

Research Article

Iterative Least Square Optimization for the Weights of NURBS Curve

Tuo-Ran Wang ¹, Ning Liu ¹, Lei Yuan ², Ke-Xin Wang ³, and Xian-Jun Sheng ¹

¹School of Electrical Engineering, Dalian University of Technology, Dalian 116025, China

²Institute of Space Long March Vehicle, Beijing 100048, China

³School of Mechanical Engineering, Dalian University of Technology, Dalian 116025, China

Correspondence should be addressed to Ning Liu; liun@dlut.edu.cn

Received 19 November 2021; Revised 9 March 2022; Accepted 29 March 2022; Published 22 April 2022

Academic Editor: George S. Dulikravich

Copyright © 2022 Tuo-Ran Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

NURBS curves have been widely applied in the field of data points approximation, and their fitting accuracy can be improved by adjusting the values of their weights. When applying the NURBS curve, it is difficult to obtain the optimal weights values due to the nonlinearity of the curve fitting problem with NURBS. In this paper, a weights iterative optimization method for NURBS curve fitting is proposed, where the geometric property of weight has been adopted to iteratively obtain the adjusting values of the weights with the least square method. The effectiveness and convergence of the proposed method are demonstrated by numerical experiments. The results show that the proposed method can obtain higher fitting accuracy than other iterative optimization methods. Meanwhile, it has the merits of data noise robustness, high accuracy with small-scale knots, and flexibility. Hence, the proposed method is suitable for applications including noisy data approximation and skinned surface generation.

1. Introduction

With the excellent mathematical properties and modeling flexibility, nonuniform rational B-splines (NURBS) curves have been widely used in many fields. For example, in reverse engineering, NURBS curves are usually applied to reconstruct item contours from sampled points. One key issue in NURBS engineering application is how to improve fitting accuracy. Motivated by this requirement, researches have performed on control point optimization, knot vector optimization, the parameters of data points optimization, and weights optimization.

The least square method is usually used to obtain the control points [1]. However, it cannot utilize the last calculation result in its computing process, is not suitable to solve big data, and is difficult to optimize the fitting curve locally. To this end, progressive iterative algorithm (PIA) and its derivative algorithms [2, 3] are proposed to optimize the control points. Although these methods effectively solve problems such as big data solving and local optimization, the

total solving efficiency is lower than the least square method. In addition, they have certain convergency problems.

The knot vector optimization methods can be divided into two categories: one based on the geometric information of data points, and one based on intelligent algorithms. Li uses the discrete curvature information of data points, picks up the curvature direction changed points, and integrates the curvature to achieve the knots selection [4]. Park treats the local curvature maximum points as the feature points and generates knots from them [5]. Liang obtains the geometric information by generating the initial fitting curve and optimizes knots by that [6]. Aguilar groups the data points by curvature then inserts and adjusts knots to achieve higher accuracy [7]. Yeh utilizes the discontinuity of each derivative of B-spline bases function to construct the characteristic equation and gets the knot vector [8]. Laube collects a series of geometric parameters and then uses support vector machine to set the knots [9]. Its efficiency and accuracy are excellent but need to be trained by a large number of samples, which is not easy to promote. Tegoeh

splits data in a bisecting method to obtain coarse knots and employs a nonlinear least squares technique to optimize knots [10]. Various intelligent optimization algorithms are applied to optimize the knot vector [11–18]. Despite having high fitting accuracy, intelligent optimization algorithms have a long calculation time and depend on the initial value. Knot vector optimization can improve the accuracy of fitting curves efficiently by optimizing the value of knots. However, the change of knot is not suitable to extend the curve fitting to skinned surface fitting. Meanwhile, the method of optimizing the knot vector needs to increase knots for accuracy improvement. The fitting curve with a large number of knots is sensitive to data noise.

There are relatively few researches on optimizing the parameters of data points and weights. Ma proposes the base curve projection method to optimize the parameters of points, but the optimization mechanism is not explained [19]. The symmetric eigenvalue decomposition is used to estimate the values of weights [20]. However, this method has no clear geometric meaning and calculates clumsily for special calculation examples. Zhang uses the simulated annealing algorithm to optimize weights [21]. This method is inefficient and relies on experience in the parameters set. Pandithevan proposes a method to modify the weights of curve iteratively, but it can only be used in two dimensions and cannot be extended to three dimensions [22]. Meng proposes a method based on the least square progressive iterative algorithm (LSPIA), which optimizes the weights by calculating the approximate partial derivative and combining the dichotomy, but the optimization is not effective [23].

In summary, most of the current researches focus on the optimization of B-spline curves, lacking the discussions on weights optimization of NURBS curves. Limited researches on weights optimization have problems, including the uncertainty of geometric meaning, calculation clumsily, inextensibility to three-dimension, and ineffective optimization. To solve these aforementioned problems, a weights iterative optimization method for NURBS curve fitting is proposed in this paper. In the proposed method, the geometric property of weight has been adopted to iteratively obtain the adjusting values of the weights with least square method, which improves the fitting accuracy. Compared with the existing researches, the proposed method has the merits of flexibility and simplicity of implementation, noise robustness, clear geometric meaning, and high accuracy.

The rest of the paper is organized as follows: a brief review of NURBS curve notations and basic steps of curve fitting is in Section 2, details on the proposed method are provided in Section 3, and calculation results and discussions are provided in Section 4. Finally, conclusions are drawn in Section 5.

2. NURBS Curve Fitting Notations

NURBS curve is expressed by

$$C(u) = \frac{\sum_{i=0}^n \omega_i \cdot V_i \cdot N_{i,k}(u)}{\sum_{i=0}^n \omega_i \cdot N_{i,k}(u)}, \quad (1)$$

where V_i is the control point of the curve, ω_i is the weight of each control point, and $N_{i,k}(u)$ is the k th order B-spline basis function at parameter u defined by a given sequence of nondecreasing knot vector and deBoor recursive equation:

$$N_{i,0}(u) = \begin{cases} 1, & U_i \leq u \leq U_{i+1}, \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

$$N_{i,k}(u) = \frac{u - U_i}{U_{i+k} - U_i} N_{i,k-1}(u) + \frac{U_{i+k+1} - u}{U_{i+k+1} - U_{i+1}} N_{i+1,k-1}(u).$$

The corresponding parameters of ordered data points Q_j , $j = 1, \dots, m$ on NURBS curve $\{t_1, t_2, \dots, t_m\}$, order of NURBS curve and knot vector $U = \{u_1, u_2, \dots, u_{n+k+1}\}$ need to be defined. The knot vector is normalized generally. To avoid singular equations in the process of getting control points, the knot vector is calculated after the parameters of data points are defined. The parameters range from 0 to 1. Define $t_1 = 0$ and calculate other parameters by cumulative chord length method ($e = 1$):

$$t_i = \frac{\sum_{j=1}^{i-1} \|Q_{j+1} - Q_j\|^e}{\sum_{j=1}^{m-1} \|Q_{j+1} - Q_j\|^e} \quad i = 2, 3, \dots, m; e \geq 0. \quad (3)$$

Although this method is simple and fast, it cannot get a high accuracy fitting curve when the number of knots is small.

