Journal of Zhejiang University SCIENCE A
ISSN 1009-3095 (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Generalized fairing algorithm of parametric cubic splines

WANG Yuan-jun, CAO Yuan[‡]

(*School of Mathematical Science, Fudan University, Shanghai 200433, China*)

E-mail: wyj803@yahoo.com.cn; caoy@fudan.edu.cn

Received May 20, 2006; revision accepted June 21, 2006

Abstract: Kjellander has reported an algorithm for fairing uniform parametric cubic splines. Poliakoff extended Kjellander's algorithm to non-uniform case. However, they merely changed the bad point's position, and neglected the smoothing of tangent at bad point. In this paper, we present a fairing algorithm that both changed point's position and its corresponding tangent vector. The new algorithm possesses the minimum property of energy. We also proved Poliakoff's fairing algorithm is a deduction of our fairing algorithm. Several fairing examples are given in this paper.

Key words: Curve fairing, Tangent vector, Energy optimization, Cubic splines

doi:10.1631/jzus.2006.A1572 **Document code:** A **CLC number:** TP391

INTRODUCTION

In Computer Aided Design, fairness is one of the basic requirements for designing curves. If the curves are not fair, the design will be unsatisfactory and the producing process will be discommodious. In practice, there are all kinds of errors such as modelling error and measuring error. It means that the information on the data points may not be accurate. In order to obtain fair curves by interpolating these data points, we should fair the data points first. This is the source of curve fairing.

There are researches in this field (Kjellander, 1983; Poliakoff, 1996; Wang *et al*., 1997; Lee, 1990; Farin and Sapides, 1989; Li *et al*., 2004). The most fundamental results are probably due to Kjellander and Poliakoff.

Kjellander (1983) presented the fairing method of uniform parametric cubic splines. Poliakoff (1996) extended Kjellander's method to non-uniform form. Their algorithms have extensive application value. However, they only modified the bad point's position and did not consider the other information. Sometimes it can cause the fairing algorithm to be invalid. In this work, we established a new algorithm that not only modifies the position but also the corresponding tangent vector at bad point.

FAIRING OF CUBIC SPLINES

Kjellander's fairing algorithm

For uniform parametric cubic splines, Kjellander expounded his fairing method as follows. Suppose n is some integer in which a uniform parametric cubic spline passes through data points $P_1(\boldsymbol{r}_1, \boldsymbol{r}_1')$ to $P_n(\boldsymbol{r}_n, \boldsymbol{r}_n')$. Then it is assumed that the curve needs to be faired at the data point $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$ for some k ($1<k<n$) and a new position for $\boldsymbol{r}_k$ is obtained by finding an expression for the change in the third derivative at $\boldsymbol{r}_k$ in terms of all the $\boldsymbol{r}_i$ and $\boldsymbol{r}_i'$, where $1\le i\le n$ and all derivatives are taken with respect to the parameter t. By equating the change in the third derivative to zero and assuming that all other data points and all other first derivatives are unchanged, a new position, $\boldsymbol{r}_{k\text{new}}$, for $\boldsymbol{r}_k$ is calculated as follows.

Let parameter t take the value t_i at each

[‡] Corresponding author

[*] Project (No. 10371026) supported by the National Natural Science Foundation of China

$P_i(\boldsymbol{r}_i, \boldsymbol{r}_i')$, so that t changes from t_i to t_{i+1} over the ith segment, and let $\boldsymbol{a}_{i,j}$ indicate the coefficients of the ith segment. So a point on the ith segment of the cubic spline can be rewritten as

$$\boldsymbol{r}(t) = \sum_{j=0}^{3} \boldsymbol{a}_{i,j}(t - t_i)^j \text{ for } t_i \le t \le t_{i+1}, \quad (1)$$

where $t_{i+1}-t_i=1$, because the parametrization is uniform. If the end conditions are $\boldsymbol{r}(t_i)=\boldsymbol{r}_i$, $\boldsymbol{r}'(t_i)=\boldsymbol{r}_i'$, $\boldsymbol{r}(t_{i+1})=\boldsymbol{r}_{i+1}$ and $\boldsymbol{r}'(t_{i+1})=\boldsymbol{r}_{i+1}'$, then using $t_{i+1}-t_i=1$, the $\boldsymbol{a}_{i,j}$ are given by

$$\begin{aligned} \boldsymbol{a}_{i,0} &= \boldsymbol{r}_i,\ \boldsymbol{a}_{i,1} = \boldsymbol{r}_i', \\ \boldsymbol{a}_{i,2} &= 3(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i) - 2\boldsymbol{r}_i' - \boldsymbol{r}_{i+1}', \\ \boldsymbol{a}_{i,3} &= 2(\boldsymbol{r}_i - \boldsymbol{r}_{i+1}) + \boldsymbol{r}_i' + \boldsymbol{r}_{i+1}'. \end{aligned} \quad (2)$$

