An equivalent parameter geometric shape representation using independent coordinates of cubic Bézier control points

Wang Zhenwei¹,∗, Zhang Ziyu¹, Nakajima Shuro² and Chen Hong¹

¹ School of Aeronautics and Astronautics, University of Electronic Science and Technology of China, Chengdu 611731, People’s Republic of China
² Faculty of Systems Engineering, Mechatronics, Wakayama University, Wakayama-city 640-8510, Japan

E-mail: wangjanwey@163.com

Received 7 July 2021, revised 6 April 2022
Accepted for publication 8 June 2022
Published 27 June 2022

Abstract
Bézier surface has been commonly applied to represent the complex geometric shape. Generally, all control points are dealt with by the same blending functions, regardless of the effect of independent coordinate. It causes to lack the modeling flexibility. Therefore, this paper proposes an equivalent parameter geometric shape representation method using the independent coordinates of control points. Since the coordinate components of control points are independent, the geometric modeling becomes more flexible. Firstly, a general Bézier curve is described in detail. Related expression is brought out in the form of independent coordinates by introducing two parameters. Then, their geometric meanings are analyzed in detail. Since both parameters are independent to parametric variables \( u \) and \( v \), Bézier curve possess the same interval in the discrete parametric space, namely equivalent parameter. Next, a bicubic Bézier subsurface patch representation is discussed, including regular and non-regular subsurface patch. A general surface expression is given out in the form of independent coordinates, as well as the parameter structure and the geometric transformations. Finally, an example of ‘Bézier tree branch’ is constructed by using...
the proposed method. Results shows that the proposed method is feasible and reasonable.

Keywords: geometric shape, independent coordinates, equivalent parameter, Bézier curve
Mathematics Subject Classification numbers: 37C20, 37E05.

(Some figures may appear in colour only in the online journal)

1. Introduction

As far as the geometric model is concerned, geometric shape representation could be one of main bottlenecks. Parameterized representation of geometric shape has brought more conveniences to complex geometric shape construction in industrial practices, such as microstructure, free surface, etc [1, 2]. With the rapid development of product design, geometric representation encounters many challenges, such as the conceptual design, the multidisciplinary design, the collaborative design [3–5]. It is well known that parameterized representation has brought more conveniences to represent complex geometric shape. More researches pointed out that geometric model is important for product design [6–9]. As the commonly-used parameterized representation, Bézier curve segment and surface patch have explicit geometric properties and well-behaved control within a convex hull for any degree [10–13]. Nowadays, many researches have been published about Bézier curve or surface representation, including arithmetic geometry, model representation, optimization algorithm, applied mathematics and approximation theories [14–18]. Moreover, more efforts have made on improvement on novel curves and their continuities, curvature properties, curve fitting, etc [19–23]. However, traditional geometric representation appears to lack geometric flexibility to some extent. Therefore, this paper proposes an equivalent parameter modeling using independent coordinate of cubic Bézier control points. This representation method greatly originates from two facts. The first is that traditional geometric representation mainly depends on control points, which completely determine the curve and surface shape. It can be called as ‘point-control representation’. Point coordinates are difficult to describe the geometric meanings of geometric elements, such as point selection, curve projection, etc. The solution is to embed control variables into the Bézier geometric shape, which is defined as ‘parameter-control representation’. The second fact is that control point coordinates are homogeneously conducted regardless of the coordinate components effect on geometric shape. In fact, control point coordinates are so independent that they are well-suited to represent complex geometric shape. Therefore, using independent coordinates is appropriate to the flexible modeling. Indeed, the coordinate independence will cause computation difficulty. However, the equivalent-parameter property can be evaluated easily because the geometric shapes have the same interval in discrete parameter space.

Above all, the proposed representation method can realize the flexibility and controllability of geometric shape representation. It can be characterized as the independent coordinates, the parameter-control representation and the equivalent parameter. The equivalent-parameter property is benefit to the complex geometric modeling. Especially, it can greatly increase the computation and representation precision for industrial practices, such as complex surface design, precision manufacturing and robot motion planning, etc. Generally, the more is the variable number of geometric models, the more comprehensive is the geometric shape representation. Meanwhile, more variables will improve the controllability of geometric modeling.
Therefore, this paper intends to represent geometric shape by using independent coordinates and multi-variables. Briefly, the proposed method carries out two operations on equivalent parameters, including the subset abstraction of cubic Bézier sub-curve and the geometric shape representation by using independent coordinates of control points. More details will be discussed in the following sections.