To control the beginning and end location of the fitting curve, the knot vector is set to be clamped, which means the first and last $k+1$ knots are the same. Then, other knots are calculated by NKTP method [24] to guarantee solvability:

$$d = \frac{m}{n - k},$$

$$i = \text{int}(j d), \alpha = j d - i, \quad (4)$$

$$u_{k+j+1} = (1 - \alpha)t_i + \alpha t_{i+1}, \quad j = 1, 2, \dots, n - p.$$

The B-spline basis function of each data point can be obtained according to parameter t_j and knot vector U through (2), and the matrix A can be built by

$$A = \begin{bmatrix} N_{1,k}(t_1) & \cdots & N_{n,k}(t_1) \\ \vdots & & \vdots \\ N_{n,k}(t_m) & \cdots & N_{n,k}(t_m) \end{bmatrix}. \quad (5)$$

The weights are calculated using the eigenvector method [24] in advance. Otherwise, set them to the same value to degenerate the NURBS form to the B-spline form. Then, the objective function is constructed:

$$f_1 = \sum_{i=1}^m |Q_i - C(t_i)|^2. \quad (6)$$

To minimize the objective function, the derivatives of f_1 with respect to each control point, $V_i (i = 1, \dots, n)$, are equal to 0:

$$\frac{\partial f_1}{\partial V_i} = 0. \quad (7)$$

Finally, obtain the control points using the following:

$$V = (A^T A)^{-1} A^T (P A \omega), \quad (8)$$

where P is the coordinate vector of data points and ω is the vector of weights.

Although the solution minimizes the distances between data points and corresponding points on the fitting curve, it is not the optimal solution to get the highest accuracy. So, the parameter of data point, knot vector, and weights still need to be optimized.

3. Iterative Optimization of Parameters and Weights

3.1. Optimization of Parameters. The parameters of data points on the final fitting curve cannot be estimated directly because of the inaccuracy of the traditional parameterization method. It is the reason that the small-scale knots NURBS curve is difficult to get high accuracy. So, the number of knots is increased to improve the accuracy. However, its increase not only occupies more computing resources but also deteriorates the robustness of curve fitting to data noise, which leads to the deformation of the fitting curve. So, it is necessary to improve the fitting accuracy by optimizing the parameters of data points.

The accuracy of fitting curve can be evaluated by the following objective function:

$$f_2 = \sum_{i=1}^m \|C(t_i) - Q_i\|, \quad (9)$$

where $C(t_i)$ is the location corresponding to data point on NURBS curve, Q_i is the location of data point, and $\| \cdot \|$ is the Euclidean distance.

For the sake of simplicity, the value of t_i is generally set to be the data parameters used in curve fitting. However, it is not the closest location to the data point on the fitting curve because of the inaccuracy of the parameters. Figure 1 shows the locations of data points and fitting curve. For the data point Q_j , $P(t_j)$ is the estimated location of Q_j on fitting curve, and $P(t_j^*)$ is the closest point on fitting curve. Therefore, the objective function f_2 can be reduced by altering the parameters of data points without changing the fitting curve.

Obviously, when each parameter of data point is set to be the value of the closest point on the fitting curve, the objective function f_2 reaches the minimum value.

In the previously mentioned optimization process, fitting curve is invariant, and only the parameters of data points are changed. However, their change leads to the change of the B-spline basic function matrix A , which results in the fact that the control points obtained before optimization do not satisfy (7). In other words, the current solution is not the least square solution anymore. So, the control points through (8) after updating matrix A need to be calculated. The new fitting curve reduces the objective function further. The previously mentioned two steps make up once optimization of data points parameters. Under the condition of the current knot vector and weights, the fitting

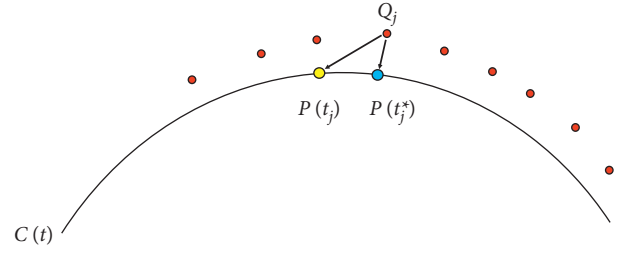


FIGURE 1: Locations of data points and fitting curve.

accuracy can be improved by implementing parameters optimization many times.

3.2. Calculation of the Closest Points on NURBS Curve. To achieve the optimization of data points parameters and get the value of fitting accuracy, the closest points on NURBS curve need to be calculated. So, the following objective function needs to be built:

$$g(u) = C(u) \cdot (C(u) - P). \quad (10)$$

According to the continuity of the curve, the closest location on the NURBS curve is the point whose parameter makes $g(u) = 0$. It can be calculated by using Newton iterative method with the convergence condition [1]:

$$\begin{aligned} u_{i+1} &= u_i - \frac{g(u_i)}{g'(u_i)} \\ &= u_i - \frac{C'(u) \cdot (C(u_i) - P)}{C''(u_i) \cdot (C(u_i) - P) + |C'(u_i)|^2}. \end{aligned} \quad (11)$$

The initial value of the iteration can be obtained by sampling. Specifically, the interval of sampling is set to be fixed on the parameter domain. The distance between each sampled point and test point is computed; then, the parameter of minimum distance point is selected as the initial value of Newton iterative method. The size of the interval can be set according to the complexity of NURBS curve.

In once optimization of data points parameters, the value of data parameter changes slightly. So, the initial value of iteration can be set as the parameter used in the last least square method fitting. A large number of examples prove the effectiveness of this initial value selection method. It can accurately obtain the closest point while improving efficiency.

3.3. Optimization of Weights. The weights of the NURBS curve have a clear geometric meaning. For a fixed point on NURBS curve, it will move close to the corresponding control point when increasing the weight of the control point adjacent to it. Similarly, when the weight is decreased, the point will leave the corresponding control point. The influence of weight is shown in Figure 2.

To improve the accuracy, the geometric property of weight can be used to optimize the fitting curve. Figure 3 shows the locations of the NURBS curve, control points, and data points.

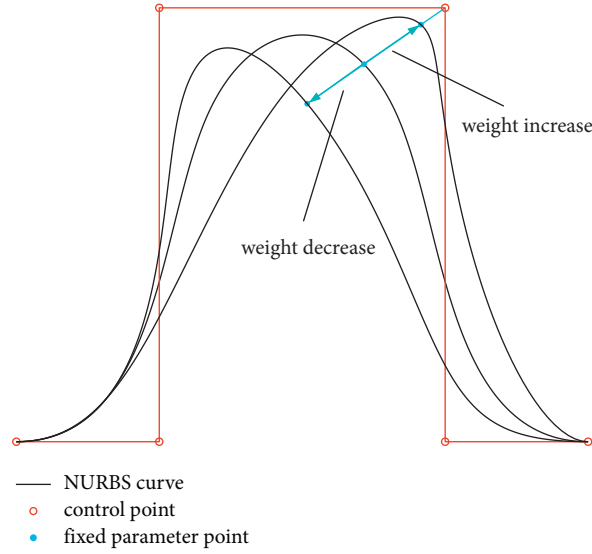


FIGURE 2: Influence of weight on NURBS curve.

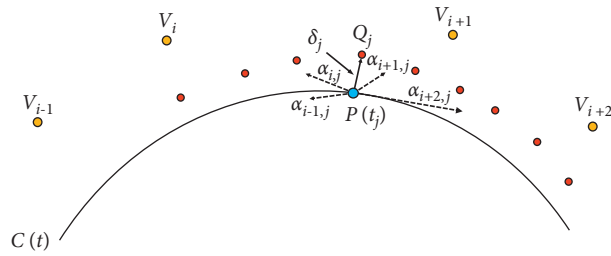


FIGURE 3: Locations of NURBS curve, control points, and data points.