The change in the third derivative at $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$ is given by

$$\begin{aligned} \boldsymbol{r}'''(t_k^+) - \boldsymbol{r}'''(t_k^-) &= 6(\boldsymbol{a}_{k,3} - \boldsymbol{a}_{k-1,3}) \\ &= 24\left(\boldsymbol{r}_k - \frac{\boldsymbol{r}_{k-1} + \boldsymbol{r}_{k+1}}{2} - \frac{\boldsymbol{r}_{k-1}' - \boldsymbol{r}_{k+1}'}{4}\right). \end{aligned} \quad (3)$$

Changing Eq.(3) to zero, we get

$$\boldsymbol{r}_{k\text{new}} = \frac{\boldsymbol{r}_{k-1} + \boldsymbol{r}_{k+1}}{2} + \frac{\boldsymbol{r}_{k-1}' - \boldsymbol{r}_{k+1}'}{4}. \quad (4)$$

This is Kjellander's algorithm. But it cannot be used for non-uniformly parametrized cubic splines, because one of the assumptions that $t_{i+1}-t_i=1$ for each i is not true for non-uniform parameterization.

Poliakoff's fairing algorithm

In order to solve the above problem, Poliakoff extended Kjellander's algorithm to non-uniformly parametrized splines. Using the same notation from the previous subsection, for some integer n, there is a non-uniform parametric cubic spline curve passing through data points $P_1(\boldsymbol{r}_1, \boldsymbol{r}_1')$ to $P_n(\boldsymbol{r}_n, \boldsymbol{r}_n')$ and that again the curve needs to be faired at $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$. In addition, for the non-uniform case, we now suppose that the increase in the parameter t, between $P_i(\boldsymbol{r}_i, \boldsymbol{r}_i')$ and $P_{i+1}(\boldsymbol{r}_{i+1}, \boldsymbol{r}_{i+1}')$ is Δ_i (with all $\Delta_i>0$). As before, we find an expression for the change in the third derivatives at $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$ in terms of all the $\boldsymbol{r}_i$ and all the first derivatives $\boldsymbol{r}_i'$. Then we equate the change in the third derivative to zero and obtain the new position $\boldsymbol{r}_{k\text{new}}$.

Again, a point on the ith segment of the cubic spline is given by

$$\boldsymbol{r}(t) = \sum_{j=0}^{3} \boldsymbol{a}_{i,j}(t - t_i)^j \text{ for } t_i \le t \le t_{i+1},\ t_{i+1} - t_i = \Delta_i > 0. \quad (5)$$

Since the end conditions are $\boldsymbol{r}(t_i)=\boldsymbol{r}_i$, $\boldsymbol{r}'(t_i)=\boldsymbol{r}_i'$, $\boldsymbol{r}(t_{i+1})=\boldsymbol{r}_{i+1}$ and $\boldsymbol{r}'(t_{i+1})=\boldsymbol{r}_{i+1}'$, so the $\boldsymbol{a}_{i,j}$ are now given by

$$\begin{aligned} \boldsymbol{a}_{i,0} &= \boldsymbol{r}_i,\ \boldsymbol{a}_{i,1} = \boldsymbol{r}_i', \\ \boldsymbol{a}_{i,2} &= 3\frac{\boldsymbol{r}_{i+1} - \boldsymbol{r}_i}{\Delta_i^2} - \frac{2\boldsymbol{r}_i' + \boldsymbol{r}_{i+1}'}{\Delta_i}, \\ \boldsymbol{a}_{i,3} &= 2\frac{\boldsymbol{r}_i - \boldsymbol{r}_{i+1}}{\Delta_i^3} + \frac{\boldsymbol{r}_i' + \boldsymbol{r}_{i+1}'}{\Delta_i^2}. \end{aligned} \quad (6)$$