2. Construction of general Bézier curve with independent coordinate

Generally, a Bézier curve segment can be expressed recursively in the form of regular polynomials. For the conveniences of discussion, a cubic Bézier curve segment is described firstly. Thus, a general Bézier curve segment representation can be easily obtained based on a cubic Bézier curve segment. A cubic Bézier curve segment is represented by four control points, as shown in figure 1. Its expression equation can be written in the form of matrix, as shown equation (1).

\[
c(u) = b^3(t(u)) \cdot m \cdot p = b^3(u) \cdot p \quad u \in [0, 1].
\]  

Where, \( t(u) \) is the vector variable, \( t(u) = (1, u, u^2, u^3) \); \( m \) is the constant matrix, \( m = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix} \); \( p \) is the matrix variable, \( p = (P_0, P_1, P_2, P_3)^T \); \( b^3(u) \) is the vector variable, which contains four blending functions, \( b^3(u) = \left( b_0^3(u), b_1^3(u), b_2^3(u), b_3^3(u) \right) \), and \( b_i^j(u) \) is the blending function, \( b_i^j(u) = C_j^i u^i (1-u)^{j-i} \), and \( C_j^i \) is the binomial coefficient, \( C_j^i = \frac{j!}{i!(j-i)!} \).

From equation (1), two basic understandings about a cubic Bézier curve representation can be obtained. In the first place, variable \( b^3(t) \) is determined by the Bézier curve degree and the blending functions. Therefore, a cubic Bézier curve shape merely depends on the matrix variable \( p \). Theoretically, the main objective of geometric shape representation is to configure out four control points. Secondly, this expression equation neglects the control point coordinates, namely the coordinate components \((x, y, z)\). Therefore, the coordinate components of control point are conducted by the same blending function vector. In fact, the coordinate components are independent to the blending functions. It is reasonable that the coordinate components
are dealt with different blending function independently. Moreover, the independent coordinate components can increase modeling flexibility and generate complex geometric shape.

Above all, the matrix variable $\mathbf{p}$ will be decomposed into three components $\mathbf{p}_x$, $\mathbf{p}_y$, and $\mathbf{p}_z$ with corresponding to the coordinate components $x$, $y$, and $z$ respectively.

Generally, the parametric variable $u$ ranges from 0 to 1. It completely determines the spatial position of any point. To improve the modeling flexibility, two parametric variables $\lambda$ and $\gamma$ are introduced to abstract a subcurve segment from a given curve. As a subset of the given curve, this subcurve segment enables to figure out more local details about geometric shape. This paper defines such operation as the subset abstraction. Based on above two variables, the corresponding control points can be expressed as $\mathbf{c}(\lambda)$ and $\mathbf{c}(\gamma)$ respectively. Thus, the subcurve segment can be determined by four control points $\mathbf{c}(\lambda), P_{11}, P_{12}, \mathbf{c}(\gamma)$. More details about the subcurve segment will be discussed as the following.

The first subcurve segment is constructed by using the control points $\mathbf{c}(\lambda), P_7, P_9, P_3$. According to equation (1), this subcurve segment can be represented as equation (2).

$$
\mathbf{c}(u) = \mathbf{t}(u) \cdot \mathbf{m} \cdot \mathbf{p} \overset{u \rightarrow \lambda}{=} \begin{bmatrix} \mathbf{c}(\lambda) \\ P_7 \\ P_9 \\ P_3 \end{bmatrix} \mathbf{t}(u) = \mathbf{t}(u) \cdot \mathbf{m} \cdot \begin{bmatrix} \mathbf{c}(\lambda) \\ P_7 \\ P_9 \\ P_3 \end{bmatrix} \quad u \in [0, 1].
$$

Where, $\mathbf{p}_c$ is the matrix variable, $\mathbf{p}_c = [\mathbf{c}(\lambda), P_7, P_9, P_3]^T$. Since the variable $\mathbf{p}$ is known, the relationship between the variables $\mathbf{p}_c$ and $\mathbf{p}$ can be obtained easily, as shown equation (3).

$$
\begin{align*}
\begin{bmatrix} \mathbf{c}(\lambda) \\ P_7 \\ P_9 \\ P_3 \end{bmatrix} &= \begin{bmatrix} (1 - \lambda)^3 & 3\lambda(1 - \lambda)^2 & 3\lambda^2(1 - \lambda) & \lambda^3 \\ 0 & (1 - \lambda)^2 & 2(1 - \lambda)\lambda & \lambda^2 \\ 0 & 0 & 1 - \lambda & \lambda \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{p}_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix} \overset{u \rightarrow \lambda}{=} \begin{bmatrix} \mathbf{c}(\lambda) \\ P_7 \\ P_9 \\ P_3 \end{bmatrix}
\end{align*}
$$
Equation (1) shows that the transformation matrix $a$ from the variable $p$ to the variable $p_c$ is a triangle matrix. All blending functions are distributed in the form of upper triangle matrix. Similarly, the second subcurve segment can be constructed by using the control points $P_0$, $P_4$, $P_5$ and $c(\lambda)$. The transformation matrix $b$ from $p_c$ to $p$ is written as equation (4). Where, those control points are expressed as the matrix variable $p_c = (P_0, P_4, P_5, c(\lambda))^T$. From equation (4), the transformation matrix $b$ is a lower triangle matrix. Obviously, the matrix variable $b$ represents the first subcurve segment at the beginning of $u = 0$ and the matrix variable $a$ expresses the second subcurve segment at the end of $u = 1$. Therefore, combining matrix variables $a$ and $b$ can establish the complete subcurve expression. The following section will discuss the subcurve expression by using the control points $c(\lambda)$, $P_{11}$, $P_{12}$ and $c(\gamma)$.

