Building a space for an analytic function that describes a spline curve

A V Tolok¹*, M A Loktev¹ and A A Sycheva¹

Laboratory of Computer Graphics, V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, 65 Profsoyuznaya street, Moscow, 117997, Russia

*tolok_61@mail.ru

Abstract. The article discusses some methods for obtaining the analytical expression of a Bezier curve and approaches to constructing analytically defined spline curves. A solution to the problem of constructing a functional region of a Bezier curve as its locally analytical description is proposed. The proposed solution is implemented through R-functional modeling (Rvachev function) and functional-voxel approach.

1. Introduction

Modern approaches to the application of high-precision analytical modeling in the process of design automation create new conditions for the computer representation of geometric models. In particular, the works [1, 2] consider modeling problems based on Rvachev functions (hereinafter R-functional modeling or R-functions) in problems of constructing objects of complex geometry. The theory of R-functions is an effective set-theoretic approach to modeling the space of the function of the form \( \Omega = f(X_n) \). However, there is a problem, since the main approaches of CAD systems are based on parametric description of functions for constructing curves and surfaces.

This approach has proven itself to be most effectively applied to computer representation. However, it cannot be used in analytical modeling without reduction to a nonparametric representation. The principles proposed in this paper allow us to solve the problem using functional voxel modeling [3] as one of the methods for computer representation of the space of a nonparametric function.

2. Previous Work

Using different spline curves has wide application in the solutions applied CAD problems. For example, work [4] uses spline approximation in CAD Systems of linear constructions. Other research papers are devoted to the use of B-splines, for example, for modeling road surfaces [5] or optimizing parametric geometry in CAD [6]. Especially worth noting are works devoted to the use of T-spline, which is one of the type of NURBS surfaces. For example, in [7], the authors presented their own isogeometric approach for optimizing forms based on T-splines. The most interesting works are devoted to Bezier segmentation [8], since the construction of Bezier curves is the main role in this article.
3. Methods

3.1. Analytical approaches to spline construction

Splines are smooth curves whose position is determined by a set of control and reference points. Splines are one of the main design tools and are used to build smooth curves and surfaces. Currently, there are many different types of spline curves and methods for constructing them. One of the types of such curves are Bezier curves, the construction of which is carried out on the basis of Bernstein polynomials [4]:

\[ b_{i,n}(t) = \binom{n}{i} t^i (1 - t)^{n-i}, \]

where \( \binom{n}{i} \) is the number of combinations of \( n \) in \( i \), \( n \) is the degree of the polynomial, \( i \) is the serial number of the support vertex. The expression of the ordinate and abscissas of the Bezier curve is a parametric equation - a linear combination of Bernstein polynomials:

\[
\begin{cases}
  x(t) = \sum_{i=0}^{n} x_i b_{i,n}(t), & 0 \leq t \leq 1 \\
  y(t) = \sum_{i=0}^{n} y_i b_{i,n}(t), & 0 \leq t \leq 1 
\end{cases}
\]

where \( x_i \) and \( y_i \) are the coordinates of the \( i \)-th reference vertex.

However, the parametric representation of the curve causes difficulties in applying it in analytical calculations - the \( x \) and \( y \) values of each point of the curve are determined by the parameter \( t \). There is a need to obtain an analytical expression of the curve in the form \( \Omega = f(x, y) \). It is possible to approach this issue in various ways. The first way is to try to express the parameter \( t \) from the expression via the \( x \) coord. For example, for a quadratic Bezier curve with reference points \( P_0(x_0, y_0), P_1(x_1, y_1), P_2(x_2, y_2) \):

\[
B(t) = (1 - t)^2 P_0 + 2t(1 - t)P_1 + t^2 P_2
\]

The parameter \( t \) at the current point \( B \) will be expressed as:

\[
t = \begin{cases}
  \frac{P_0 - P_1 \pm \sqrt{(P_0 - 2P_1 + P_2)(P_0 - P_1^2 - P_0P_2)}}{P_0 - 2P_1 + P_2}, & P_0 - 2P_1 + P_2 \neq 0 \\
  \frac{B - P_0}{2(P_1 - P_0)}, & P_0 - 2P_1 + P_2 = 0 \text{ и } P_1 \neq P_0 \\
  \frac{B - P_0}{P_2 - P_1}, & P_0 = P_1 \neq P_3
\end{cases}
\]