The change in the third derivative at $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$ is given by

$$\begin{aligned} \boldsymbol{r}'''(t_k^+) - \boldsymbol{r}'''(t_k^-) &= 6(\boldsymbol{a}_{k,3} - \boldsymbol{a}_{k-1,3}) \\ &= 12\left(\frac{1}{\Delta_{k-1}^3} + \frac{1}{\Delta_k^3}\right)(\boldsymbol{r}_k - \boldsymbol{r}_{k\text{new}}), \end{aligned} \quad (7)$$

where

$$\boldsymbol{r}_{k\text{new}} = \frac{\dfrac{\boldsymbol{r}_{k-1}}{\Delta_{k-1}^3} + \dfrac{\boldsymbol{r}_{k+1}}{\Delta_k^3} + \dfrac{1}{2}\left(\dfrac{\boldsymbol{r}_{k-1}'}{\Delta_{k-1}^2} - \dfrac{\boldsymbol{r}_{k+1}'}{\Delta_k^2}\right) + \dfrac{\boldsymbol{r}_k'}{2}\left(\dfrac{1}{\Delta_{k-1}^2} - \dfrac{1}{\Delta_k^2}\right)}{\dfrac{1}{\Delta_{k-1}^3} + \dfrac{1}{\Delta_k^3}}. \quad (8)$$

This is Poliakoff's algorithm. It is easy to verify that Eq.(8) reduces to Eq.(4) when all $\Delta_i=1$.

THE IMPROVED FAIRING ALGORITHM

Main idea

When we collect the data points' information from the design model, it is easier for us to obtain accurate point position than to obtain the points' tangent vector. Therefore, if the data point position is inaccurate, its tangent vector may not be accurate also. So, not only the bad point's position but also its tangent vector should be modified. From Eq.(8), we can find that the expression of $\boldsymbol{r}_{k\text{new}}$ including item $\boldsymbol{r}_k'$. If $\boldsymbol{r}_k'$ is not accurate, $\boldsymbol{r}_{k\text{new}}$ will also be bad. From this point, we think that we should correct the above algorithms. Similar to above algorithms, we assume that $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$ is the only point which needs to be faired, and a curve's internal strain energy can be approximated by $W=C\int(\boldsymbol{r}''(t))^2\mathrm{d}t$, where $C=(EI)^2/2$, and EI is the rigidity coefficient of splines. In general, if a point $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$ is bad, the curve's energy will increase. This is equivalent to the change of $\boldsymbol{r}''(t)$ at $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$ being bigger, that is to say, $|\boldsymbol{r}'''(t_k^+)-\boldsymbol{r}'''(t_k^-)|$ is big. By contrast, Kjellander and Poliakoff let the change of $|\boldsymbol{r}'''(t_k^+)-\boldsymbol{r}'''(t_k^-)|$ approach to zero and got the expression of $\boldsymbol{r}_{k\text{new}}$. We can calculate the changes at $P_{k-1}(\boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1}')$ and $P_{k+1}(\boldsymbol{r}_{k+1}, \boldsymbol{r}_{k+1}')$ similarly,

$$\begin{aligned}\frac{1}{6}(\boldsymbol{r}'''(t_{k-1}{}^{+})-\boldsymbol{r}'''(t_{k-1}{}^{-})) &= \boldsymbol{a}_{k-1,3}-\boldsymbol{a}_{k-2,3}\\ &= 2\left(\frac{1}{\Delta_{k-1}^3}+\frac{1}{\Delta_{k-2}^3}\right)\boldsymbol{r}_{k-1}-2\left(\frac{\boldsymbol{r}_k}{\Delta_{k-1}^3}+\frac{\boldsymbol{r}_{k-2}}{\Delta_{k-2}^3}\right)\\ &\quad+\left(\frac{\boldsymbol{r}_k'}{\Delta_{k-1}^2}+\frac{\boldsymbol{r}_{k-2}'}{\Delta_{k-2}^2}\right)+\boldsymbol{r}_{k-1}'\left(\frac{1}{\Delta_{k-1}^2}+\frac{1}{\Delta_{k-2}^2}\right),\end{aligned} \tag{9}$$

and

$$\begin{aligned}\frac{1}{6}(\boldsymbol{r}'''(t_{k+1}{}^{+})-\boldsymbol{r}'''(t_{k+1}{}^{-})) &= \boldsymbol{a}_{k+1,3}-\boldsymbol{a}_{k,3}\\ &= 2\left(\frac{1}{\Delta_{k+1}^3}+\frac{1}{\Delta_{k}^3}\right)\boldsymbol{r}_{k+1}-2\left(\frac{\boldsymbol{r}_{k+2}}{\Delta_{k+1}^3}+\frac{\boldsymbol{r}_{k}}{\Delta_{k}^3}\right)\\ &\quad+\left(\frac{\boldsymbol{r}_{k+2}'}{\Delta_{k+1}^2}+\frac{\boldsymbol{r}_{k}'}{\Delta_{k}^2}\right)+\boldsymbol{r}_{k+1}'\left(\frac{1}{\Delta_{k+1}^2}+\frac{1}{\Delta_{k}^2}\right).\end{aligned} \tag{10}$$