\[
\begin{align*}
P_0 &= P_0 \\
P_4 &= (1 - \lambda)P_0 + \lambda P_1 \\
P_5 &= (1 - \lambda)P_4 + \lambda P_5 = (1 - \lambda)^2 P_0 + (1 - \lambda)\lambda P_1 + \lambda^2 P_6 \\
P_6 &= (1 - \lambda)P_5 + \lambda^2 P_2 = (1 - \lambda)^2 P_0 + 2(1 - \lambda)\lambda P_1 + \lambda^2 P_2 \\
c(\lambda) &= (1 - \lambda)^3 P_0 + 3\lambda(1 - \lambda)^2 P_1 + 3\lambda^2(1 - \lambda)P_2 + \lambda^3 P_3;
\end{align*}
\]

As far as the subcurve is concerned, main difficulty is to keep the equivalent-parameter property about both subcurve segments. Moreover, two variables $\lambda$ and $\gamma$ are not equal to zero. Therefore, the parametric variable $\varepsilon$ is defined as:

\[\varepsilon = \frac{\gamma - \lambda}{1 - \lambda}, \quad 0 \leq \lambda \leq \gamma \leq 1.\]

When the parametric variable $\lambda$ is given, $\gamma$ will vary from $\lambda$ and 1. As a result, the range of parametric variable $\varepsilon$ is from 0 to 1, namely $\varepsilon \in [0, 1]$. Thus, substitution the variable $\lambda$ in the variable $b$ with the variable $\varepsilon$ can guarantee equivalent-parameter property. In equation (4), the variable $p$ is replaced by the variable $p_c$ and the variable $p_c$ is replaced by the variable $p_{1c}$. As a result, the transformation from $p_{1c}$ to $p_c$ can be obtained, as shown equation (5).
Finally, a cubic Bézier subcurve representation can be obtained. Since control points can be scaled arbitrarily, the scale variable $k$ is given out. Namely, $k > 1$ refers to magnification, and $0 < k < 1$ means minification, and $k = -1$ is the symmetric transformation. Finally, a cubic Bézier subcurve is expressed as equation (6).

\[
\begin{align*}
\mathbf{c}(u) &= k \cdot \mathbf{b}^3(u) : \\
&= \begin{bmatrix}
\frac{\mathbf{b}_0(e)}{k} & 0 & 0 & 0 \\
\frac{\mathbf{b}_1(e)}{k} & \frac{\mathbf{b}_1(\lambda)}{k} & 0 & 0 \\
\frac{\mathbf{b}_2(e)}{k} & \frac{\mathbf{b}_2(\lambda)}{k} & 0 & 0 \\
\frac{\mathbf{b}_3(e)}{k} & \frac{\mathbf{b}_3(\lambda)}{k} & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\mathbf{P}_0 & \mathbf{P}_1 & \mathbf{P}_2 & \mathbf{P}_3
\end{bmatrix}
\end{align*}
\]

By combining equations (3) and (5), a subcurve representation can be obtained. Since control points can be scaled arbitrarily, the scale variable $k$ is given out. Namely, $k > 1$ refers to magnification, and $0 < k < 1$ means minification, and $k = -1$ is the symmetric transformation. Finally, a cubic Bézier subcurve is expressed as equation (6).

\[
\begin{align*}
\mathbf{c}(u) &= \mathbf{b}^3(u) : \\
&= \begin{bmatrix}
\frac{\mathbf{b}_0(e)}{k} & 0 & 0 & 0 \\
\frac{\mathbf{b}_1(e)}{k} & \frac{\mathbf{b}_1(\lambda)}{k} & 0 & 0 \\
\frac{\mathbf{b}_2(e)}{k} & \frac{\mathbf{b}_2(\lambda)}{k} & 0 & 0 \\
\frac{\mathbf{b}_3(e)}{k} & \frac{\mathbf{b}_3(\lambda)}{k} & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\mathbf{P}_0 & \mathbf{P}_1 & \mathbf{P}_2 & \mathbf{P}_3
\end{bmatrix}
\end{align*}
\]