First, optimizing the single location of NURBS curve is discussed. For simplification, we assume that the influence of the weight on curve is linear. Specifically, assume the weight alteration of single control point V_i is linear to the move of

the location at parameter t_j . The extent and direction of influence can be represented by the partial derivative of the curve with respect to the weight, and it is calculated by

$$\begin{aligned} \alpha_{i,j} &= \frac{\partial C(u)|_{u=t_j}}{\partial \omega_i} \\ &= \frac{\omega_i \cdot N_{i,k}(t_j) \cdot \sum_{i=0}^n \omega_i \cdot N_{i,k}(t_j) - N_{i,k}(t_j) \cdot \sum_{i=0}^n \omega_i \cdot V_i \cdot N_{i,k}(t_j)}{\left(\sum_{i=0}^n \omega_i \cdot N_{i,k}(t_j)\right)^2}, \end{aligned} \quad (12)$$

where i is the number of control point and j is the number of data point.

Then, the alteration of weight $\Delta \omega_i$ can be calculated according to the distance δ_j between Q_j and $P(t_j)$ by

$$\Delta \omega_i = \frac{\delta_j}{\alpha_{i,j}}. \quad (13)$$

A single alteration is not effective enough due to linearization. Like the Newton iterative method, the alterations

can be implemented many times with convergence conditions to improve the accuracy. However, the alteration of a single weight can only move the point in a certain direction. In order to achieve the alteration of any direction, multiple weights of control points need to be altered.

There is a new problem that the influence of different weights will couple with each other. To simplify calculations, their influences are linearized either. Then, the coefficient matrix B is constructed according to the partial derivatives of the curve with respect to different weights at the current

parameter. The following linear equations in the form of matrix can be solved to get the alterations μ of weights:

$$B\mu = \delta. \quad (14)$$

The alterations of every weight should be as small as possible for reducing the error of linearization. And the precise adjustment of the curve can be achieved by multiple times calculation of alterations.

In actual curve approximation, it is hard to make the fitting curve passing through every data point due to the limited number of control points. Therefore, the least square method can be used to reduce the fitting error as a compromise. The objective function is defined as follows:

$$\begin{aligned} h &= \sum_{j=1}^m \left(\delta_j - \sum_{i=1}^n \mu_i \alpha_{i,j} \right)^2 \\ &= \sum_{j=1}^m \left(\delta_j - \sum_{i=1}^n \mu_i \alpha_{i,j} \right) \cdot \left(\delta_j - \sum_{i=1}^n \mu_i \alpha_{i,j} \right) \\ &= \sum_{j=1}^m \left(\delta_j \cdot \delta_j - 2 \sum_{i=1}^n (\mu_i \alpha_{i,j}) \cdot \delta_j + \sum_{i=1}^n \mu_i \alpha_{i,j} \cdot \sum_{i=1}^n \mu_i \alpha_{i,j} \right). \end{aligned} \quad (15)$$

It is the function of n variables μ_1, \dots, μ_n . To minimize the function value h , let the partial derivatives of h with respect to each variable equal 0. Then, the equation about variable μ_l can be listed:

$$\begin{aligned} \frac{\partial h}{\partial \mu_l} &= \sum_{j=1}^m \left(-2\alpha_{l,j} \cdot \delta_j + 2\alpha_{l,j} \sum_{i=1}^n \mu_i \alpha_{i,j} \right) \\ &= 0. \end{aligned} \quad (16)$$

After processing, n equations can be constructed:

$$\sum_{j=1}^m \sum_{i=1}^n \mu_i \alpha_{i,j} \cdot \alpha_{i,j} = \sum_{j=1}^m \alpha_{l,j} \cdot \delta_j. \quad (17)$$

The equations are constructed in the matrix form as follows:

$$\begin{pmatrix} \sum_{j=1}^m \alpha_{1,j} \cdot \alpha_{1,j} & \sum_{j=1}^m \alpha_{1,j} \cdot \alpha_{2,j} & \cdots & \sum_{j=1}^m \alpha_{1,j} \cdot \alpha_{n,j} \\ \sum_{j=1}^m \alpha_{2,j} \cdot \alpha_{1,j} & \sum_{j=1}^m \alpha_{2,j} \cdot \alpha_{2,j} & \cdots & \sum_{j=1}^m \alpha_{2,j} \cdot \alpha_{n,j} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{j=1}^m \alpha_{n,j} \cdot \alpha_{1,j} & \sum_{j=1}^m \alpha_{n,j} \cdot \alpha_{2,j} & \cdots & \sum_{j=1}^m \alpha_{n,j} \cdot \alpha_{n,j} \end{pmatrix}_{n \times n} \cdot \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{pmatrix}_{n \times 1} = \begin{pmatrix} \sum_{j=1}^m \alpha_{1,j} \cdot \delta_j \\ \sum_{j=1}^m \alpha_{2,j} \cdot \delta_j \\ \vdots \\ \sum_{j=1}^m \alpha_{n,j} \cdot \delta_j \end{pmatrix}_{n \times 1}. \quad (18)$$

When solving the previously mentioned equation, the left matrix is in ill-condition generally. To prevent the influence of truncation error, the truncated SVD method is used to solve the equation [25]. After the decomposition, the value of the minimum and other singular values differ by several orders of magnitude. So, the minimum value is removed to acquire the solution stability.

The solution is the alterations of the weights that minimize the error. The optimization of weights is realized by adding the alteration to the weights of the current curve. Combined with optimization of parameters, optimizing the weights of the curve a couple of times can improve the fitting accuracy iteratively.

It is worth mentioning that the previously mentioned weights optimization method is implemented based on the existing fitting curve. So, it has strong flexibility that can combine with other optimization methods to generate a higher accuracy curve.

3.4. Optimization Process of NURBS Curve Fitting.

The NURBS curve fitting process based on the optimization of

parameters and weights is shown in Figure 4, which includes the following steps.

Step 1. Set the number of knots and generate a knot vector with NKTP method. The number of knots should be set according to the number of turns of data points.

Step 2. Use the least square method to generate the initial NURBS fitting curve. Optimize the parameters of data point and the weights of the curve.

Step 3. Check whether the current fitting accuracy meets the requirement. The curve fitting is completed when the accuracy reaches the prescribe fitting error. Otherwise, insert the knot at the parameter which has the maximum error, and use the algorithm of closest point search to update the parameters of data points. Then, execute Step 4.

Step 4. Calculate the new control points using the least square method according to the current knot vector, parameters of data points, and weights of control points.