It is easy to see that both of the above equations include items $\boldsymbol{r}_k$ and $\boldsymbol{r}_k'$, so the bad point $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$ not only affects the curve's shape at $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$, but also affects the curve's shape near $P_{k-1}(\boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1}')$ and $P_{k+1}(\boldsymbol{r}_{k+1}, \boldsymbol{r}_{k+1}')$. This enlightened us considering the curve near points $P_{k-1}(\boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1}')$, $P_k(\boldsymbol{r}_k, \boldsymbol{r}_k')$ and $P_{k+1}(\boldsymbol{r}_{k+1}, \boldsymbol{r}_{k+1}')$, and give a better change at bad point.

The improved fairing algorithm

Based on the above ideas, we give an improved algorithm for non-uniform parametric cubic splines. For convenience, we give some notations first.

$\{P_i(t_i, \boldsymbol{r}_i, \boldsymbol{r}_i')\}_{i=1}^n$: The original data points; $\{P_{i\text{new}}(t_i, \boldsymbol{r}_{i\text{new}}, \boldsymbol{r}_{i\text{new}}')\}_{i=1}^n$: The modified data points; $\boldsymbol{r}(t)$: The spline curve interpolating original data points; $\boldsymbol{r}_{\text{new}}(t)$: The spline curve interpolating modified data points; $W = C\int_{t_1}^{t_n}(\boldsymbol{r}''(t))^2\,\mathrm{d}t$: The total strain energy of curve $\boldsymbol{r}(t)$; $W^* = C\int_{t_1}^{t_n}(\boldsymbol{r}_{\text{new}}''(t))^2\,\mathrm{d}t$: The total strain energy of curve $\boldsymbol{r}_{\text{new}}(t)$; $W_i = C\int_{t_1}^{t_{n+1}}(\boldsymbol{r}''(t))^2\,\mathrm{d}t$: The strain energy of the ith segment of curve $\boldsymbol{r}(t)$ (i=1, …, n−1); $W_i^* = C\int_{t_1}^{t_{n+1}}(\boldsymbol{r}_{\text{new}}''(t))^2\,\mathrm{d}t$: The strain energy of the ith segment of curve $\boldsymbol{r}_{\text{new}}(t)$ (i=1, …, n−1); $\delta(\boldsymbol{r})$: The change to the position of the spline; $\eta(\boldsymbol{r}')$: The change to the first derivative of the spline; $\delta^*(\boldsymbol{r})$:The optimization change to the position of the spline; $\eta^*(\boldsymbol{r}')$:The optimization change to the first derivative of the spline.

Based on the same criterion, we suppose that the smaller the curve's strain energy, the fairer the curve. Now the new fairing algorithm is as follows

$$\begin{cases}\boldsymbol{r}_{k\text{new}} = \boldsymbol{r}_k + \delta^*(\boldsymbol{r}_k),\\ \boldsymbol{r}_{k\text{new}}' = \boldsymbol{r}_k' + \eta^*(\boldsymbol{r}_k'),\end{cases} \tag{11}$$

where

$$\begin{cases}\delta^*(\boldsymbol{r}_k)=\dfrac{\Delta_k^2\Delta_{k-1}^2}{(\Delta_k+\Delta_{k-1})^3}[2\Delta_k\Delta_{k-1}(\boldsymbol{a}_{k-1,3}-\boldsymbol{a}_{k,3})\\ \qquad\qquad+(\Delta_k-\Delta_{k-1})(\boldsymbol{a}_{k,2}-\boldsymbol{a}_{k-1,2}-3\boldsymbol{a}_{k-1,3}\Delta_{k-1})],\\ \eta^*(\boldsymbol{r}_k)=\dfrac{\Delta_k\Delta_{k-1}}{(\Delta_k+\Delta_{k-1})^3}[3\Delta_k\Delta_{k-1}(\Delta_k-\Delta_{k-1})(\boldsymbol{a}_{k-1,3}-\boldsymbol{a}_{k,3})\\ \qquad\qquad+2(\Delta_k^2-\Delta_k\Delta_{k-1}+\Delta_{k-1}^2)(\boldsymbol{a}_{k,2}-\boldsymbol{a}_{k-1,2}-3\boldsymbol{a}_{k-1,3}\Delta_{k-1})],\end{cases} \tag{12}$$

and $\boldsymbol{a}_{i,2}$, $\boldsymbol{a}_{i,3}$ (i=k−1, k) are the coefficients of the non-uniform spline.