According to the geometric algebra, the variable $k\mathbf{p}_c$ can be regarded as the projection of variable $\mathbf{p}_c$ on the projection center $P_c$. Therefore, the projection center variable $P_c$ is introduced into equation (6). Consequently, a complete cubic Bézier subcurve representation is established, as shown equation (7).

\[
\begin{align*}
\mathbf{c}(u) &= k \cdot \mathbf{b}^3(u) : \\
&= \begin{bmatrix}
\frac{\mathbf{b}_0(e)}{k} & 0 & 0 & 0 \\
\frac{\mathbf{b}_1(e)}{k} & \frac{\mathbf{b}_1(\lambda)}{k} & 0 & 0 \\
\frac{\mathbf{b}_2(e)}{k} & \frac{\mathbf{b}_2(\lambda)}{k} & 0 & 0 \\
\frac{\mathbf{b}_3(e)}{k} & \frac{\mathbf{b}_3(\lambda)}{k} & 0 & 0
\end{bmatrix}
\begin{bmatrix}
P_0 - P_c \\
P_1 - P_c \\
P_2 - P_c \\
P_3 - P_c
\end{bmatrix}
\end{align*}
\]

4090
Generally, above variables have different effects on subcurve shape. Figure 2 shows some subcurve shapes under conditions of different variables $k$ and $P_e$. Both subfigures are plotted under conditions of the same variables $\lambda = 0.2$ and $\varepsilon = 0.75$. All subcurves and given curve adopt the same parametric variable interval $\Delta u = 0.1$. It can be seen that the resulting subcurve segments are coincident with the given curve. Figure 2 proves that using the same control points can generate different cubic Bézier curve segments. Moreover, more local details can be obtained by adjusting the variable $k$. Above all, the variables $\lambda$, $\varepsilon$, $k$ and $P_e$ enable to realize the selection, projection and scaling operations on the given curve. The resulting subcurve shapes are not only controlled by control points, but also above four variables. Therefore, the subcurve representation becomes easier and more flexible.

Based on above discussions, a general representation for Bézier subcurve with $i$ degree are shown equation (8).

$$c(u) = k \cdot b'(u) \cdot \begin{bmatrix} b_0^i(\varepsilon) & 0 & \ldots & 0 & 0 \\ b_1^i(\varepsilon) & b_0^i(\varepsilon) & \ldots & 0 & 0 \\ b_2^i(\varepsilon) & b_1^i(\varepsilon) & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{i-1}^i(\varepsilon) & b_{i-2}^i(\varepsilon) & \ldots & 0 & 0 \\ b_i^i(\varepsilon) & b_{i-1}^i(\varepsilon) & \ldots & b_0^i(\varepsilon) & 0 \end{bmatrix} \begin{bmatrix} b_0^\lambda & b_1^\lambda & \ldots & b_{i-1}^\lambda & b_i^\lambda \\ 0 & b_0^{i-1}\lambda & \ldots & b_{i-2}^{i-1}\lambda & b_{i-1}^{i-1}\lambda \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & b_0^\lambda & b_1^\lambda \\ 0 & 0 & \ldots & 0 & b_0^\lambda \end{bmatrix} + P_e.$$  

Equation (8) is rewritten as equation (9) in the form of independent coordinate.
Where, the project center $P_e$ is combined into the matrix variable $p$, namely $Q = (p, P_e)^T$.

$$\mathbf{c}(t) = k \cdot \mathbf{x}(t, \varepsilon, \lambda) \cdot \mathbf{s}(\mathbf{P}, P_e) + P_e$$

$$\begin{vmatrix} c_x(t) \\ c_y(t) \\ c_z(t) \end{vmatrix} = \begin{vmatrix} \mathbf{x}(t, \varepsilon, \lambda) \cdot \mathbf{d}(k) + \mathbf{v} \end{vmatrix} \\ \begin{vmatrix} \mathbf{Q} = (Q_0, Q_1, Q_2) \end{vmatrix}$$

\[ \mathbf{e} = \left( e_1, e_2, e_3 \right) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \]

(9)