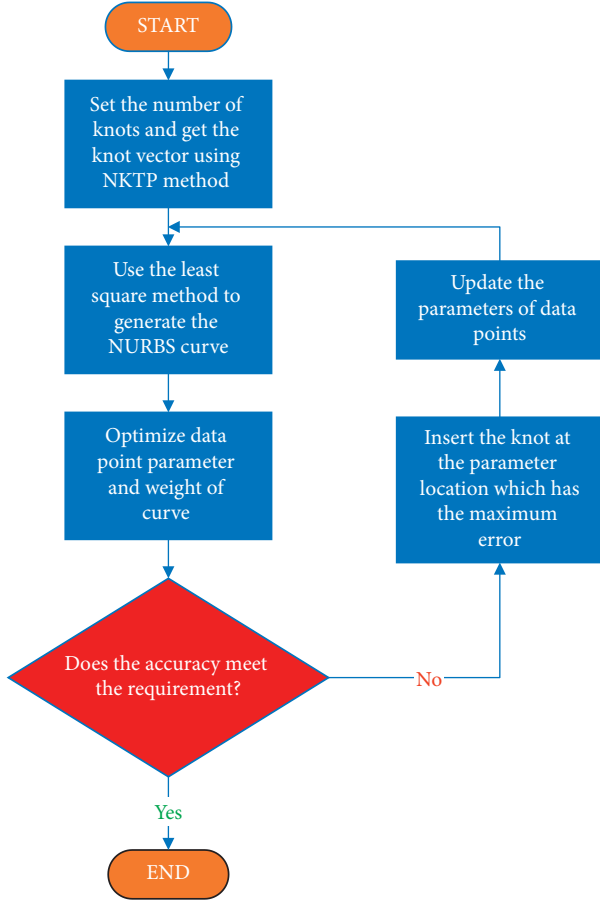


FIGURE 4: Flowchart of the NURBS curve fitting for data points.

Repeat Step 3 until the fitting accuracy is satisfied. Finally, get the fitting NURBS curve that meets the accuracy requirement.

The pseudocode version of process is shown in Algorithm 1.

4. Numerical Experiments and Discussions

To verify the effectiveness of the proposed method, the desktop computer with a 3.0 GHz Intel i7-9700 CPU and 16 GB RAM is used to approximate the examples with MATLAB software. The accuracy of the approximation can be evaluated as follows:

$$\begin{aligned}
 E_{\text{IAE}} &= \sum_{i=1}^m \frac{\|C(t_i) - Q_i\|}{Q_{\text{rng}}}, \\
 E_{\text{MAX}} &= \frac{1}{Q_{\text{rng}}} \max_i \|C(t_i) - Q_i\|, \\
 E_{\text{RMS}} &= \frac{1}{Q_{\text{rng}}} \sqrt{\frac{1}{m} \sum_{i=1}^m (C(t_i) - Q_i)^2},
 \end{aligned} \tag{19}$$

where E_{IAE} is the sum of the normalized fitting errors, E_{MAX} is the normalized max error, E_{RMS} is the normalized root mean

squared error, and $Q_{\text{rng}} = \max(x_{\text{max}} - x_{\text{min}}, y_{\text{max}} - y_{\text{min}})$ is the maximum length of an edge of the axis-aligned bounding box.

4.1. Fitting Parameters Selection. Due to the requirement of the continuity of first derivatives, at least the quadratic NURBS curve needs to be used. The continuity of second derivatives is also needed because of the implementation of data parameters optimization. At least, the cubic NURBS curve is needed. With the order of the NURBS curve increasing, the timing is increasing because of the increment of the nonzero element of (18). Meanwhile, the effect is reduced when the order of the NURBS curve rises. It is because the amount of control points that influence a single point on NURBS curve is increasing, and the optimization of weights is limited further. So, cubic is the best order to apply the proposed method.

To investigate the iterative convergence, the proposed method is implemented to reconstruct the fitting curve from the sampled data points. The number of knots is set as 14 (including the repeated clamped knots) and the knot vector is obtained by the NKTP method. The fitting curve is shown in Figure 5. The knots are represented by the small unfilled blue diamonds.

As observed, the shape of the fitting curve gradually converges to the data points after iterations. The fitting error decreases along with the increase of iterations. In order to balance the accuracy and the calculation time, an appropriate number of iterations should be selected.

The previously mentioned example is used to test the change of the RMS error with respect to the number of iterations. The results for different knots curves are shown in Figure 6. It can be observed that the rates of decrease of errors are not constant. The errors for the curves of different knots decrease rapidly during the first 10 iterations. Their rates of decrease slowed down from 10 iterations to 20 iterations. The errors are decreasing slowly after 20 iterations. And the greater the number of knots is, the fewer the iterations used to make the error stabilize. Similar situations exist in other examples. To improve the accuracy and ensure efficiency, the number of iterations is set to be 20 in the subsequent examples.

4.2. Comparison and Discussion of the Fitting Results. To verify the effectiveness of the proposed method, take the contour of “face” as an example. The performance of the proposed method is compared with other iterative methods, including NCFO method [23] and LSPIA method [3], and the method only performs optimization of data points parameters. The 375 points sampled from the side face of the cartoon character are shown in Figure 7(a). Considering that the local curvature of contour change greatly, the knot vectors of all methods are obtained using the discrete integral of data points [4]. The fitting curves are shown in Figure 7(b). Their numbers of knots are set to 24 and numbers of iterations are set to 20. The knots are represented by the small unfilled blue diamonds. The errors are shown in Table 1.

```

{Initialization}
Generate the initial fitting curve in traditional method.
for  $i = 1$  to  $m$  do
    Calculate the minimum distance location on fitting curve to data point  $Q_i$ .
end for
{Main Loop}
while (prescribe fitting error < max fitting error) do
    Insert a knot at the location where has the maximum error on fitting curve.
    for  $i = 1$  to  $m$  do
        Renew the data point parameter on fitting curve.
    end for
    Build the equation and get the alterations of the weights using proposed method.
    Get the new fitting curve using the least square method.
    for  $i = 1$  to  $m$  do
        Calculate the minimum distance location on fitting curve to data point  $Q_i$ .
    end for
end while
    
```

ALGORITHM 1: Pseudocode version of fitting data points with proposed method.