Algorithm:

Step 1: Interpolate the primitive points using non-uniform cubic spline with natural end conditions;

Step 2: Find the worst bad points by the curvature plot of the spline;

Step 3: Fair the point by changing its position to a new one;

Step 4: Interpolate the points again;

Step 5: Repeat Step 2~Step 4;

Step 6: Stop until all the points are good.

MATHEMATICAL JUSTIFICATION

In order to allow the integral of the square of the 2nd derivative with respect to the parameter to be a good approximation to the strain energy, Kjellander assumed that $|\boldsymbol{r}'|$ is close to a constant along the curve. Poliakoff made the same assumption. We also make the assumption in our fairing algorithm. Then, we have a Lemma as follows:

Lemma 1 Changing $P_k(t_k, \boldsymbol{r}_k, \boldsymbol{r}'_k)$ to $P_{k\text{new}}(t_k, \boldsymbol{r}_{k\text{new}}, \boldsymbol{r}'_{k\text{new}})$, with other points unchanged, the change of the spline curve's strain energy is

$$W^* - W = 4CF(\delta(\boldsymbol{r}_k), \eta(\boldsymbol{r}'_k)), \tag{13}$$

where

$$\begin{aligned} &F(\delta(\boldsymbol{r}_k), \eta(\boldsymbol{r}'_k)) \\ &= 3(\boldsymbol{a}_{k,3} - \boldsymbol{a}_{k-1,3}) \cdot \delta(\boldsymbol{r}_k) + (-\boldsymbol{a}_{k,2} + \boldsymbol{a}_{k-1,2} \\ &\quad + 3\boldsymbol{a}_{k-1,3}\Delta_{k-1}) \cdot \eta(\boldsymbol{r}'_k) + \left[\left(\frac{3}{\Delta_k^3} + \frac{3}{\Delta_{k-1}^3}\right)\delta^2(\boldsymbol{r}_k)\right. \\ &\quad \left. + \left(\frac{1}{\Delta_k} + \frac{1}{\Delta_{k-1}}\right)\eta^2(\boldsymbol{r}'_k) + \left(\frac{3}{\Delta_k^2} - \frac{3}{\Delta_{k-1}^2}\right)\delta(\boldsymbol{r}_k) \cdot \eta(\boldsymbol{r}'_k)\right], \end{aligned} \tag{14}$$

"·" means inner product.

From Lemma 1, we can prove the following Theorem:

Theorem 1 Let a non-uniform parametric cubic spline through points $\{P_i(t_i, \boldsymbol{r}_i, \boldsymbol{r}'_i)\}_{i=1}^n$ and $\Delta_i = t_{i+1} - t_i > 0$ (i=1, …, n−1). Suppose that $P_k(t_k, \boldsymbol{r}_k, \boldsymbol{r}'_k)$ has been changed to $P_{k\text{new}}(t_k, \boldsymbol{r}_{k\text{new}}, \boldsymbol{r}'_{k\text{new}})$ by using Eqs.(11) and (12). Assume that a new parametric cubic spline curve interpolates the new points. Then the new curve's strain energy is no greater than the original one and it has the property of optimal energy.

Corollary 1 Especially, taking $\eta(\boldsymbol{r}'_k) = 0$ in Eq.(14), namely $\boldsymbol{r}'_k$ is unchanged. We get

$$\begin{aligned} &W^* - W \\ &= 4C\left\{3(\boldsymbol{a}_{k,3} - \boldsymbol{a}_{k-1,3}) \cdot \delta(\boldsymbol{r}_k) + \left(\frac{3}{\Delta_k^3} + \frac{3}{\Delta_{k-1}^3}\right)\delta^2(\boldsymbol{r}_k)\right\}. \end{aligned}$$

If it is minimized, we have

$$\delta^*(\boldsymbol{r}_k) = -\frac{\boldsymbol{a}_{k,3} - \boldsymbol{a}_{k-1,3}}{2/\Delta_k^3 + 2/\Delta_{k-1}^3},$$

when introduced into Eq.(11) gives the same result Eq.(8) as in Polakoff's algorithm.