The high complexity of such an expression is obvious even in the case of a quadratic Bezier curve. With increasing degree of the Bezier curve, the expression of the parameter \( t \) will become more and more difficult to express and difficult to calculate. Another way to express the parameter \( t \) can be a local geometric approach. The local geometric characteristics \( n_1, n_2, n_3 \) for a plane curve are the slope angles between the normal to the tangent at the point of the curve and the coordinate axes \( OX, OY, OZ \). To obtain the local geometric characteristics of the curve, it is necessary to take two values of the parameter \( t_A \) and \( \Delta t_A \) at a point from 0 to 1. Substituting the parameter values into the expression of the Bezier curve, we obtain the coordinates \( X_A \) and \( \Delta X_A \). Having thus obtained two points \( A(t_A, X_A) \) and
\[ \Delta A(t_A + \Delta t_A, X_A + \Delta X_A), \] we obtain the expression of the straight line passing through them from the determinant of the matrix: expressed and difficult to calculate.

\[
\begin{vmatrix}
  t & x & 1 \\
  t_A & X_A & 1 \\
  t_A + \Delta t_A & X_A + \Delta X_A & 1
\end{vmatrix}
= -\Delta X_A t + \Delta t_A x + (t_A(X_A + \Delta X_A) - X_A(t_A + \Delta t_A))
\]

Thus, the coefficients of the line \( At + Bx + C = 0 \) will be expressed as:

\[
A = -\Delta X_A, \quad B = \Delta t_A, \quad C = t_A(X_A + \Delta X_A) - X_A(t_A + \Delta t_A)
\]

Local geometric characteristics are determined by dividing the coefficients of the line by the norm \( N \):

\[
N = \sqrt{A^2 + B^2 + C^2}, \quad n_1 = \frac{A}{N}, n_2 = \frac{B}{N}, n_3 = \frac{C}{N}
\]

These characteristics at each point in the region will describe the simulated curve; therefore, it is necessary to save them for further calculations. From the expression of this line in the context of local geometric characteristics, it is possible to express the parameter \( t \):

\[
n_1 t + n_2 x + n_3 = 0 \\
t = \frac{-n_2 x + n_3}{-n_1}
\]

Now, when the parameter \( t \) is expressed in terms of the \( x \) coordinate and the local geometric properties of the Bezier curve, it becomes possible to set the \( x \) coordinate, obtain the local geometric characteristics of the given point from the stored data, calculate the \( t \) parameter from them and determine the value of the \( y \) coordinate at this point:

\[ y(t) = (1 - t)^2 y_0 + (2 - t)t y_1 + t^2 y_2 \]

Figure 1 presents examples of constructing a Bezier curve in this way with reference points (red, connected by red lines).

As can be seen from the image, the algorithm does not work if the midpoint is located outside the boundaries of the first and last points. This problem is caused by the parametric nature of the Bezier curve. The analytic expression could not as accurately as parametrically describe its form. The solution to this problem may be to rotate the origin so that the second point is always between the first last. The implementation of this approach successfully works in the case of constructing a curve by 3 points, however, the construction of more complex curves will require the use of coordinate rotations and their complex relationship with each other.
One can try to construct a region of surface values \( z = ty - tx \), where the parameters \( tx \) and \( ty \) determine the values of the coordinates \( X \) and \( Y \), respectively. For each value of the parameter \( tx \), it is necessary to consider all the values of the parameter \( ty \) and determine by substituting the values of the \( X \) and \( Y \) coordinates in the expression of the Bezier curve. The sign of the difference \( ty \) and \( tx \) will allow us to construct a region of the Bezier curve value. The condition \( tx = ty \) describes the boundary of the positive and negative region, i.e. the curve itself. In case of a positive value of the difference, the point of the region \((X, Y)\) is shaded in gray, in the case of a negative - in blue (Fig. 2).

![Figure 2. The surface \( z = ty - tx \) of the Bezier curve.](image)

As you can see from the image, this approach also failed. As a result of bending the Bezier curve, the surface sign \( z = ty - tx \) changes.