Figure 3 shows the independent coordinate effect on geometric shape. Under conditions of $P_e = (0, 0, 0)$, $\lambda = 0.2$ and $\varepsilon = 0.75(\gamma = 0.8)$, those subcurves describe the effect of variables $k = (k_x, k_y, k_z)$ on geometric shape. Where, all horizontal subcurves have the same variable $k_y = 1$ and all vertical subcurves have the same variable $k_z = 1$. Those geometric shapes show that the variable $k_x$ causes the subcurve segment to be scaled proportionally in the direction $x$. The geometric shape in the directions $y$ and $z$ are not changed. Thus, modifying the variable $k$ can generate different Bézier subcurve segment. Meanwhile, all resulting subcurve segments originate from the given curve. It means that complex geometric shape can be obtained by iteration calculation of variable $k$. Figure 4 illustrates two examples of Bézier curve sets. In the left subfigure, subcurves are plotted under conditions of two groups of the variable $k$. They show that the variables $k_x = -1$ and $k_y = 1$ realize the symmetric transformation about $y$ axis, and the variables $k_x = 1$ and $k_y = -1$ realize the symmetric transformation about $x$ axis. Moreover, all subcurve segments have the same angle range and two boundary lines have the equal interval. It could bring more convenience to fluid machine design and mechanic field analysis. In the right subfigure, the geometric shape looks like a ‘tree branch’. Where, the variable $k$ is set to the constants $(1, 1)$. The resulting subcurve segments show the effect of variables $u = (\lambda_x, \gamma_y)$ and $u = (\lambda_y, \gamma_y)$ on the geometric shape. Moreover, all branches are constructed by using the same cubic Bézier curve and the same control points.
3. Construction of Bézier surface patch with independent coordinate

Based on section 2, Bézier surface patch can be constructed by using independent coordinate. Generally, a surface patch needs 16 control points, as shown in figure 5(a). The control point matrix variable \( p \) can be written as \( (p_0, p_1, p_2, p_3)^T \) or \( (q_0, q_1, q_2, q_3)^T \). Elements \( p_i \) and \( q_j \) are the control points in the directions \( u \) and \( v \), namely, \( p_i = (P_{i0}, P_{i1}, P_{i2}, P_{i3})^T \) and \( q_j = (P_{0j}, P_{1j}, P_{2j}, P_{3j})^T \). Firstly, four Bézier curve segments \( c_0(u), c_1(u), c_2(u) \) and \( c_3(u) \) can be obtained along the direction \( u \), as shown equation (10). Similarly, other four Bézier curve segments \( c_0(v), c_1(v), c_2(v) \) and \( c_3(v) \) can be obtained in the direction \( v \), as shown equation (11). Meanwhile, the variables \( k \), \( \gamma \), and \( \lambda \) are decomposed in the directions \( x \), \( y \) and \( z \). For example, \( k_{1,x,u} \) refers to \( x \) component of the variable \( k \) in the direction \( u \), and \( c_{0,x}(u) \) is \( x \) component of the first curve \( c_{0,x}(u) \) in the direction \( u \). Combining equations (10) and (11) will generate a general bicubic Bézier surface patch \( s(u, v) \), as shown equation (12). It can be seen that all four curves in the directions \( u \) and \( v \) have the same parameter set \( \{ k_{1,x,u}, \gamma_{1,x,u}, \lambda_{1,x,u} \} \) or \( \{ k_{1,y,v}, \gamma_{1,y,v}, \lambda_{1,y,v} \} \). Therefore, this surface patch is homogeneous.

In equation (10), \( c_{0,x} \) composes of \( x \)-projection components of four control point sets \( p_i \) on blending function base vector \( \Gamma_{0,x} \), which is related to the first cubic curve parameter sets \( \{k_{1,x,u}, \gamma_{1,x,u}, \lambda_{1,x,u} \} \) in the direction \( u \). \( c_{i}(u) \) has related projection meanings. Moreover, four
diagonal elements in \( c_i(u) \) are equal to \( x \)-components of \( c_0(u) \), \( c_1(u) \), \( c_2(u) \) and \( c_3(u) \) correspondingly. Generally, there are two transformations, including the regular transformation and the non-regular transformation. The first transformation uses the same parametric variables for four curves in the directions \( u \) and \( v \), as shown equation (11). Since four control points can generate four curves. Therefore, there will be \( 4^4 = 256 \) subcurve segments. Consequently, total number of subsurface patches is \( 256^2 \cdot 256 = 65536 \). Figure 5(b) shows some regular bicubic subsurface patches. In fact, it is impossible to use all subsurface patches for the geometric shape representation.