Hence, Poliakoff's algorithm is energy optimal when $\boldsymbol{r}'_k$ is unchanged. It means we really extended Poliakoff's algorithm.

EXAMPLES AND DISCUSSIONS

Bad points identification

In order to fair the spline curve, we should find the bad data points first. In this paper, we take the change in the third derivative at the data point as a rule. The point that has the biggest change of the third derivative is the worst. Li *et al*.(2004) think that monotone curvature is preferred and the curvature plot should change gradually at the knot. In fact, our rule is near to the rule that the curvature plot should change gradually at the knot. Differently, Li *et al*.(2004) paid attention to the monotony, extremum, inflection points and flat points of the curvature plot. We pay more attention to the changes of the bad point and its adjacent points.

It is easy to deal with the case that only one bad point exists, so we apply our fairing algorithm to this case. If there are several separated bad points (the changes of the third derivatives at these points are all big), we can fair them one by one.

Fairing example

In this section, we give two examples to illustrate the validness. In our examples, we take chord length parametrization and give the natural boundary conditions at both ends.

Example 1 We took seven standard points: $\{(i,1)\}_{i=1}^7$, and moved the 4th point (4, 1) to (4.3, 1.6).

The cubic interpolating spline oscillated (Fig.1a) and the curvature curve showed where inflection appears (Fig.1b). Fig.1c is 4th point faired according to Poliakpff's algorithm and is interpolated again. Fig.1d is the curvature curve corresponding to Fig.1c. Fig.1e and Fig.1f are similar work done by new fairing algorithm. All numerical values are listed in Table 1. It is easy to see the difference between the two algorithms.

Example 2 We took twelve scattered data points. Fig.2a is cubic spline interpolated by these points; and Fig.2b is its curvature curve. Fig.2c and Fig.2d are faired seven times by Poliakoff's algorithm. Fig.2e and Fig.2f are faired seven times by the new algorithm. The numerical values are shown in Table 2.

Table 1 The data of Example 1

Primitive points	Faired by Poliakoff's algorithm	Faired by our algorithm
P_1 (1,1)	P_1 (1,1)	P_1 (1,1)
P_2 (2,1)	P_2 (2,1)	P_2 (2,1)
P_3 (3,1)	P_3 (3,1)	P_3 (3,1)
P_4 (4.3,1.6)	P_4 (4.7627,1.0195)	P_4 (4.2942,1.0193)
P_5 (5,1)	P_5 (5,1)	P_5 (5,1)
P_6 (6,1)	P_6 (6,1)	P_6 (6,1)
P_7 (7,1)	P_7 (7,1)	P_7 (7,1)

Discussions

Kjellander and Poliakoff supposed that the first derivative is close to a constant, so the strain energy formula they adopted is appropriate to the real strain energy of the spline. Though we give the modification

Table 2 The data of Example 2

Primitive points	Faired seven times by Poliakoff's algorithm	Faired seven times by our algorithm
P_1 (1,0.3000)	P_1 (1,0.3000)	P_1 (1,0.3000)
P_2 (2,0.4832)	P_2 (2.0309,0.4413)	P_2 (2.0096,0.4351)
P_3 (3,0.5943)	P_3 (2.9323,0.5511)	P_3 (3.0050,0.5514)
P_4 (4,0.6731)	P_4 (4.0157,0.6481)	P_4 (4.0020,0.6471)
P_5 (5,0.7538)	P_5 (4.9988,0.7221)	P_5 (5.0029,0.7199)
P_6 (6,0.7876)	P_6 (6,0.7876)	P_6 (6,0.7876)
P_7 (7,0.8319)	P_7 (7,0.8319)	P_7 (7,0.8319)
P_8 (8,0.8706)	P_8 (8,0.8706)	P_8 (8,0.8706)
P_9 (9,0.9050)	P_9 (9,0.9050)	P_9 (9,0.9050)
P_{10} (10,0.9360)	P_{10} (10,0.9360)	P_{10} (10,0.9360)
P_{11} (11,0.9642)	P_{11} (11,0.9642)	P_{11} (11,0.9642)
P_{12} (12,0.9901)	P_{12} (12,0.9901)	P_{12} (12,0.9901)

