



Interactive Generation of 3D Class A Bézier Curve Segments

Norimasa Yoshida¹, Ryo Fukuda² and Takafumi Saito³

¹Nihon University, norimasa@acm.org

²Nihon University

³Tokyo University of Agriculture and Technology, txsaito@cc.tuat.ac.jp

ABSTRACT

This paper presents a method for interactively generating a 3D class A Bézier curve segment by specifying two endpoints and their tangents. We clarify geometric properties of 3D class A Bézier curves and use them for efficiently generating 3D class A Bézier curve segments satisfying the specified positional and tangential constraints. The characteristics of typical 3D class A Bézier curves are also clarified.

Keywords: class A Bézier curves, curvature, torsion, interactive control.

DOI: 10.3722/cadaps.2010.163-172

1 INTRODUCTION

In the design of aesthetic surfaces, such as the exterior surfaces of automobiles, the use of aesthetic curves is very important. Such aesthetic curves should be curves whose curvature plots are composed of relatively few regions of monotonically varying curvature[3]. The monotonicity of the curvature is important since it dominates the distortion of the reflected image of curved surfaces generated by the curves.

For generating curves with monotonically varying curvature, there has been a lot of work of which most are related to planar curves. Farin and Sapidis used the curvature plot of a given curve so that the curve meets the aesthetic requirements [2]. Higashi et al. has proposed to use an evolute to control the curvature variation for generating aesthetic curves[6]. Wang et al. proposed to use the monotone curvature condition to design fair curves [17]. Farin proposed class A Bézier curves to generate 3D curves with monotone curvature and torsion [4], which was inspired by Mineur's typical curves[9]. In [4], however, interactive generation method for class A Bézier curves was not described. Yoshida et al. proposed a method for interactively controlling planar class A Bézier curves [13]. Log-aesthetic planar curves, which can be considered as the generalization of the Clothoid, Nielsen's spiral, logarithmic spiral, the circle involute, and circles have been proposed by Harada [5], Miura[10], and Yoshida and Saito [11]. Quasi-log-aesthetic planar curves [12] in rational cubic Bézier forms and the 3D extension of log-aesthetic space curves [14] have also been proposed. Walton and Meek proposed a G^2 curve design method using a pair of Pythagorean Hodograph (PH) quintic curves with monotone curvature [16]. In general, the simultaneous control of the monotonicity of the curvature (and torsion for space curves) and the endpoint constraints (two endpoints and their tangents) is not easy. The papers [11,12,13,14] have addressed the problem. In case of space curves, log-aesthetic space curve

segments[14] can be interactive controlled with user-specified endpoint constraints. However, the curve is not compatible with free-form curves, such as Bézier curves or NURBS curves.

In this paper, we extend our previous work of interactively controlling 2D class A Bézier curves[13] to 3D class A Bézier curves. In case of 2D class A Bézier curves, we used the geometric properties of the curves for interactive control. However, such geometric properties cannot be directly applicable to 3D class A Bézier curves. We clarify geometric properties of 3D class A Bézier curves and use them for efficiently generating the curves satisfying the specified positional and tangential constraints.

The rest of the paper is organized as follows. Section 2 reviews 3D class A Bézier curves. Two kinds of 3D class A Bézier curves, which are typical and general, are also introduced in Section 2. Section 3 presents a method for interactively drawing typical 3D class A Bézier curves by clarifying geometric properties of the curves. Section 4 describes a method for interactively drawing general 3D class A Bézier curves by using a projection whose normal is specified by a user. We present a novel method that reduces the generation of 3D class A Bézier curves to finding 2D class A Bézier curves and solving linear equations. Section 5 shows our results and clarifies several characteristics of typical 3D class A Bézier curves. Finally, we present conclusions and future work in Section 6.

2 3D CLASS A BÉZIER CURVES

Let $\mathbf{b}_0, \mathbf{b}_1, \dots, \mathbf{b}_n$ be the control point vectors in R^3 of a Bézier curve of degree n . Let $\Delta \mathbf{b}_i = \mathbf{b}_{i+1} - \mathbf{b}_i$. 3D class A Bézier curves are curves that have the following relationship and the curvature and torsion are monotonically varying:

$$\Delta \mathbf{b}_i = M^i \Delta \mathbf{b}_0 \quad (i = 1, \dots, n-1) \tag{2.1}$$

where M is a 3×3 matrix. Note that when we refer to class A, it means that the curvature and torsion are monotonically varying. 3D class A Bézier curves are originally proposed by Farin[4]. For a symmetric matrix M , Cao et al. have presented a method for checking the monotonicity of the curvature and torsion[1]. Matrix M , which we use in this paper, is not symmetric in general. To check the monotonicity of the curvature and torsion, we simply compute these values sampling points on the curve.