\[
\begin{bmatrix}
    c_0(u) = \begin{pmatrix} c_0(x) = k_{0,0} \cdot x(u, c_{0,0}, \lambda_{0,0}) \cdot p_{0,T} \\ c_{0,y} = k_{0,0} \cdot x(u, c_{0,0}, \lambda_{0,0}) \cdot p_{0,T}^y \\ c_{2,x} = k_{2,0} \cdot x(u, c_{2,0}, \lambda_{2,0}) \cdot p_{2,T}^x \\ c_{2,y} = k_{2,0} \cdot x(u, c_{2,0}, \lambda_{2,0}) \cdot p_{2,T}^y \\ c_{3,x} = \varepsilon_k \cdot x(u, c_{3,0}, \lambda_{3,0}) \cdot p_{3,T}^x \\ c_{3,y} = \varepsilon_k \cdot x(u, c_{3,0}, \lambda_{3,0}) \cdot p_{3,T}^y \\ \end{pmatrix} \\
    c_1(u) = \begin{pmatrix} c_1(x) = k_{1,0} \cdot x(u, c_{1,0}, \lambda_{1,0}) \cdot p_{1,T} \\ c_{1,y} = k_{1,0} \cdot x(u, c_{1,0}, \lambda_{1,0}) \cdot p_{1,T}^y \\ c_{2,x} = k_{2,0} \cdot x(u, c_{2,0}, \lambda_{2,0}) \cdot p_{2,T}^x \\ c_{2,y} = k_{2,0} \cdot x(u, c_{2,0}, \lambda_{2,0}) \cdot p_{2,T}^y \\ c_{3,x} = \varepsilon_k \cdot x(u, c_{3,0}, \lambda_{3,0}) \cdot p_{3,T}^x \\ c_{3,y} = \varepsilon_k \cdot x(u, c_{3,0}, \lambda_{3,0}) \cdot p_{3,T}^y \\ \end{pmatrix} \\
    c_2(u) = \begin{pmatrix} c_2(x) = k_{2,0} \cdot x(u, c_{2,0}, \lambda_{2,0}) \cdot p_{2,T}^x \\ c_{2,y} = k_{2,0} \cdot x(u, c_{2,0}, \lambda_{2,0}) \cdot p_{2,T}^y \\ c_{3,x} = \varepsilon_k \cdot x(u, c_{3,0}, \lambda_{3,0}) \cdot p_{3,T}^x \\ c_{3,y} = \varepsilon_k \cdot x(u, c_{3,0}, \lambda_{3,0}) \cdot p_{3,T}^y \\ \end{pmatrix} \\
    c_3(u) = \begin{pmatrix} c_3(x) = \varepsilon_k \cdot x(u, c_{3,0}, \lambda_{3,0}) \cdot p_{3,T}^x \\ c_{3,y} = \varepsilon_k \cdot x(u, c_{3,0}, \lambda_{3,0}) \cdot p_{3,T}^y \\ \end{pmatrix} \\
    \end{bmatrix}
\]

(10)
\[
\begin{align*}
\mathbf{c}_0(v) &= \left( k_{0,xx} \mathbf{x}(v, \varepsilon_{0,xx}, \lambda_{0,xx}) \cdot \mathbf{q}_{0,xx} \right), \\
\mathbf{c}_1(v) &= \left( k_{1,xx} \mathbf{x}(v, \varepsilon_{1,xx}, \lambda_{1,xx}) \cdot \mathbf{q}_{1,xx} \right), \\
\mathbf{c}_2(v) &= \left( k_{2,xx} \mathbf{x}(v, \varepsilon_{2,xx}, \lambda_{2,xx}) \cdot \mathbf{q}_{2,xx} \right), \\
\mathbf{c}_3(v) &= \left( k_{3,xx} \mathbf{x}(v, \varepsilon_{3,xx}, \lambda_{3,xx}) \cdot \mathbf{q}_{3,xx} \right).
\end{align*}
\]

\[
\begin{align*}
\mathbf{c}_0(v) &= \left( k_{0,xx} 0 0 0 \right), \\
\mathbf{c}_1(v) &= \left( 0 k_{1,xx} 0 0 \right), \\
\mathbf{c}_2(v) &= \left( 0 0 k_{2,xx} 0 \right), \\
\mathbf{c}_3(v) &= \left( 0 0 0 k_{3,xx} \right).
\end{align*}
\]

\[
\begin{align*}
\mathbf{s}_1(u, v) &= \left( k_{0,uv} 0 0 0 \right), \\
\mathbf{s}_2(u, v) &= \left( 0 k_{1,uv} 0 0 \right), \\
\mathbf{s}_3(u, v) &= \left( 0 0 k_{2,uv} 0 \right), \\
\mathbf{s}_4(u, v) &= \left( 0 0 0 k_{3,uv} \right).
\end{align*}
\]

In Figure 5(b), variable \( u_i \) refers to parameters \( \lambda \) and \( \gamma \) in the \( i \)th sub-surface patch, namely \( u_i = (\lambda_{i,x}, \gamma_{i,x}; \lambda_{i,y}, \gamma_{i,y}) \). Where, \( \lambda_{i,x} \) and \( \gamma_{i,x} \) represent the \( x \)- and \( y \)-component of parameters \( \lambda \) and \( \gamma \) in the direction \( u \). For example, \( u_1 = (0.1, 0.3; 0.1, 0.3) \) means that all \( x \)- and \( y \)-components of parameters \( \lambda \) and \( \gamma \) are set to 0.1 and 0.3 in direction of \( u \) and \( v \). From the parameter list of surface patch, it can be seen that element function can be obtained with respect to the variable \( u_i \). Parameters \( \lambda_{i,y} \) and \( \gamma_{i,y} \) can realize the subsurface patch selection in the direction \( v \). Parameters \( \lambda_{i,x} \) and \( \gamma_{i,x} \) can realize the subsurface patch selection in the direction \( u \). Parameters \( \lambda_{i,x} \) and \( \gamma_{i,y} \) cause the subsurface tension along the direction \( u \). Parameters \( \lambda_{i,y} \) and \( \gamma_{i,x} \) has the tension effect along the direction \( v \).