A convenient way to construct the Bezier curve is the de Casteljau’s algorithm (Fig. 3) [4]. Although it does not provide an analytical description of the curve, it is noteworthy. Using the reference points \( P_0, P_1, P_2 \) for each value of the parameter \( t \), the position of the segment \( Q_0Q_1 \) is determined. This segment lies on a line tangent to the desired curve. The point of tangency, and hence the point of the desired curve, is the point \( R \). Thus, point by point, the Bezier curve is constructed.

![Figure 3. Construction of the Bezier curve by 3 points.](image)

The convenience of this algorithm is the ability to build more complex curves on a larger number of points without significantly complicating the process. At the same time, the necessary elements for building are increased. Consider the construction of a Bezier curve to 4 reference points \( P_1, P_2, P_3, P_4 \) (Fig. 4). Similarly to the previous case, for each value of \( t \), it is necessary to find a point on the curve \( A \) lying on the tangent \( R_0R_1 \). The position of point \( A \) in this segment is determined by the parameter \( t \). In turn, the position of the ends of the segment \( R_0 \) and \( R_1 \) is determined by the parameter \( t \) on the segments \( Q_0Q_1 \) and \( Q_1Q_2 \), respectively. The position of the points \( Q_0, Q_1, Q_2 \) is determined by the parameter \( t \) on the segments \( P_0P_1, P_1P_2, P_2P_3 \).

![Figure 4. Construction of the Bezier curve by 4 points.](image)
Consider another way to construct a smooth curve using the example of a polynomial of the third degree:

\[ y = ax^3 + bx^2 + cx + d \]

The position of the curve is determined by 4 points \( P_0(x_0, y_0), P_1(x_1, y_1), P_2(x_2, y_2), P_3(x_3, y_3) \) so that:
1. The curve passes through the point \( P_0 \).
2. The curve passes through the point \( P_3 \).
3. The first derivatives of adjacent splines at the point \( P_0 \) are equal.
4. The first derivatives of adjacent splines at the point \( P_3 \) are equal.

Points \( P_1 \) and \( P_2 \) determine the position of the tangent to the curve at points \( P_0 \) and \( P_3 \), i.e. determine the derivative. Then these conditions are described by a system of equations:

\[
\begin{align*}
  y_0 &= ax_0^3 + bx_0^2 + cx_0 + d \\
  y_3 &= ax_3^3 + bx_3^2 + cx_3 + d \\
  \frac{y_1 - y_0}{x_1 - x_0} &= 3ax_0^2 + bx_0 + c \\
  \frac{y_2 - y_3}{x_2 - x_3} &= 3ax_3^2 + bx_3 + c
\end{align*}
\]

The solution to this system of equations are the coefficients of the curve \( a, b, c, d \). The results of constructing such a curve are presented in Figure 5.

![Figure 5](image)

**Figure 5.** Construction of a smooth cubic spline using the tangent.

As can be seen from the images presented, all the conditions are satisfied and a smooth line is obtained. Applying this approach is also convenient to construct more spline segments. However, as can be seen from the third image, in this way it is not possible to construct closed contours - the curve bends along the tangents in the opposite direction. The considered approaches to constructing a smooth curve do not allow obtaining an analytically defined curve capable of supporting various configuration variations. However, the previously discussed de Casteljau’s algorithm can be used to obtain areas of the curve value, which in turn is its analytical description. Each tangent to a Bezier curve defines two half-planes with a positive and negative value of the function space and zero on the boundary. It is possible to apply set-theoretic operations to these areas. The intersection of the areas of two lines will give them a common positive area. Figure 6 shows the intersection of the domains of the tangent functions obtained for \( t=t_1 \) and \( t=t_2 \).

![Figure 6](image)

**Figure 6.** The intersection of the positive domains of the functions of two tangents.
The sequential intersection of the domains of the functions of all the tangents necessary for constructing the curve will allow us to obtain the region of the curve function with positive values inside its region, negative outside it and zero on the boundary.

3.2. Voxel modeling based on R-functions

The solution of this problem requires an analytical representation of the function and the use of set-theoretic operations on them. Set-theoretic operations on the analytical representation of functions are carried out by the method of R-functional modeling [1, 2]. The complete system of R-functions defines binary intersection and union operations, as well as the unary operator of negation of analytically defined functions $X$ and $Y$:

$$X \land \alpha Y = \frac{1}{1 + \alpha} \left( X + Y - \sqrt{X^2 + Y^2 - 2\alpha XY} \right)$$

$$X \lor \alpha Y = \frac{1}{1 + \alpha} \left( X + Y + \sqrt{X^2 + Y^2 - 2\alpha XY} \right)$$

$$X \equiv -Y$$

The use of these operations in the calculation of complex expressions requires thorough mathematical training. Computer calculation of expression values at all points of analytical objects directly depends on the complexity of the calculated expressions and requires significant resources for storing and processing the necessary information. However, the computing apparatus of R-functions fulfills the condition necessary for the functional-voxel method of modeling — a positive value inside the domain of the function, negative outside it and zero at the boundary. Thus, it is possible to combine these approaches.