Fig. 1 shows an example of 3D class A Bézier curves with its curvature and torsion plots. In the figure, curves are drawn by quadratic prisms so that the user can sense the change of the torsion. Also, the shadow of the curve on the plane is also drawn so that the user can understand the shape of the 3D curve. To draw a 3D class A Bézier curve, we need to specify $\mathbf{b}_0, \mathbf{b}_1$, the degree n and M such that the generated curve has monotonically varying curvature and torsion. Finding M that generates a curve with monotonically varying curvature and torsion is not easy. Moreover, the endpoint of the curve is not known unless we draw the curve. In this paper, we will present a method for drawing a 3D class A Bézier curve by specifying endpoint conditions, which are two endpoints and their tangent vectors.

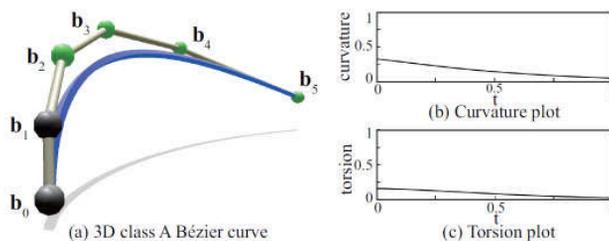


Fig. 1: 3D class A Bézier curve.

Farin called a class A Bézier curve generated by a matrix M_T , which is composed of a uniform scaling and a rotation around some axis, a *typical* class A Bézier curves. We call a class A Bézier

curves generated by a general matrix M_G , which is not just a composition of uniform scaling and a rotation, a *general* class A Bézier curves. We start from a generation of 3D typical class A Bézier curves and then present a method for generating 3D general class A Bézier curves.

3 INTERACTIVE GENERATION OF 3D TYPICAL CLASS A BÉZIER CURVES

We present a method for drawing a 3D typical class A Bézier curve generated by matrix M_T , which is composed of a uniform scaling and a rotation around some axis. Let $s, \theta, \mathbf{u} = [u_0 \ u_1 \ u_2]$ ($|\mathbf{u}|=1$) be the scaling factor, the angle of rotation, and the axis of rotation, respectively. Then M_T can be represented by

$$M_T = R \cdot S \quad (3.1)$$

where

$$S = \begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & s \end{bmatrix}, \quad (3.2)$$

$$R = I + (\sin \theta)A + (1 - \cos \theta)A^2. \quad (3.3)$$

Here, I is the identity matrix and

$$A = \begin{bmatrix} 0 & u_2 & -u_1 \\ -u_2 & 0 & u_0 \\ u_1 & -u_0 & 0 \end{bmatrix}. \quad (3.4)$$

To interactively generating a 3D typical class A Bézier curve, we specify two endpoints and their tangents by four points $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2$ and \mathbf{a}_3 . $\mathbf{a}_0, \mathbf{a}_3$ are two endpoints. $\mathbf{a}_1, \mathbf{a}_2$ are used to compute the tangents $\mathbf{v}_0, \mathbf{v}_1$ by $\mathbf{v}_0 = (\mathbf{a}_1 - \mathbf{a}_0)/|\mathbf{a}_1 - \mathbf{a}_0|$ and $\mathbf{v}_1 = (\mathbf{a}_3 - \mathbf{a}_2)/|\mathbf{a}_3 - \mathbf{a}_2|$. Therefore, changing the lengths $|\mathbf{a}_1 - \mathbf{a}_0|$ and $|\mathbf{a}_3 - \mathbf{a}_2|$ do not change the tangents $\mathbf{v}_0, \mathbf{v}_1$ unless they are 0. We use $\mathbf{a}_1, \mathbf{a}_2$ to specify tangential directions at two endpoints.

We are given the degree n and four points $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$, which specifies two endpoints $\mathbf{a}_0, \mathbf{a}_3$ and their tangents $\mathbf{v}_0, \mathbf{v}_1$. We would like to find a 3D typical class A Bézier curve satisfying the specified endpoint constraints. Thus, we need to find $\mathbf{b}_0, \mathbf{b}_1$ and M_T such that $\mathbf{a}_0 = \mathbf{b}_0$, $\mathbf{a}_3 = \mathbf{b}_n$, $\Delta \mathbf{b}_0 / |\Delta \mathbf{b}_0| = \mathbf{v}_0$ and $\Delta \mathbf{b}_{n-1} / |\Delta \mathbf{b}_{n-1}| = \mathbf{v}_1$.

For the position and the tangent to agree with the specified ones at \mathbf{a}_0 , we need to set $\mathbf{b}_0 = \mathbf{a}_0$ and $\mathbf{b}_1 = \mathbf{b}_0 + b_0 \mathbf{v}_0$, where $b_0 = |\Delta \mathbf{b}_0|$ is currently not known. However, changing b_0 means just scaling the curve, we simply set $b_0 = 1$ (b_0 can be an arbitrary value if $b_0 > 0$). Once the 3D typical class A Bézier curve is generated, b_0 can be easily computed by $|\mathbf{a}_3 - \mathbf{a}_0| / |\mathbf{b}_n - \mathbf{b}_0|$.