In Figure 5(b), all regular subsurface patches have similar shapes. The reason is that all control curves use the same blending functions. Theoretically, four control points can respectively use different blending functions in the directions \( u \) and \( v \). It will generate non-regular cubic...
For non-regular subsurface patch, different blending functions are respectively applied to every four control points in the directions $u$ and $v$. The blending functions are written as the matrix variables $\Gamma_{u,i}$ and $\Gamma_{v,j}$. In other words, any control point $P_{ij}$ is blended by the $i$th element in the matrix $\Gamma_{u,i}$ and the $j$th element in the matrix $\Gamma_{v,j}$. Surface is the blending sum of all control points. Thus, a general non-regular Bézier surface patch is expressed as equation (13). Where, $\Gamma_{w,x}$ can be obtained by equations (10) and (11), and $w$ means any coordinate components $x$, $y$, and $z$.

$$s_w(u_w, v_w) = \sum_{i,j=1}^{n} (\Gamma_{w,u})_{ij} \cdot \{P_w\}_{ij} \cdot (\Gamma_{w,v})_{ij} w = x, y, z.$$  \hspace{1cm} (13)

Some non-regular bicubic subsurface patches are shown in figure 6. These surface patches show the effect of parameters $\lambda$ and $\gamma$ on geometric shape. Here, the parameter vector $\{\lambda_{x,u}, \gamma_{x,u}, \lambda_{y,u}, \gamma_{y,u}\}$ is used to express the two-dimensional shape in the direction $u$. The parameter vector $\{\lambda_{x,u}, \gamma_{x,u}, \lambda_{y,u}, \gamma_{y,u}, \lambda_{z,u}, \gamma_{z,u}\}$ is used to express the three-dimensional shape in the direction $u$. The parameter $k$ is set to a constant, $k = (1, 1, 1)$. From figure 6, it can be seen that all subsurface boundaries do not coincide with the original surface patch. Moreover, the parameter vector $\{\lambda_{x,u}, \gamma_{x,u}\}$ realizes the subsurface selection in the direction $u$ and the parameter vector $\{\lambda_{x,u}, \gamma_{x,u}, \lambda_{y,u}, \gamma_{y,u}, \lambda_{z,u}, \gamma_{z,u}\}$ abstracts subsurface in the direction $v$. Through setting different parameter vector, some complex bicubic subsurface patch can be obtained, as shown as figures 6(c) and (f). Indeed, other subsurfaces can be obtained by setting parameter $k$, which have been discussed in section 2.

As an example, the cubic Bézier subcurve segment is combined with the bicubic Bézier surface patch to construct a tree branch, as shown in figure 7. It includes three pieces of leaf, a control point matrix and a tree branch. Three pieces of leaf use the same control point matrix. Firstly, a basic shape based on the control point matrix is determined for construing three pieces of leaf. Then, applying the movement transformation to those basic shapes realizes leaf.
positioning along the tree branch. For leaf model, the related parameters are listed in those figures 7(d)–(f). For convenience of expression, the geometric transformation is only applied in the direction $v$. If more transformations are applied along the directions $u$ and $v$, the tree branch will become complex.

Above all, the proposed method enables to express the regular and non-regular Bézier curve and surface, and it possesses the flexibility and controllability of geometric modeling to some extent.

4. Conclusion

To improve the flexibility and controllability of geometric modeling, this paper proposes an equivalent parameter geometric shape representation method using the independent coordinates of control points. The proposed method applies different blending functions to related coordinate components of control points. Therefore, the geometric modeling becomes more flexible. Meanwhile, main parameters are defined and discussed in detail, such as the scale parameter vector, the selection parameter vector, etc. Those parameters can realize the scaling operation and the selection operations for the given curve. Moreover, different geometric models can be constructed by adjusting those parameters, but not by manipulating control points. Especially, the proposed method has obviously geometric significant, such as the center projection, the subset abstraction, etc. It will be benefit to geometric modeling for complex product design. Further researches will be to construct a closure model to realize more precision modeling and control. Above all, theoretical analysis and experiments show that the proposed method is reasonable and feasible for complex geometric shape.

