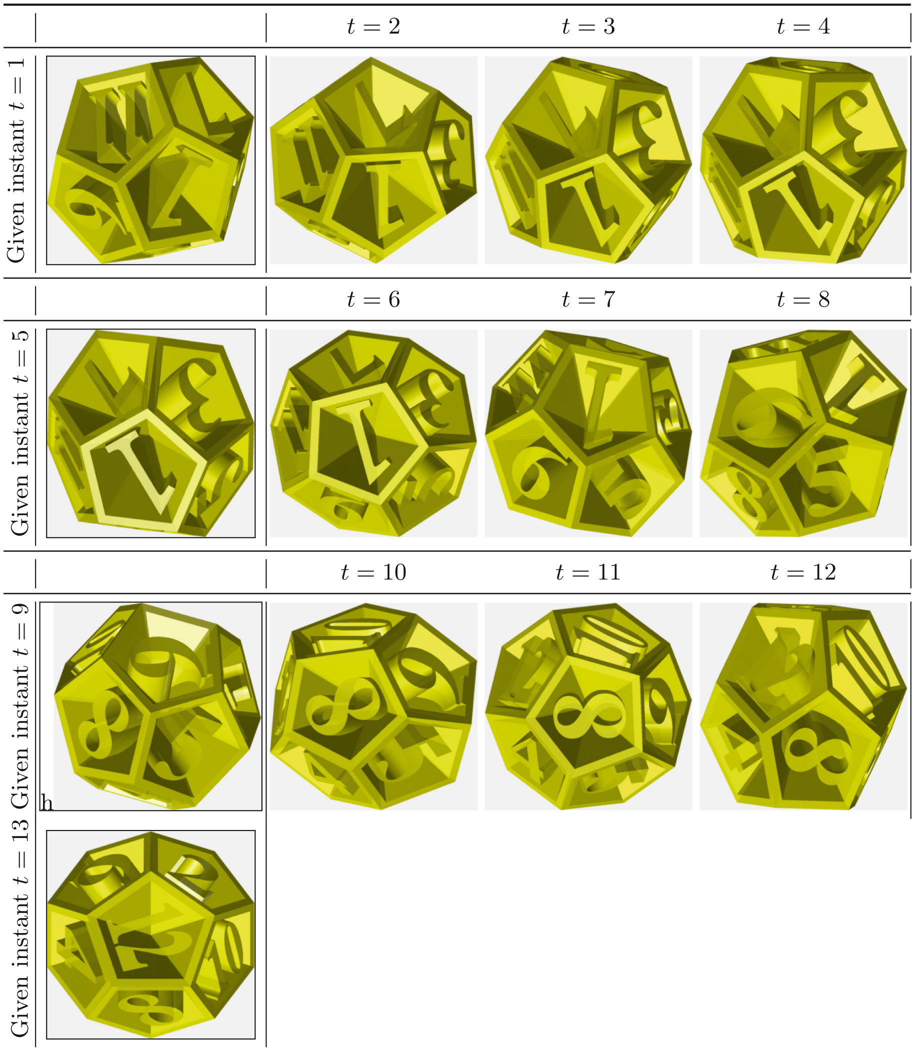

Fig. 1. Example of an interpolating path on $SO(3)$ applied to rotate a 12 sided dice at given time instants $(1, 5, 9, 13)$.

(i) $Z_1^- = Exp_{R_1}((B_1^1)^-)$.

(ii) $Z_2^- = Exp_{Z_1^+}\left(\frac{1}{3}\left((d\varphi_{\tilde{R}_1})_{Z_1^-}\left(\dot{\alpha}(1,\tilde{R}_0,Z_1^-)\right) - 4\dot{\alpha}(0,Z_1^-,\tilde{R}_1)\right)\right)$.

(iii) $Z_{i+1}^- = Exp_{Z_i^+}\left(\left((d\varphi_{\tilde{R}_i})_{Z_i^-}\left(\dot{\alpha}(1,Z_{i-1}^+,Z_i^-)\right) - 2\dot{\alpha}(0,Z_i^-,\tilde{R}_i)\right)\right)$, $i = 2, ..., N-2$.

We illustrate the proposed method to construct a smooth interpolating path on $SO(3)$ from four rotation matrices R_1, R_2, R_3, and R_4. We display the result in Fig. 1 where rotations are applied to rotate a 12 sided dice and the given time instants are displayed in a box. We can easily check that the resulting curve path is smooth including at the interpolation points.

4 Conclusion

In this paper, we have introduced a new framework and algorithms to study the fitting problem of C^2 Bézier curves to a finite set of time-indexed data points on the special orthogonal group $SO(n)$. The proposed method takes into account the global symmetries defined in the joint points. Therefore, the presented approach is valid on any locally symmetric space and other Riemannian symmetric spaces. In the future, we intend to extend the theory and then apply it to more general nonlinear manifolds.

Acknowledgement. This research was partially supported by The National Center for Scientific Research as CNRS PRIME Grant and the I-Site Clermont Auvergne project.

References

1. Barr, A., Currin, B., Gabriel, S., Hughes, J.: Smooth interpolation of orientations with angular velocity constraints using quaternions. ACM SIGGRAPH Comput. Graph. **26**, 313–320 (1992)
2. Popiel, T., Noakes, L.: Bézier curves and C2 interpolation in Riemannian manifolds. J. Approx. Theory **148**(2), 111–127 (2007)
3. Fang, Y., Hsieh, C., Kim, M., Chang, J., Woo, T.: Real time motion fairing with unit quaternions. Comput. Aided Des. **30**, 191–198 (1998)
4. Zefran, M., Kumar, V., Croke, C.: On the generation of smooth three-dimensional rigid body motions. IEEE Trans. Robot. Autom. **14**, 576–589 (1998)
5. Camarinha, M., Silva Leite, F., Crouch, P.: Splines of class C^k on non-Euclidean spaces. IMA J. Math. Control Inf. **12**(4), 399–410 (1995)
6. Arnould, A., Gousenbourger, P.-Y., Samir, C., Absil, P.-A., Canis, M.: Fitting smooth paths on Riemannian manifolds: endometrial surface reconstruction and preoperative MRI-based navigation. In: Nielsen, F., Barbaresco, F. (eds.) GSI 2015. LNCS, vol. 9389, pp. 491–498. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25040-3_53
7. Shoemake, K.: Animating rotation with quaternion curves. ACM SIGGRAPH **85**(19), 245–254 (1985)

8. Hart, J., Francis, G., Kaufman, L.: Visualizing quaternion rotation. ACM Trans. Graph. **13**(3), 256–276 (1994)
9. Ge, Q., Ravani, B.: Geometric construction of Bézier motions. ASME J. Mech. Des. **116**, 749–755 (1994)
10. Nielson, G., Heiland, R.: Animated rotationsusing quaternions and splines on a 4D sphere. Program. Comput. Softw. **18**(4), 145–154 (1992)
11. Samir, C., Adouani, I.: C^1 interpolating Bézier path on Riemannian manifolds, with applications to 3D shape space. Appl. Math. Comput. **348**, 371–384 (2019)
12. Helgason, S.: Differential Geometry, Lie Groups, and Symmetric Spaces (1978)
13. Adams, F.: Lectures on Lie Groups (1982)