Now we need to find a matrix M_T such that $(\mathbf{b}_n - \mathbf{b}_0) / |\mathbf{b}_n - \mathbf{b}_0| = (\mathbf{a}_3 - \mathbf{a}_0) / |\mathbf{a}_3 - \mathbf{a}_0|$ and $\Delta \mathbf{b}_{n-1} / |\Delta \mathbf{b}_{n-1}| = \mathbf{v}_1$. The former condition corresponds to the positional coincidence at \mathbf{a}_3 under an appropriate scaling changing b_0 . From Eqn. (2.1) and (3.1) and the coincidence of tangential directions, the following equation must be satisfied:

$$\mathbf{v}_1 = R^{n-1} \mathbf{v}_0. \quad (3.5)$$

Eqn. (3.5) means that the axis of rotation \mathbf{u} of R is in the plane that is equiangular to both \mathbf{v}_0 and \mathbf{v}_1 . See Fig.2 (a) and (b). If we set $\hat{\mathbf{u}} = \mathbf{v}_1 \times \mathbf{v}_0$, the axis of rotation \mathbf{u} must be a vector that is a rotation of $\hat{\mathbf{u}}$ about the axis $\mathbf{v}_0 - \mathbf{v}_1$ by angle ϕ , which is not known yet. See Fig. 2(c).

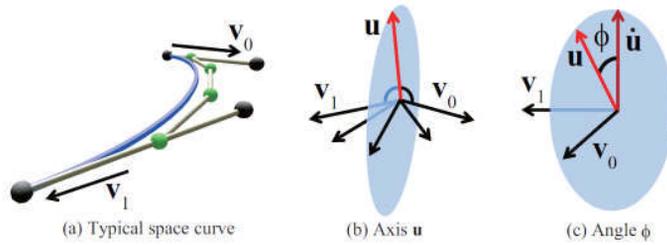


Fig. 2: The axis of rotation \mathbf{u} has one degree of freedom.

For an arbitrary ϕ , we can find the angle of rotation θ of R in the following manner. We project \mathbf{v}_0 and \mathbf{v}_1 to a plane whose normal is \mathbf{u} by a parallel projection, and call them $\hat{\mathbf{v}}_0$ and $\hat{\mathbf{v}}_1$, respectively. See Fig. 3. Let ψ be the angle between $\hat{\mathbf{v}}_0$ and $\hat{\mathbf{v}}_1$. By setting $\theta = \psi / (n - 1)$, Eqn. (3.5) is always satisfied without depending on ϕ . This means that the tangential coincidence at \mathbf{a}_3 is satisfied.

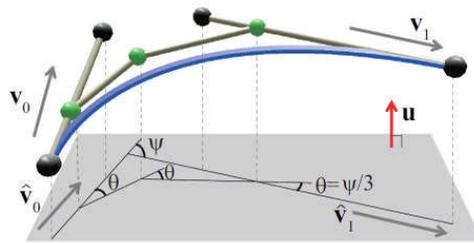


Fig. 3: Computing the angle θ in a projected plane with normal \mathbf{u} .

The remaining constraint that needs to be satisfied is the positional coincidence at \mathbf{a}_3 . We use an optimization such that the following function becomes 0.

$$f(\phi, s) = \frac{\sum_{i=0}^{n-1} b_i M_T^i \mathbf{v}_0}{\left| \sum_{i=0}^{n-1} b_i M_T^i \mathbf{v}_0 \right|} \cdot \frac{\mathbf{a}_3 - \mathbf{a}_0}{|\mathbf{a}_3 - \mathbf{a}_0|} - 1. \tag{3.6}$$

We use the downhill simplex method for optimization. Now, we can generate 3D typical class A Bézier curves that satisfy the endpoint constraints. Note that although the parameters ($\Delta \mathbf{b}_0$ and M_T) for generating a Bézier curve are always found, the curve may not be class A, which means the curvature and torsion are not monotonically varying. In such a situation, we simply do not draw the curve.

4 INTERACTIVE GENERATION OF 3D GENERAL CLASS A BÉZIER CURVES

Using the method of Section 3, we can generate 3D typical class A Bézier curves with specified endpoint conditions. By introducing 3D general class A Bézier curves, we can generate a variety of class A Bézier curves with the same endpoint conditions. In 3D general class A Bézier curves, the matrix M in Eqn. (2.1) is not just a composition of a uniform scaling and a rotation around some axis.