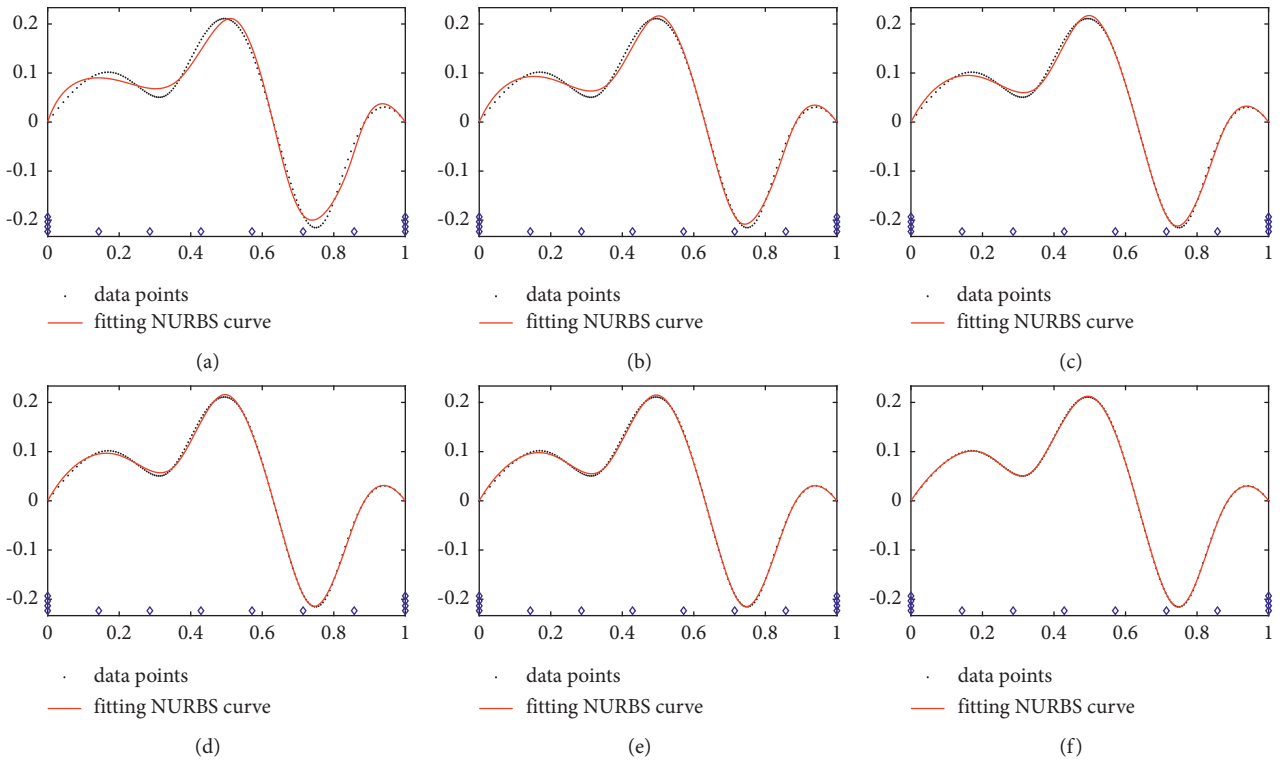


FIGURE 5: Optimizing the fitting curve iteratively.

As can be seen from Figure 7(b) and Table 1, the proposed method has the highest accuracy in all kinds of methods. Compared with the optimization without weight, E_{max} , E_{RMS} , and E_{IAE} of the proposed method are reduced by 39.4%, 16.3%, and 15%, respectively. The NCFO method has the largest maximum error because of the inaccuracy of approximate calculation. The LSPIA method has the largest RMS error and accumulated error because of the inefficiency of each iteration. Compared with these methods, the proposed method optimizes weights effectively and

approximates the large curvature locations better without increasing the overall error.

To investigate the data noise robustness of the proposed method, a curve fitting test is performed with the data points containing random errors. The 201 points sampled from a six-degree Bezier curve with random errors that follow the normal distribution $N(0, 0.001^2)$ of mean 0 and variance 0.001^2 . The data points are shown in Figure 8(a). The fitting curves are shown in Figure 8(b). The knot vectors of all methods are obtained by NKTP method. Their numbers of

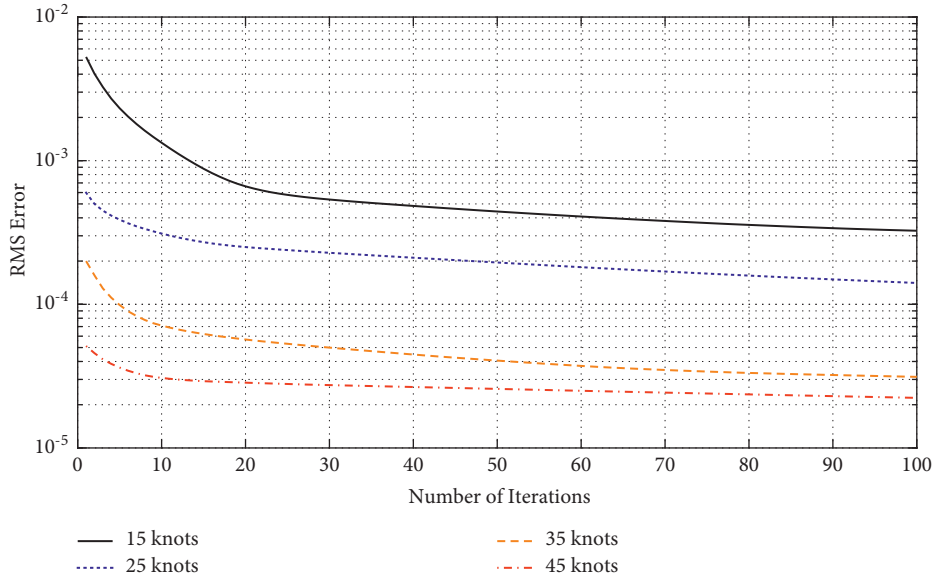


FIGURE 6: RMS error of different knots curves with the numbers of iterations.

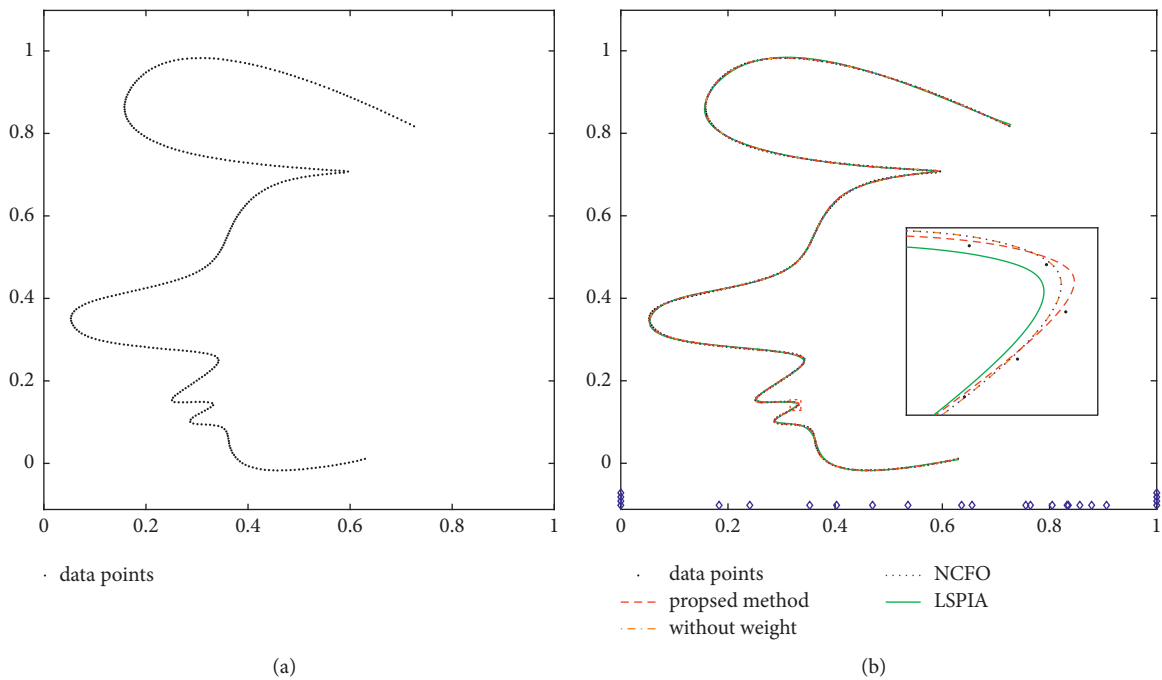


FIGURE 7: (a) Data points of "face." (b) The fitting curves of each method.

TABLE 1: The fitting errors of each method for "face."