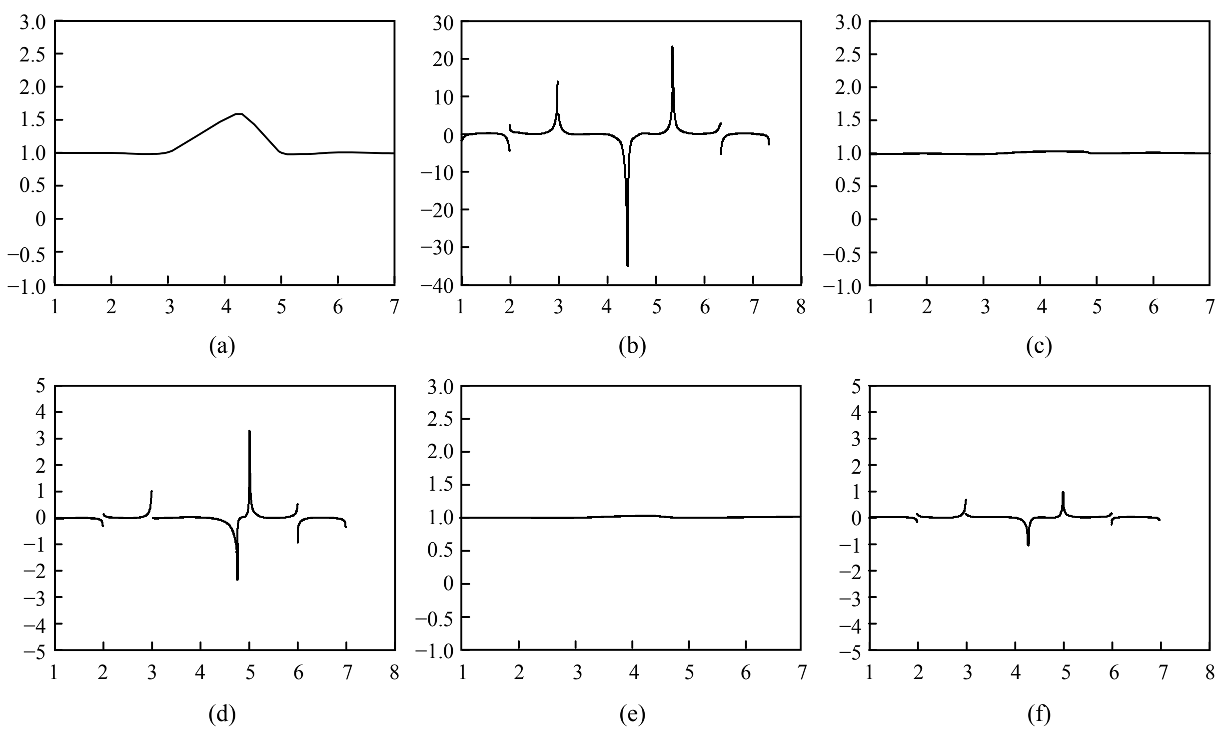

Fig.1 The figures of Example 1. (a) The cubic interpolating spline; (b) The curvature curve of (a); (c) The cubic interpolating spline faired by Poliakoff's algorithm; (d) The curvature curve of (c); (e) The cubic interpolating spline faired by our algorithm; (f) The curvature curve of (e)