The idea of generating 3D general class A Bézier curves is first projecting the four points $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ to an arbitrary plane whose normal is specified by \mathbf{u} , computing a 2D class A Bézier curve in the plane as described in [13], and getting the information back to 3D. The 2D class A Bézier curve can be either typical or general. For drawing a general class A Bézier curve, the user needs to specify the direction of projection \mathbf{u} in addition to four points $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ and the degree n . The user may

also specify five other parameters that change the shape of the curve. These parameters will be described shortly in this section.

Without loss of generality, we assume that the projection plane is xy plane, that is $\mathbf{u} = [\mathbf{0} \ 0 \ 1]^T$. If $\mathbf{u} \neq [\mathbf{0} \ 0 \ 1]^T$, we initially rotate the four points $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2$ and \mathbf{a}_3 such that \mathbf{u} becomes $[\mathbf{0} \ 0 \ 1]^T$. Then we rotate the four points and the computed 3D Bézier control points back after the curve is generated. This assumption simplifies the computation of the matrix M_G , which generates a 3D general class A Bézier curve.

We first project the four points $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2$ and \mathbf{a}_3 to the xy plane (whose normal is \mathbf{u}) by a parallel projection and rename them $\hat{\mathbf{a}}_0, \hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2$ and $\hat{\mathbf{a}}_3$, respectively. Let $\hat{\mathbf{v}}_0 = (\hat{\mathbf{a}}_1 - \hat{\mathbf{a}}_0) / |\hat{\mathbf{a}}_1 - \hat{\mathbf{a}}_0|$ and $\hat{\mathbf{v}}_1 = (\hat{\mathbf{a}}_3 - \hat{\mathbf{a}}_2) / |\hat{\mathbf{a}}_3 - \hat{\mathbf{a}}_2|$. See Fig.4 (a). Note that either of $|\hat{\mathbf{a}}_1 - \hat{\mathbf{a}}_0|$ or $|\hat{\mathbf{a}}_3 - \hat{\mathbf{a}}_2|$ is 0, the curve cannot be generated.

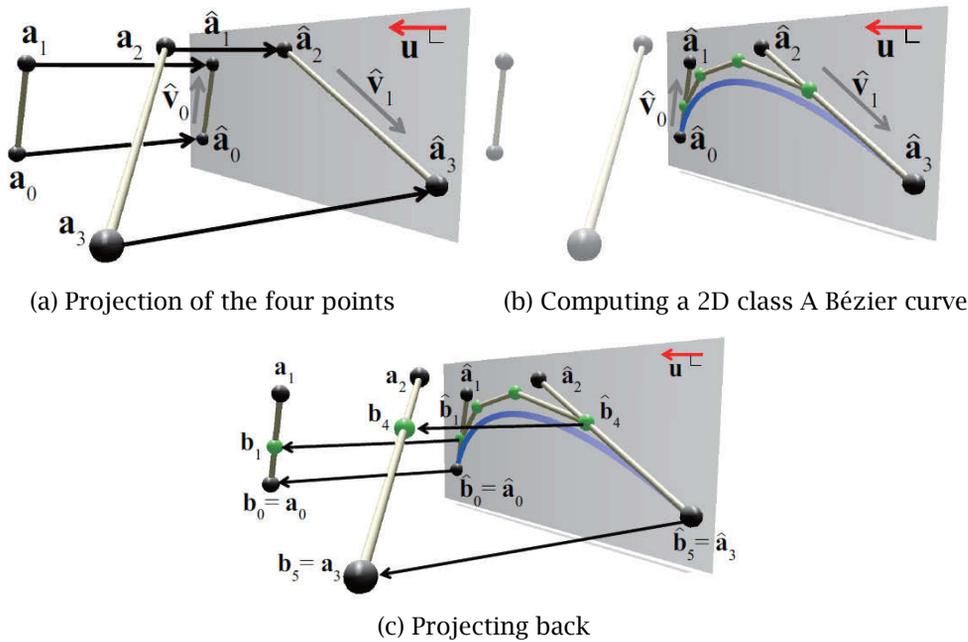


Fig. 4: Generating a general class A Bézier curves.

We then generate a 2D class A Bézier curve whose endpoints are $\hat{\mathbf{a}}_0$ and $\hat{\mathbf{a}}_3$ and their tangents are $\hat{\mathbf{v}}_0$ and $\hat{\mathbf{v}}_1$, respectively, using the method described in [13]. See Fig. 4 (b). The 2D class A Bézier curve can be either typical or general. When generating a 2D general class A Bézier curve, the user may need to specify additional four parameters that perturbs the 2×2 matrix [13]. Let $\hat{\mathbf{b}}_0, \hat{\mathbf{b}}_1, \dots, \hat{\mathbf{b}}_n$ be the computed control points of 2D class A Bézier curve.