Optimize method	Error index		
	E_{\max}	E_{RMS}	E_{IAE}
Proposed method	0.002975	0.000779	0.224482
Without weight	0.004906	0.000931	0.264127
NCFO	0.011497	0.001102	0.275200
LSPIA	0.007295	0.001402	0.417264

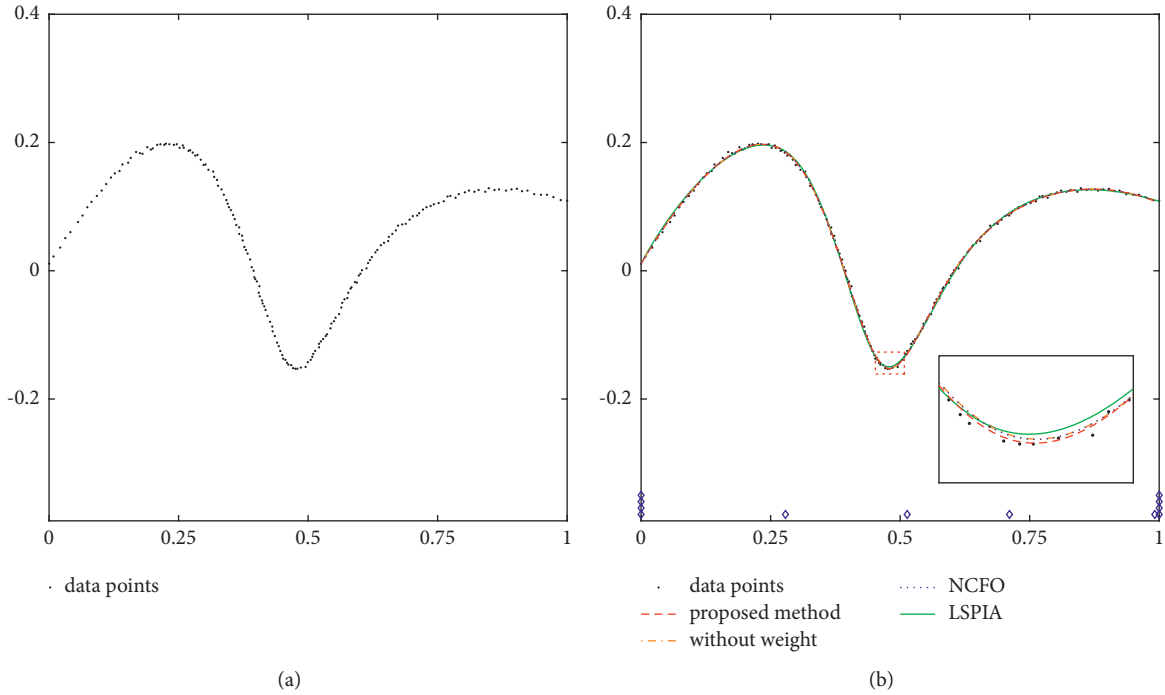


FIGURE 8: (a) Data points with noise. (b) The fitting curves of each method.

TABLE 2: The fitting errors of each method for “curve with noise.”

Optimize method	Error index		
	E_{MAX}	E_{RMS}	E_{IAE}
Proposed method	0.006322	0.001757	0.232865
Without weight	0.006871	0.001975	0.263212
NCFO	0.006796	0.001912	0.254431
LSPIA	0.007500	0.002449	0.314092

knots are set to 12, and numbers of iterations are set to 20. The knots are represented by the small unfilled blue diamonds. The errors are shown in Table 2.

From Figure 8(b) and Table 2, it can be seen that the fitting accuracy of the proposed method is still higher than other methods. Due to the least square method, the proposed method is robust to the data noise. It can fit the high-curvature location better and improve the accuracy when the data points with noise.

To test the influence of the number of knots on the fitting accuracy, the performance of the proposed method is compared with other optimization methods of knot vector through the “parameter curve” example. There are iterative methods IKI [6], AdpCrv [7], and an instant method FAKP [8].

The data points shown in Figure 9(a) are sampled from the parametric equations $x(u) = u(\cos(2u)+0.5)$ and $y(u) = u\sin(u)$. The 401 points are sampled along the arc length, with higher sampling density where curvature is higher. The sampling interval contains a variation that follows the normal distribution $N(0, 0.003^2)$ of mean 0 and variance 0.003^2 . The proposed method uses the basic NKTP method to get the knot vector, which does not optimize the knot vector. Each method runs 15 times with different numbers of knots. The size of knots is defined according to the

relative number between data points and knots. The max error and RMS error are selected as the evaluation index of fitting accuracy. The results are shown in Figures 9(b) and 9(c).

As indicated by Figures 9(b) and 9(c), the proposed method achieves the highest accuracy when the number of knots is much less than the number of data points (less than 50). The reason is that the optimization of data points in the proposed method works effectively when the knots are small-scale. When the number of knots is limited, the effect of optimization of knot vector is also limited. The correctness of each parameter of data point plays a key role in improving the accuracy of the fitting curve. So, the proposed method can fit the data points better than other optimization of knot vector methods at small-scale knots.

With the increase of knots, the fitting error of each knot vector optimization method decreases more rapidly than the proposed method. Actually, the optimizations of knots vector and data points parameters are essentially optimizing the value of B-spline basic function to each data parameter. They improve the fitting accuracy by optimizing the coefficient matrix A . The knot vector can improve the accuracy effectively when the number of knots increases. Meanwhile, the optimization effect of the data parameters is limited at large-scale knots. So, the accuracy of the proposed method is less than the FAKP method and the AdpCrv method at the number of knots is more than 50. The IKI method has the highest error because it inserts the knot at the middle of the segment, which is less effective for the data with local high curvature.

It is worth mentioning that weight optimization is not suitable for the fitting of curves with discontinuous parts. However, due to the strong flexibility of the proposed method, an appropriate optimization of knot vector can be

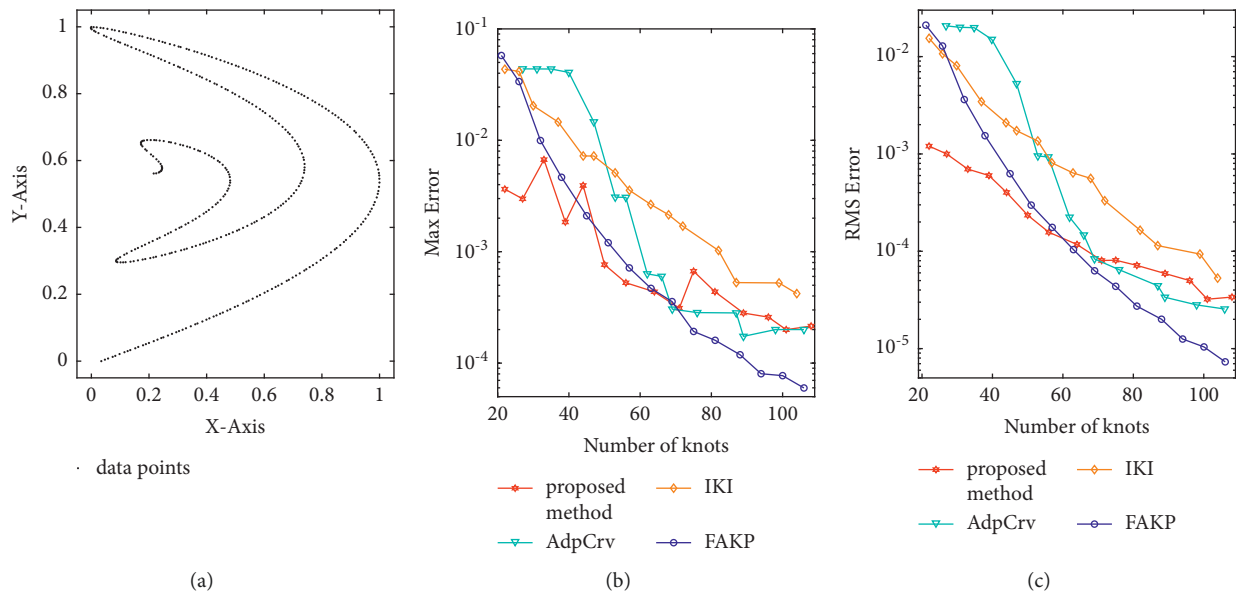


FIGURE 9: (a) Data points of “parameter curve.” (b) Maximum errors of each method with respect to different numbers of knots. (c) RMS errors of each method with respect to different numbers of knots.