Acknowledgments

This work was financially supported by The National Natural Science Foundation of China (No. 52175217) and Sichuan Science and Technology Program (No. 2021JDRC0013).
ORCID iDs

Wang Zhenwei  
https://orcid.org/0000-0003-1323-5303

References

[1] Shi H S, Wang Z W and Chen H 2021 Fringe center extraction for free surface measurement based on R channel IEEE 2nd Int. Conf. on Big Data, Artificial Intelligence and Internet of Things Engineering, ICBIAIE pp 88–92
[2] Yating Y, Yanbin L, Qin H and Chen X 2020 Microwave measurement and imaging for multiple corrosion cracks in planar metals Mater. Des. 196 109151
[3] Malak R J Jr, Aughenbaugh J M and Paredis C J J 2009 Multi-attribute utility analysis in set-based conceptual design Comput. Aided Des. 41 214–27
[4] Shai O, Reich Y and Rubin D 2007 Creative conceptual design: extending the scope by infused design Comput. Aided Des. 41 117–35
[5] Sriram R D, Syzkman S and Durham D 2006 Special issue on collaborative engineering J. Comput. Inf. Sci. Eng. 6 93–5
[6] Chandrasegaran S K, Ramani K, Sriram R D, Horváth I, Bernard A, Harik R F and Gao W 2013 The evolution, challenges, and future of knowledge representation in product design systems Comput. Aided Des. 45 204–28
[7] Antonelli M, Beccari C V, Casciola G, Ciarloni R and Morigi S 2013 Subdivision surfaces integrated in a CAD system Comput. Aided Des. 45 1294–305
[8] Cao H, Zheng H, Hu G and Abbas M 2021 Adjusting the energy of ball curves by modifying movable control balls Comput. Appl. Math. 40 76
[9] Jüttler B and Sampoli M L 2000 Hermite interpolation by piecewise polynomial surfaces with rational offsets Comput. Aided Geomet. Des. 17 361–85
[10] Bashir U, Abbas M and Ali J M 2013 The $G^2$ and $C^2$ rational quadratic trigonometric Bézier curve with two shape parameters with applications Appl. Math. Comput. 219 10183–97
[11] BiBi S, Abbas M, Misro M Y and Hu G 2019 A novel approach of hybrid trigonometric Bézier curve to the modeling of symmetric revolution curves and symmetric rotation surfaces IEEE Access 7 165779–92
[12] Arnal A and Monterde J 2011 A third order partial differential equation for isotropic boundary based triangular Bézier surface generation J. Comput. Appl. Math. 236 184–95
[13] Ahn Y J and Hoffmann C 2014 Circle approximation using LN Bézier curves of even degree and its application J. Math. Anal. Appl. 410 257–66
[14] Simeonov P, Zafiris V and Goldman R 2012 q-Blossoming. A new approach to algorithms and identities for q-Bernstein bases and q-Bézier curves J. Approx. Theory 164 77–104
[15] Maqsood S, Abbas M, Hu G, Ramli A L A and Miura K T 2020 A novel generalization of trigonometric Bézier curve and surface with shape parameters and its applications Math. Probl Eng. 2020 1–25
[16] Aphirukmatakun C and Dejduermrong N 2011 Multiple degree elevation and constrained multiple degree reduction for DP curves and surfaces Comput. Math. Appl. 61 2296–9
[17] Majeed A, Abbas M, Sittar A A, Misro M Y and Kamran M 2021 Airplane designing using quadratic trigonometric B-spline with shape parameters AIMS Math. 6 7669–83
[18] Chang J-W, Choi Y-K, Kim M-S and Wang W 2011 Computation of the minimum distance between two Bézier curves/surfaces Comput. Graph. 35 677–84
[19] Rababah A, Lee B-G and Yoo J 2006 A simple matrix form for degree reduction of Bézier curves using Chebyshev–Bernstein basis transformations Appl. Math. Comput. 181 310–8
[20] BiBi S, Abbas M, Miura K T and Misro M Y 2020 Geometric modeling of novel generalized hybrid trigonometric Bézier-like curve with shape parameters and its applications Mathematics 8 967
[21] Majeed A, Abbas M, Miura K T, Kamran M and Nazir T 2020 Surface modeling from 2D contours with an application to craniofacial fracture construction Mathematics 8 1246
[22] Usman M, Abbas M and Miura K T 2020 Some engineering applications of new trigonometric cubic Bezier-like curves to free-form complex curve modeling J. Adv. Mech. Des. Syst. Manuf. 14 JAMDSM0048

[23] Yu Y T, Yan Y, Wang F, Tian G Y and Zhang D J 2014 An approach to reduce lift-off noise in pulsed eddy current nondestructive technology NDT E Int. 63 1–6