Since the projection plane is xy plane, the x, y coordinates of the control points $\mathbf{b}_0, \mathbf{b}_1, \dots, \mathbf{b}_n$ of the 3D general class A Bézier curve are equal to the x, y coordinates of $\hat{\mathbf{b}}_0, \hat{\mathbf{b}}_1, \dots, \hat{\mathbf{b}}_n$, respectively. Thus, we need to find the z coordinates of $\mathbf{b}_0, \mathbf{b}_1, \dots, \mathbf{b}_n$. Since $\mathbf{b}_0 = \mathbf{a}_0$ and $\mathbf{b}_n = \mathbf{a}_3$, the z coordinates of \mathbf{b}_0 and \mathbf{b}_n are those of \mathbf{a}_0 and \mathbf{a}_3 , respectively. The z coordinate of \mathbf{b}_1 can be computed by projecting $\hat{\mathbf{b}}_1$ back toward the line formed by \mathbf{a}_0 and \mathbf{a}_{n-1} . The z coordinate of \mathbf{b}_{n-1} can also be computed in a similar manner. See Fig. 4 (c). The remaining coordinates of $\mathbf{b}_2, \dots, \mathbf{b}_{n-2}$ must be computed such that the z

coordinates of \mathbf{b}_{n-1} and \mathbf{b}_n (computed by backward projection) are the same as the ones computed using M_G .

Let $M_2 = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ be the 2×2 matrix that generates the 2D class A Bézier curve on the projection plane. By setting the upper left 2×2 of matrix M_G equal to M_2 and the (1,3) and (2,3) elements of the matrix be both 0, the x and y coordinates of the 2D class A Bézier curve agree with the x and y coordinates of the 3D class A Bézier curve. Thus M_G becomes

$$M_G = \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ \delta_0 & \delta_1 & e \end{bmatrix}. \quad (4.1)$$

Here, δ_0, δ_1 are currently unknown and e is a parameter which is specified by a user. We initially set $e = 0$.

Let the third row of $\prod_{i=0}^{j-1} M_G^i$ be $M_{G_{j,z}}$. Note that since the (1,3) and (2,3) elements of M_G are 0, $M_{G_{j,z}}$ is linear with respect to δ_0 and δ_1 . For the z coordinates ($b_{n-1,z}$ and $b_{n,z}$) of \mathbf{b}_{n-1} and \mathbf{b}_n computed by backward projection to be the same as the ones computed using M_G , the following equations must be satisfied.

$$b_{n-1,z} = \sum_{i=0}^{n-2} M_{G_{i,z}} \cdot \Delta \mathbf{b}_0 + b_{0,z}. \quad (4.2)$$

$$b_{n,z} = \sum_{i=0}^{n-1} M_{G_{i,z}} \cdot \Delta \mathbf{b}_0 + b_{0,z}. \quad (4.3)$$

Since Eqn. (4.2) and (4.3) are linear equations, we can compute δ_0 and δ_1 . Using M_G and Eqn. (2.1), we can generate all the control points that satisfies the endpoint constraints. Again, note that the generated curve may not be class A. In such a case, we just do not draw the curve segment.

5 RESULTS AND THE CHARACTERISTICS OF TYPICAL CLASS A BÉZIER CURVES

5.1 Generation Results of 3D Class A Bézier Curves

Fig. 5, 6 and 7 show examples of three kinds of class A Bézier curves of degree 5 with the same endpoint conditions. To help understand the 3D shape of the curve segments, the shadows of the curves are also drawn. Fig. 5 shows typical class A Bézier curves whereas Fig. 6 and 7 show general class A Bézier curves. The curves of Fig. 6 and 7 are different in that the parameters a, b, c, d and e in Eqn. (4.1) are different. To show the characteristics of these curves, we show the curvature and torsion plots as well as logarithmic curvature graphs (LCGs) and logarithmic torsion graphs (LTGs) [14,15]. LCGs and LTGs can be used to identify the shape information of the curve. In case of a 2D curve whose LCG is linear with slope α , it indicates that the curvature of the curve is (under an appropriate scaling) equal to that of log-aesthetic curves with its slope of the LCG α . Especially when α is -1, 0, 1, or 2, the curve is the Clothoid, Nielsen's spiral, logarithmic spiral, or the circle involute, respectively. The same argument is true for a 3D curve. If the LTG of a 3D curve is linear whose slope is β , it gives us the information of the torsion function similarly as in the curvature function. Note that the curves whose LCGs and LTGs are both linear are called log-aesthetic space curves [14]. We find that the shapes of LCGs and LTGs of typical 3D class A Bézier curves are rather restricted in comparison with general 3D class A Bézier curves. This restriction will be investigated more detail in the next section. As is shown in the LCGs and LTGs in Fig. 6 and 7, the general 3D class A Bézier curves have wider variety of shapes than typical 3D class A Bézier curves.