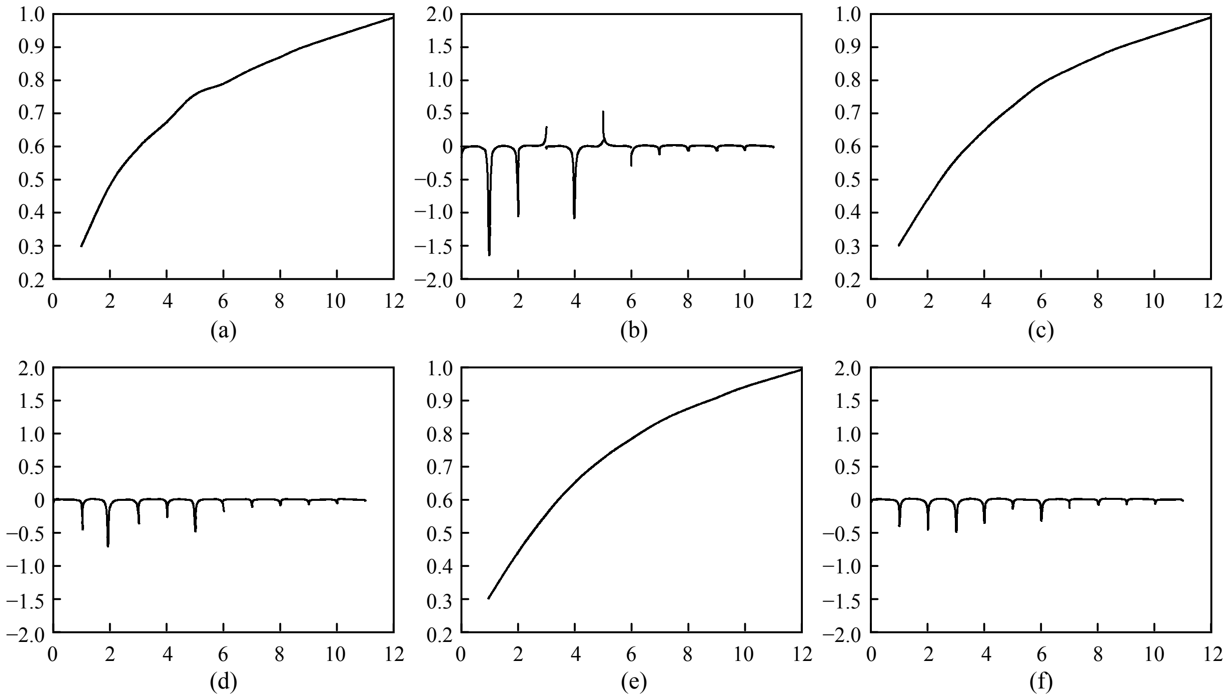

Fig.2 The figures of Example 2. (a) The cubic interpolating spline; (b) The curvature curve of (a); (c) The cubic interpolating spline faired seven times by Poliakoff's algorithm; (d) The curvature curve of (c); (e) The cubic interpolating spline faired seven times by our algorithm; (f) The curvature curve of (e)

formula of the first derivative, we do not use it in our fairing examples because it is not required in the process of interpolating a cubic spline. On the other hand, as the impact of the first derivative is not so notable as the position of the points in cubic parametric spline, so we only use the formula of the position change.

In fact, the first derivative at the bad data point is also abnormal, even at the neighboring points. In order to simplify the question, we only give the changing formula of the first derivative at the bad point. In other interpolation, such as Herimite interpolation, the formula of the derivative may be useful.

According to the rule for distinguishing the bad point, we should give a small positive number first. If the third derivative change is not greater than that given number, we will stop the fairing to the bad point.

CONCLUSION

In cubic parametric spline fairing field, both Kjellander and Poliakoff composed their algorithms based on the strain energy criterion. But they only changed the bad point's position without changing the corresponding tangent vector. In this paper, based on the same criterion, we give an improved fairing algorithm which not only changes the bad point's position but also changes its tangent vector. We also proved the new algorithm's property of energy optimization. Numerous examples showed that this algorithm is valid.

In future, we will extend this algorithm to other spline curves. A similar result of quintic spline curve is also obtained. We will recite this in another paper. What is more, as the application of surface is broader than curve, we will devote ourselves to the fairing algorithm of the surface.

References

Farin, G., Sapides, N., 1989. Curvature and the fairness of curves and surfaces. *IEEE Computer Graphics and Applications*, **9**(2):52-57. [doi:10.1109/38.19051]

Kjellander, J.A.P., 1983. Smoothing of cubic parametric splines. *Computer-Aided Design*, **15**(3):175-179. [doi:10.1016/0010-4485(83)90085-4]

Lee, E.T.Y., 1990. Energy, fairness, and a counterexample. *Computer-Aided Design*, **22**(1):37-40. [doi:10.1016/0010-4485(90)90027-A]

Li, W., Xu, S., Zheng, J., Zhao, G., 2004. Target curvature driven fairing algorithm for planar B-spline curves. *Computer Aided Geometric Design*, **21**:499-513.

Poliakoff, J.F., 1996. An improved algorithm for automatic fairing of non-uniform parametric cubic splines. *Computer-Aided Design*, **28**(1):59-66. [doi:10.1016/0010-4485(95)00048-8]

Wang, X., Cheng, F., Barsky, B.A., 1997. Energy and B-spline interproximation. *Computer-Aided Design*, **29**(7):485-496. [doi:10.1016/S0010-4485(96)00087-5]