The function-voxel modeling method allows to obtain an analytical representation of any function by means of proportional voxel images. Each voxel image stores the local geometric characteristics of the function at each point of the voxel space [3]. The application of the functional-voxel approach to the calculation of R-functional expressions makes it possible to simplify the calculation of the resulting function. To construct the functional area of the spline, as indicated in the first section, the method requires the consistent application of the operation of intersecting functions. The indicated prerequisite of the functional-voxel approach is carried out for any value of the parameter $\alpha$ of the complete system of R-functions. Therefore, to solve the problem, it is possible to simplify the intersection operation for $\alpha = 1$:

$$X \land_1 Y = 0.5 \left( X + Y - \sqrt{X^2 + Y^2 - 2XY} \right) = 0.5 \left( X + Y - \sqrt{(X - Y)^2} \right) = 0.5(X + Y - |X - Y|)$$

Multiplying the resulting function by 0.5 has no effect on the sign of the expression. Therefore, in the framework of the task, this can be neglected:

$$X \land_1 Y = X + Y - |X - Y|$$

Thus, the calculation of the desired expression can be divided into a sequence of four actions:
1. Addition of domains of functions $X$ and $Y$.
2. Subtraction of the domains of functions $X$ and $Y$.
3. Determination of the modulus of the difference of the domains of functions $X$ and $Y$.
4. Determination of the difference of the sum of the domains of the functions $X$ and $Y$ and the modulus of the difference of the domains of the functions $X$ and $Y$.

The description of the function within the framework of functional voxel modeling, as already mentioned, is a set of voxel images (model images or M-images) that describe the local geometric characteristics at each point of the function region - the normal components of increased dimension.

Consider the intersection of two diagonal lines. In Fig. 7, in the first row, four M-images of the first straight line $C_1$, $C_2$, $C_3$, $C_4$ and the image of its functional region $Z$ are shown - blue on negative values and gray on positive. The images $C_1$, $C_2$, and $C_4$ describe the slope of the normal components $n_1$, $n_2$, $n_3$ at each point of the image to the coordinate axes $OX$, $OY$, and $OZ$, and the image $C_4$ characterizes
the component \( n_4 \), which ensures the tangent position at the corresponding point in space. Corresponding images of the second straight line are presented in the second row.

Figure 7. M-images and functional areas of two lines.

The values \( n_1, n_2, n_3, n_4 \) obtained from the images \( C_1, C_2, C_3, C_4 \) are used in further calculations of the above four steps, for each of which its own \( n_1, n_2, n_3, n_4 \) will be found. For the first line \( X \), we denote them as \( n_1^X, n_2^X, n_3^X, n_4^X \). For the second line \( Y \), respectively, \( n_1^Y, n_2^Y, n_3^Y, n_4^Y \). The sum of the ranges of values of the functions \( X \) and \( Y \) is defined as:

\[
\begin{align*}
n_1^{X+Y} &= n_1^X n_3^Y + n_1^Y n_3^X \\
n_2^{X+Y} &= n_2^X n_3^Y + n_2^Y n_3^X \\
n_3^{X+Y} &= n_3^X n_3^Y \\
n_4^{X+Y} &= n_4^X n_3^Y + n_4^Y n_3^X
\end{align*}
\]

The difference between the ranges of values of the functions \( X \) and \( Y \):

\[
\begin{align*}
n_1^{X-Y} &= n_1^X n_3^Y - n_1^Y n_3^X \\
n_2^{X-Y} &= n_2^X n_3^Y - n_2^Y n_3^X \\
n_3^{X-Y} &= n_3^X n_3^Y \\
n_4^{X-Y} &= n_4^X n_3^Y - n_4^Y n_3^X
\end{align*}
\]