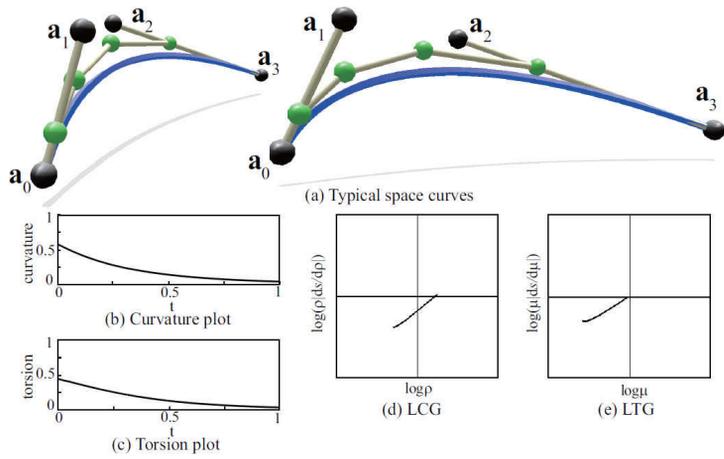


Fig. 5: An example of typical 3D class A Bézier curves.

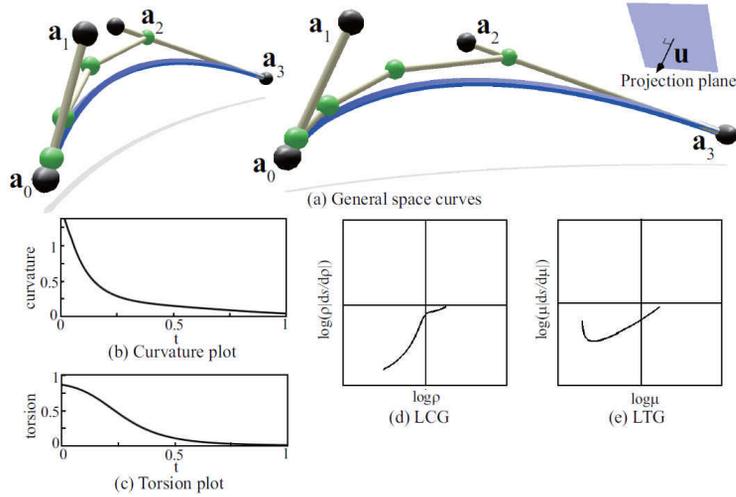


Fig. 6: An example of general 3D class A Bézier curves.

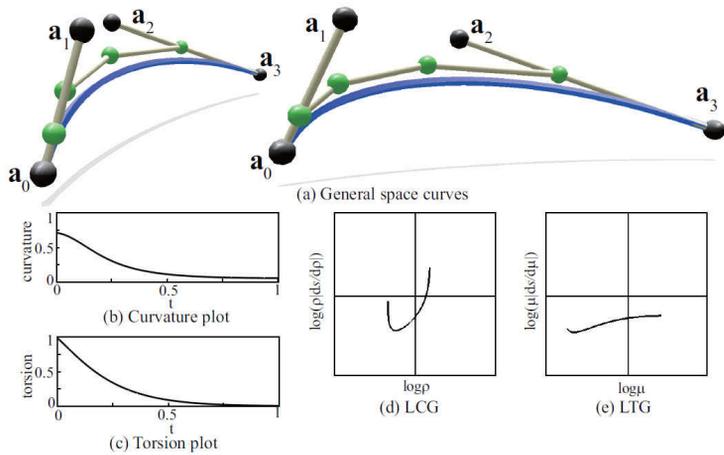


Fig. 7: Another example of general 3D class A Bézier curves.

5.2 Characteristics of Typical 3D class A Bézier Curves

To investigate the characteristics of typical 3D class A Bézier curves, we shows the LCGs and LTGs[13.14] of typical 3D class A Bézier curves with their slopes of approximated lines by least squares and their variances (var). See Fig. 8. Without depending on the two endpoints and their tangents, we found that the LCGs and LTGs get closer to a straight line whose slope is 1 as the degree gets higher. A curve whose slopes of the LCG and LTG are both 1 is the 3D extension of logarithmic spiral:

$$L(t) = (e^{at} \cos t, e^{at} \sin t, be^t).$$

Here a and b are constants. Although this does not prove that typical 3D class A Bézier curves are approximations to 3D logarithmic spirals, this fact shows that the representation space of typical 3D class A Bézier curves is rather limited.