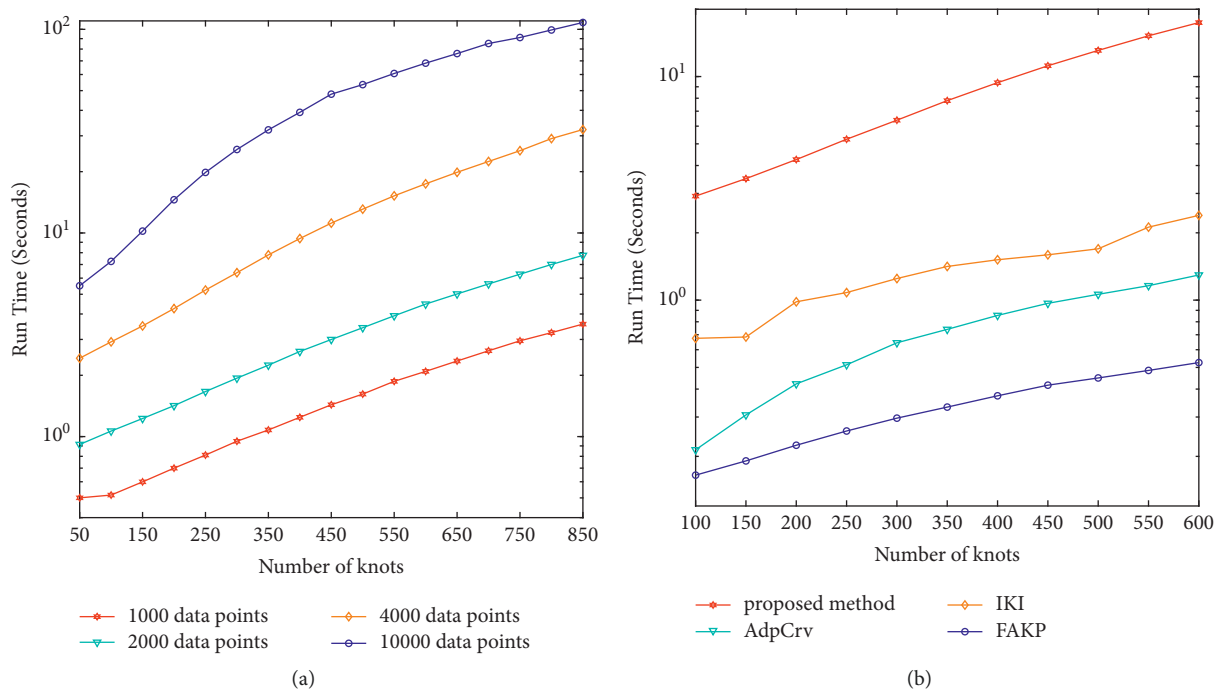


FIGURE 10: (a) Timing of the proposed method under different parameters. (b) Timing of each method with respect to different numbers of knots.

selected to make discontinuous parts on fitting curve. After that, the proposed method can improve the accuracy of the fitting curve further.

4.3. Running Time Experiment. In order to determine the running time of the proposed method, the time experiment is conducted. The proposed method operates on the dataset sampled from the parametric curve shown in Figure 9(a). Different numbers of data points are sampled

to determine the influence of the number of data points. The convergence criteria are set to be a fixed ratio of RMS error fall, which is equal to 5%. The result is shown in Figure 10(a). The running time of the proposed method is linear to the number of data points and knots. So, it is feasible to solve the large-scale problem in the proposed method. The time experiments of other optimization methods are conducted. The number of data points sampled from the parametric curve is set to be 4000. The result is shown in Figure 10(b).

```

{u direction fits}
Set the number of knots in the u direction and generate a uniform knot vector.
for  $i=1$  to  $l$  do
    Fit the  $i$ th row data points and get a NURBS curve in the proposed method.
end for
{v direction fits}
Set the number of knots in the v direction and generate a uniform knot vector.
for  $i=1$  to  $m$  do
    Fitting the  $i$ th column control points of every NURBS curve in 4 dimensions.
end for
{generate surface}
Generate the NURBS surface using the control points obtained in v direction fitting and the knot vectors of u and v direction.

```

ALGORITHM 2: Pseudocode version of fitting skinned surface with proposed method.

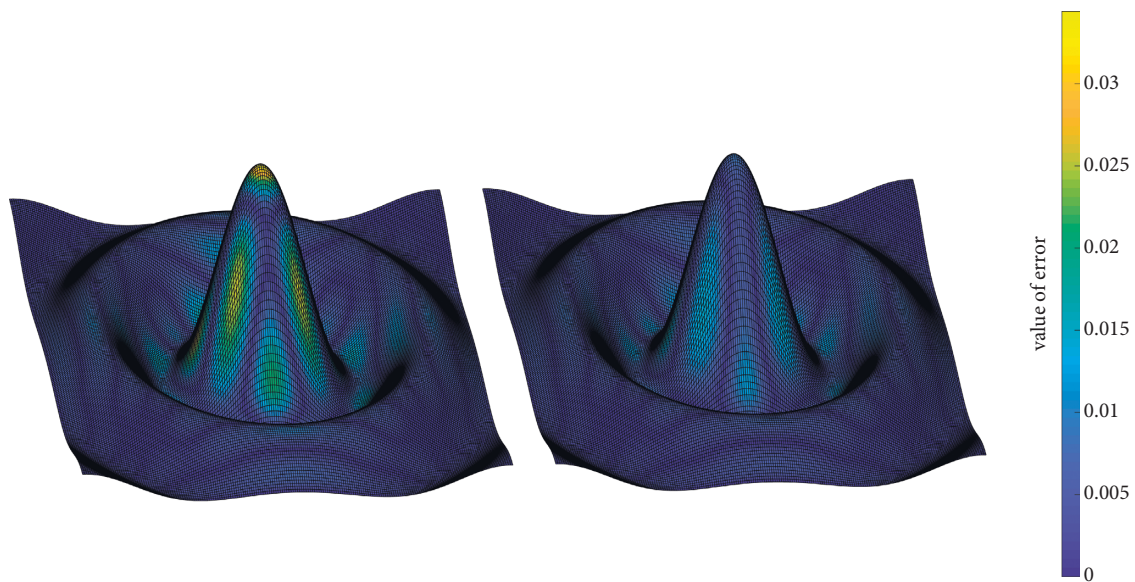


FIGURE 11: Fitting surfaces of “sine surface.”

As indicated by Figure 10(b), the proposed method is time-consuming compared with other knot vector optimization methods. The reason is the implementation of data parameter optimization and the calculation of partial derivatives. So, the effect and efficiency of the proposed method are unsatisfactory when the number of knots is large. However, the proposed method can achieve higher fitting accuracy and run within a finite time when the knots are small. In other words, the proposed method is more suitable to fit the small-scale data points in a small number of knots.