To calculate the absolute value of the difference between \( X \) and \( Y \), it is necessary to calculate a local function at each point of a given region of space:

\[
\begin{align*}
z &= -n_1^{X-Y} x - n_2^{X-Y} y + n_4^{X-Y} \\
n_3^{|X-Y|} &= n_3^{X-Y} \\
z < 0, n_1^{X-Y} &= 1 - n_1^{X-Y} \\
z > 0, n_1^{X-Y} &= n_3^{X-Y}
\end{align*}
\]

The last action is the difference obtained at stages 1 and 3 of the subexpressions:

\[
\begin{align*}
n_1^{X+Y-[X-Y]} &= n_1^{X+Y} n_3^{[X-Y]} - n_1^{[X-Y]} n_3^{X+Y} \\
n_2^{X+Y-[X-Y]} &= n_2^{X+Y} n_3^{[X-Y]} - n_2^{[X-Y]} n_3^{X+Y} \\
n_3^{X+Y-[X-Y]} &= n_3^{X+Y} n_3^{[X-Y]} \\
n_4^{X+Y-[X-Y]} &= n_4^{X+Y} n_3^{[X-Y]} - n_4^{[X-Y]} n_3^{X+Y}
\end{align*}
\]
The obtained values $n_{1,2,3,4}$ are used to construct four M-images and image the range of values of the resulting function (Fig. 8).

Thus, the operation of intersecting the regions of two functions is realized by applying the functional-voxel approach.

3.3. Voxel modeling of a Bezier curve based on R-fuctions

The operations discussed in section 3.2 allow us to construct the functional area of the Bezier curve in the way suggested at the end of section 3.1 (Fig. 1). To do this, iteratively compute the intersection of the function regions of each subsequent tangent to the curve:

1. Find the range of values of the first tangent curve ($t = 0$).
2. Find the range of values of the next tangent to the curve ($t = t + \Delta t$, where $\Delta t$ is the step of changing $t$ over the entire interval from 0 to 1).
3. Intersect the areas of the data functions of two tangents.
4. Continue building up $t$ to obtain a tangent function at each stage of the region and calculate the intersection of the region of the value of each new tangent with the function obtained in the previous step.

Figure 9 shows an example of a curve constructed as a result of applying the proposed approach — four M-images and a functional representation of the region. The extreme right image shows the reference points of the curve (red) and all the tangents used for calculations (black).

In a similar way, a curve is constructed based on a larger number of points. Figure 10 shows the results of constructing the function areas of the curves of various configurations, constructed from 4 points. Applying the data approach, as can be seen from the figures, the construction of closed areas is also possible.

4. Conclusion

Such a computer model will make it possible to use the presented space of a parametric function in the procedures of R-functional modeling, which adds a whole class of CAD models to the developed tools that provide the design of complex curved surfaces.
References

[1] Rvacev V A 1982 *Theory of R-functions and Some Applications* (Kiev: Naukova Dumka) p 552
[2] Sheiko T I, Maksimenko-Sheiko K V, Litvinova Yu S ana Lisin D A 2017 R-functions and chevron surfaces in machine building. *J. Problemy mashinostroeniya*. 20(2) pp 54-60
[3] Tolok A V 2016 *Functional Voxel Method in Computer Modeling* (Moscow: Fizmatlit) p 112
[4] Karpov D A and Struchenkov V I 2016 Dynamic Programming as a Method of Spline Approximation in the CAD Systems of Linear Constructions *Rossiyskiy tekhnologicheskiy zhurnal* (Russian Technological Journal) 7(3) pp 77-88
[5] Wedel A, Franke U, Badino H and Cremers D 2008 B-spline modeling of road surfaces for freespace estimation *IEEE Intelligent Vehicles Symposium* pp 828-833
[6] Zhang X 2018 *CAD-based geometry parametrisation for shape optimisation using Non-uniform Rational B-splines* (Doctoral dissertation, Queen Mary University of London).
[7] Lian H, Kerfriden P and Bordas S P A 2017 Shape optimization directly from CAD: An isogeometric boundary element approach using T-splines *Computer Methods in Applied Mechanics and Engineering* 317 pp 1-41.
[8] Guo M, Wang W, Zhao G and Du X 2019 Bézier Segmentation of T-spline Solids in Parametric Domain *J. Computer-Aided Design & Applications* 17(3) pp 502-512