deg.	Typical space curve	LCG	slope	LTG	slope
			var		var
3		$\log\left(\rho \left \frac{dr}{dr}\right \right)$	0.462	$\log\left(\mu \left \frac{dr}{dr}\right \right)$	0.140
					7.022
5		$\log\left(\rho \left \frac{dr}{dr}\right \right)$	0.760	$\log\left(\mu \left \frac{dr}{dr}\right \right)$	0.618
					0.275
10		$\log\left(\rho \left \frac{dr}{dr}\right \right)$	0.842	$\log\left(\mu \left \frac{dr}{dr}\right \right)$	0.747
					0.066
20		$\log\left(\rho \left \frac{dr}{dr}\right \right)$	0.923	$\log\left(\mu \left \frac{dr}{dr}\right \right)$	0.879
					0.006
30		$\log\left(\rho \left \frac{dr}{dr}\right \right)$	0.950	$\log\left(\mu \left \frac{dr}{dr}\right \right)$	0.920
					0.002
50		$\log\left(\rho \left \frac{dr}{dr}\right \right)$	0.969	$\log\left(\mu \left \frac{dr}{dr}\right \right)$	0.954
					7.0×10^{-4}
99		$\log\left(\rho \left \frac{dr}{dr}\right \right)$	0.990	$\log\left(\mu \left \frac{dr}{dr}\right \right)$	0.987
					6.3×10^{-4}

Fig. 8: LCGs and LTGs of typical 3D class A Bézier curves with various degrees.

The position of four points, thus two endpoints and their tangents, dictates whether the curve is class A or not. In case that the curve is not class A, we do not draw the curve. Fig. 9 shows the drawable regions of typical 3D class A Bézier curves. We set $\mathbf{a}_0 = [-1 \ 0 \ 0]$, $\mathbf{a}_1 = [-1 + \cos \eta \ \sin \eta \ 0]$, $\mathbf{a}_2 = [\cos \phi \cos \varphi \ \sin \phi \ \cos \phi \sin \varphi]$ and $\mathbf{a}_3 = [1 \ 0 \ 0]$. For $\eta = 20^\circ, 40^\circ, 60^\circ$ and 80° and degrees 3, 4, 5, 7 and 10, we changed ϕ and φ within $0 \leq \phi \leq 90$ and $-90 \leq \varphi \leq 90$, respectively. If a class A Bézier curve is generated, we draw the point \mathbf{a}_2 in either red or cyan. We draw the point in red if the curvature and torsion are both monotonically increasing from \mathbf{a}_0 to \mathbf{a}_3 . We draw the point in cyan if the curvature and torsion are both monotonically decreasing. Note that changing the length $|\mathbf{a}_1 - \mathbf{a}_0|$ or $|\mathbf{a}_2 - \mathbf{a}_3|$ does not affect the shape of the curve unless they are 0. Since polynomial curves cannot represent curves with constant curvature, the red and cyan regions in Fig. 9 are always disconnected. When η is small, the starting point (\mathbf{a}_0) tend to become the curvature minimum thus the red region is larger than the cyan region. When η becomes large, the starting point tend to become the curvature maximum and the cyan region gets larger. The drawable region gets larger as the degree gets higher.

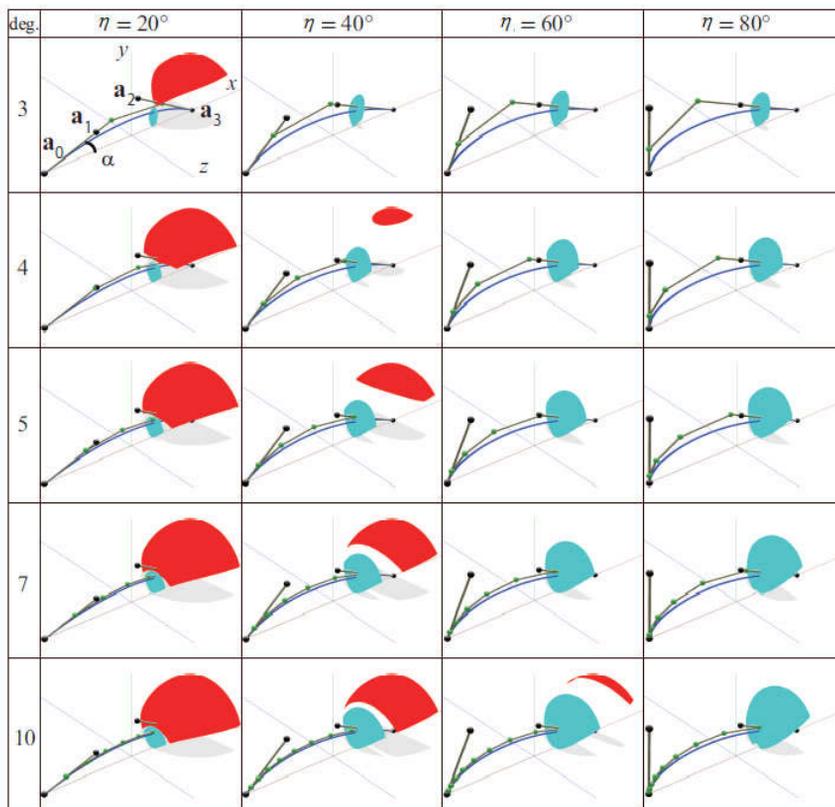


Fig. 9: Drawable regions of 3D typical class A Bézier curves.