4.4. Application in Skinned Surface Fitting. The optimizations of data parameters and NURBS curve weights are suitable to fit the skinned surface in which the knot vector is constant. So, the proposed curve fitting method can be applied in the skinned surface fitting. First, group the data points sampled from the surface by row. Next, fit each row by a fixed knot vector and optimize the curve by the proposed method. The knots can be set uniformly.

Then, replace the data points with rows of control points. Because of the weights' optimization, the control points are in four dimensions. Fit them by column in a fixed knot vector to get the control points of the surface. Finally, combine the surface control points and the knot vector in u and v directions to get the NURBS surface. The pseudocode of fitting skinned surface is listed in Algorithm 2.

To verify the effectiveness of the proposed method, the “sine surface” is taken to be an example. The equation of surface is $z = \sin(r)/r$, where the r is the distance between the point and axis Z . The x and y are range from -10 to 10 . The 201×201 points are sampled from the surface. The 16×16 knots bicubic NURBS surface is fitted. The surface under optimization (right) and the surface without optimization (left) are shown in Figure 11. The value of error is represented by color. Without increasing or changing the knots, the proposed method reduces the maximum error from 0.01544 to 0.0066 , which is a reduction of 57.2% . The RMS error is reduced from 0.0066 to 0.0045 , which is a reduction of 31.8% .

5. Conclusion

This paper proposes a data parameters optimization-based method for optimizing the weights of NURBS fitting curve. By considering the geometric property of weight, the alteration values of the weights are calculated with the linearization assumption and the least square method. The process of fitting data points is provided and the effectiveness is demonstrated by numerical experiments. The experimental results show that the proposed method has better fitting accuracy and data noise robustness than other iterative optimization methods. Meanwhile, compared with the knot vector optimization method, the proposed method can obtain the fitting curve at small-scale knots with higher accuracy. Considering the merits of good data noise robustness and high fitting accuracy, the proposed method is suitable for applications such as fitting noisy data and generating skinned surfaces.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (52075069, 52005079) and the Fundamental Research Funds for the Central Universities (DUT21RC(3)069).

References

- [1] L. A. Piegl and W. Tiller, *The NURBS Book*, The NURBS Book, Springer, Berlin, Germany, 1997.
- [2] H. L. H. Bao and G. Wang, "Totally positive bases and progressive iteration approximation," *Computers & Mathematics with Applications*, vol. 50, no. 3-4, pp. 575–586, 2005.
- [3] C. Deng and H. Lin, "Progressive and iterative approximation for least squares B-spline curve and surface fitting," *Computer-Aided Design*, vol. 47, pp. 32–44, 2014.
- [4] W. Li, S. Xu, G. Zhao, and P. G. Li, "Adaptive knot placement in B-spline curve approximation," *Computer-Aided Design*, vol. 37, no. 8, pp. 791–797, 2005.
- [5] H. Park and J.-H. Lee, "-spline curve fitting based on adaptive curve refinement using dominant points," *Computer-Aided Design*, vol. 39, no. 6, pp. 439–451, 2007.
- [6] F. Liang, J. Zhao, S. Ji, C. Fan, and B. Zhang, "A novel knot selection method for the error-bounded B-spline curve fitting of sampling points in the measuring process," *Measurement Science and Technology*, vol. 28, no. 6, Article ID 65015, 2017.
- [7] E. Aguilar, H. Elizalde, D. Cárdenas, O. Probst, P. Marzocca, and R. A. Ramirez-Mendoza, "An adaptive curvature-guided approach for the knot-placement problem in fitted splines," *Journal of Computing and Information Science in Engineering*, vol. 18, no. 4, Article ID 41011, 2018.
- [8] R. Yeh, Y. S. G. Nashed, T. Peterka, and X. Tricoche, "Fast automatic knot placement method for accurate B-spline curve fitting," *Computer-Aided Design*, vol. 128, Article ID 102905, 2020.
- [9] P. LaubeM, O. Franz, and G. Umlauf, "Learnt knot placement in B-spline curve approximation using support vector machines," *Computer Aided Geometric Design*, vol. 62, Article ID S886307247, 2018.
- [10] D. V. T. T. Tegoeh and W. Rongling, "A direct method to solve optimal knots of B-spline curves: an application for non-uniform B-spline curves fitting," *PLoS One*, vol. 12, no. 3, Article ID e173857, 2017.
- [11] X. Zhao, C. Zhang, B. Yang, and P. Li, "Adaptive knot placement using a GMM-based continuous optimization algorithm in B-spline curve approximation," *Computer-Aided Design*, vol. 43, no. 6, pp. 598–604, 2011.
- [12] K. Uyar and E. Ülker, "B-spline curve fitting with invasive weed optimization," *Applied Mathematical Modelling*, vol. 52, no. dec, pp. 320–340, 2017.
- [13] A. Gálvez and A. Iglesias, "Particle swarm optimization for non-uniform rational B-spline surface reconstruction from clouds of 3D data points," *Information Science*, vol. 192, no. 6, pp. 174–192, 2012.
- [14] E. Ülker and A. Arslan, "Automatic knot adjustment using an artificial immune system for B-spline curve approximation," *Information Science*, vol. 179, no. 10, pp. 1483–1494, 2009.
- [15] G. Akemi, Andrés, and Iglesias, "Efficient particle swarm optimization approach for data fitting with free knot -splines," *Computer-Aided Design*, vol. 43, 2011.
- [16] O. V. B. Delgado-Marquez and M. Pasadas, "Evolutionary computation for optimal knots allocation in smoothing splines," *Applied Mathematical Modelling*, vol. 37, no. 8, pp. 5851–5863, 2013.
- [17] A. Gálvez and A. Iglesias, "A new iterative mutually coupled hybrid GA-PSO approach for curve fitting in manufacturing," *Applied Soft Computing Journal*, vol. 13, no. 3, pp. 1491–1504, 2013.
- [18] A. Gálvez, A. Iglesias, A. Avila, C. Otero, and C. Machado, "Elitist clonal selection algorithm for optimal choice of free knots in B-spline data fitting," *Applied Soft Computing*, vol. 26, 2015.
- [19] W. Ma and J. Kruth, "Parameterization of randomly measured points for least squares fitting of B-spline curves and surfaces," *Computer-Aided Design*, vol. 27, no. 9, pp. 663–675, 1995.
- [20] W. Ma and J.-P. Kruth, "NURBS curve and surface fitting for reverse engineering," *International Journal of Advanced Manufacturing Technology*, vol. 14, no. 12, pp. 918–927, 1998.
- [21] J. Z. S. Feng and H. Cui, "Optimized NURBS curve and surface fitting using simulated annealing," in *Proceedings of the International Symposium on Computational Intelligence & Design*, Changsha, China, December 2009.
- [22] P. Pandithevan and G. S. Kumar, "Optimal allocation of weights for NURBS curve fitting," in *Proceedings of the International Conference on Computer Aided Engineering (CAE 2007)*, Madras, India, December 2007.
- [23] Z. M. L. Yajuan and D. Chongyang, "Optimizing NURBS curves fitting by least squares progressive and iterative approximation," *Journal of Computer-Aided Design & Computer Graphics*, vol. 32, no. 04, pp. 59–65, 2020.
- [24] L. A. Piegl and W. Tiller, "Surface approximation to scanned data," *The Visual Computer*, vol. 16, no. 7, pp. 386–395, 2000.
- [25] PerChristian and Hansen, "The truncatedSVD as a method for regularization," *Bit Numerical Mathematics*, vol. 27, 1987.