6 CONCLUSIONS

This paper presented a method for interactively generating a 3D class A Bézier curve by specifying two endpoints and their tangents. For typical 3D class A Bézier curves, we clarified geometric properties of the curves and used them for efficient generation. For general class A Bézier curves, we proposed a

novel method that reduces the generation of general 3D class A Bézier curves to finding 2D class A Bézier curves in a plane whose normal is specified by a user and solving linear equations. Using LCGs and LTGs, we showed that the slopes of LCGs and LTGs of typical 3D class A Bézier curves gets closer to 1 as the degree gets higher. We also clarified experimental drawable regions of typical 3D class A Bézier curves.

There are several interesting directions for future research. For generating general class A Bézier curves, we need to specify 5 additional parameters in addition to the normal of the projection and endpoint constraints. Reducing the number of additional parameters and finding how the curve shape will change depending on the parameter are useful. For an arbitrary (non-symmetric) matrix M , we also need a more efficient and mathematically rigorous way for checking the monotonicity of the curvature and torsion. It is pointed out that in a space curve resulting from the bending of a physical wire, the curvature and torsion are oppositely varying[8,7], which means the curvature is low at the region of high torsion and vice versa. Log-aesthetic space curves[14] include curves whose curvature and torsion are varying oppositely. The generalization of class A Bézier curves to curves whose curvature and torsion are oppositely varying is also an interesting direction for future research.

REFERENCES

- [1] Cao, J.; Wang, G.: A note on class A Bézier curves, *Computer Aided Geometric Design*, 25(7), 2008, 523-528.
- [2] Farin, G.; Sapidis, N.: Curvature and the Fairness of Curves and Surfaces, *IEEE Computer Graphics & Applications*, 9(2), 1989, 52-57.
- [3] Farin, G.: *Curves and Surfaces for Computer Aided Geometric Design*, 5th ed. Morgan Kaufmann, 2001.
- [4] Farin, G.: Class A Bézier curves, *Computer Aided Geometric Design*, 23(7), 2006, 573-581.
- [5] Harada, T.; Yoshidamo, F.; Moriyama, M.: An aesthetic curve in the field of industrial design, *Proceedings of IEEE Symposium on Visual Languages*, IEE Computer Society Press, 38-47, 1999.
- [6] Higashi, M.; Tsutamori, H.; Hosaka, M.: Generation of smooth surfaces by controlling curvature variation, *Computer Graphics Forum*, 15(3), 1996, 187-196.
- [7] Levien, R, L.: *From Spiral to Spline: Optimal Techniques in Interactive Curve Design*, PhD thesis, University of California, Berkeley, California, 2009.
- [8] Mehlum, E.: Appell and the apple (nonlinear splines in space). In M. Dæhlen, T. Lyche, and L. L. Schumaker, editors, *Mathematical Methods for Curves and Surfaces: Ulvik, Norway*. Vanderbilt University Press, 1994.
- [9] Mineur, Y.; Lichah, T.; Castelain, H.; Giaume, H.: A shape controlled fitting method for bezier curves, *Computer Aided Geometric Design*, 15(9), 1998, 879-891.
- [10] Miura, K. T.: A generation equation of aesthetic curves and its self-affinity, *Computer Aided Design & Applications*, 3(1-4), 2006, 457-464.
- [11] Yoshida, N.; Saito, T.: Interactive Aesthetic Curve Segments, *The Visual Computer (Pacific Graphics)*, 22(9-11), 2006, 896-905.
- [12] Yoshida, N.; Saito, T.: Quasi-Aesthetic Curves in Rational Cubic Bézier Forms, *Computer-Aided Design & Applications*, 4(1-4), 2007, 477-486.
- [13] Yoshida, N.; Saito, T.: Interactive Control of Planar Class A Bézier Curves using Logarithmic Curvature Graphs, *Computer-Aided Design & Applications*, 5(1-4), 2008, 121-130.
- [14] Yoshida, N.; Fukuda, R.; Saito, T.: Log-Aesthetic Space Curve Segments, *SIAM/ACM Joint Conference on Geometric and Physical Modeling (GDSPM)*, 2009, 35-46.
- [15] Yoshida, N.; Fukuda, R.; Saito, T.: Logarithmic Curvature and Torsion Graphs, in *Mathematical Methods for Curves and Surfaces 2008* edited by Daehlen et al., LNCS 5862, Springer, 2010, 434-443.
- [16] Walton, D. J.; Meek, D. S.: G^2 curve design with a pair of Pythagorean Hodograph quintic spiral segments, *Computer Aided Geometric Design*, 24(5), 267-285, 2007.
- [17] Wang, Y.; Zhao, B.; Zhang, L.; Xu, J.; Wang, K.; Wang, S.: Designing fair curves using monotone curvature pieces, *Computer Aided Geometric Design*, 25(5), 2004, 515-527.