Golden interpolation

Ying He*, Jincai Chang*b,

*a College of Information Engineering, North China University of Science and Technology, Tangshan 063210, China
*b College of Sciences, North China University of Science and Technology, Tangshan 063210, China

ARTICLE INFO

Article history:
Received September 13, 2018

Keywords:
Golden section
Aesthetic interpolation
Golden cuspidal hill
Golden domed hill

ABSTRACT

For the classic aesthetic interpolation problem, we propose an entirely new thought: apply the golden section. For how to apply the golden section to interpolation methods, we present three examples: the golden step interpolation, the golden piecewise linear interpolation and the golden curve interpolation, which respectively deal with the applications of golden section in the interpolation of degree 0, 1, and 2 in the plane. In each example, we present our basic ideas, the specific methods, comparative examples and applications, and relevant criteria. And it is worth mentioning that for aesthetics, we propose two novel concepts: the golden cuspidal hill and the golden domed hill. This paper aims to provide the reference for the combination of golden section and interpolation, and stimulate more and better related researches.

© 2018 Elsevier B.V. All rights reserved.

1. Introduction

The golden section has a long history: the earliest discovery can be traced back to the Greek Pythagoras School in the sixth century BC, and then be systematically studied by the Greek mathematician Eudoxus in the fourth century BC, and be given a complete geometry definition and proof in the Euclidean Geometry in the third century BC [1]. The golden section is mysterious and is found related to many aesthetic things in the nature [2], such as our body [3], the canopy structure of forage plants [4] and so on. In mathematics, people are also surprised to find the widespread existence of the golden section. For example, the literature [5] describes the discovery that super central configurations of the n-body problem have surprising connections with the golden section ratio. For another example, the literature [6] points out that the golden section exists in the spline function. Since its birth, golden section has been connected to beauty and has become a universally recognized aesthetic law [7]. Researches show that the aesthetic experience of golden section is not entirely subjective and there are an objective biological basis [8] and some psychological basis [9]. Though there has been no definite explanation for why the golden section is beautiful, it is found that it has repeatedly played an effective role in practice. Nowadays, the aesthetic value of golden section has been widely applied in the fields of art, nature, architecture, mathematics, philosophy, production practice and so on. The golden section ratio is consistently represented by the Greek letter \( \varphi \), and so is it in the following sections of this paper.

Interpolation is a classic problem. In the interpolation problem, it is required to construct a function that exactly matches prescribed data, such as a series of points and derivatives. Common interpolation methods include the nearest neighbor interpolation, the spline interpolation, the Lagrange interpolation, the Hermite interpolation and so on [10, 11, 12]. Nowadays interpolation methods are used not only for estimation, but also for computer-aided design, shape modeling, geometric processing and many other fields [13] in which the appearance of the interpolation function graph is of great importance. As a result, aesthetic interpolation forms a fundamental problem to explore a better appearance. There have been lots of researches struggling for it. In general, the existing methods are in ac-
cordance with the following thoughts: (1) Improve the fairness of the interpolation curve(surfce). This thought was in embryo when it was 1966 [14], and then it progressively developed [15] [16]. In 1994, Nowacki and Lü [17] went further and adopted a fairness criterion about it. Nowadays, minimizing some energy functions becomes a universally accepted approach to improve the fairness of the interpolation, in which the strain energy(also called bending energy) and curvature variation energy are mostly used, e.g. the literatures [18] [19] [20] [21].

(2) Interpolate with a log-aesthetic interpolation curve. This thought can date back to 1997 when Harada analyzed many aesthetic curves including feature curves of automobiles and then proposed the log-aesthetic curves [22]. Nowadays this thought has also been widely applied to enhance the beauty of interpolation or approximation function graphs, e.g. the literatures [23] [24] [25]. (3) If a curve has monotonous curvature, then it is deemed more pleasant. Spirals have monotonous curvature, and interpolating with a spiral gradually becomes a criterion for aesthetic interpolation. So there is lots of work studying the aesthetic interpolation according to this thought, e.g. the literatures [26] [27] [28].

As a famous aesthetic law, golden section has been widely used to enhance the beauty of things, but as far as we know, no one has applied the golden section principle to solve the aesthetic interpolation problem. The literature [29] studied the approximation of spirals by piecewise curves of fewest circular arcs, in which there existed an application of the golden section, but it was used for the extremal value finding in iterative computation rather than for changing the appearance of the curve. In this paper, we propose a novel thought for the fundamental aesthetic interpolation problem: apply the golden section principle to improve the beauty of the interpolation function graph. We call this kind of interpolation “golden interpolation”. In the following of this paper, when we use the adjective “golden” before a concept, we mean there is the adjective “golden” before a concept, we mean there is the adjective “golden” before a concept, we mean there is the adjective “golden” before a concept, we mean there is the adjective “golden” before a concept, we mean there is the adjective “golden” before a concept, we mean there is the adjective “golden” before a concept, we mean there is the adjective “golden” before a concept, we mean there is the adjective “golden” before a concept. And finally, in Section 7, we conclude this paper and propose some prospects. For the convenience of the computer implementation, we describe our methods in the form of algorithms by using pseudo codes and put them in Appendix B.

2. Control interpolation by adding a node transform

Since the geometric characteristics of many interpolation function graphs are clearly related to the interpolation data nodes, we can easily change the function graph into the shape we want by transforming the data nodes. So in this paper, we do interpolation following the two steps: first, transform the data nodes \( A_0, A_1, \ldots, A_n \) into \( B_0, B_1, \ldots, B_m \); second, construct a traditional interpolation function for \( B_0, B_1, \ldots, B_m \).

On considering the demand for this paper, we define two kinds of data node transforms:

**Definition 1.** If the number of data nodes \( A_0, A_1, \ldots, A_n \) is smaller than the number of data nodes \( B_0, B_1, \ldots, B_m \), then the transform is called an extension transform.

**Definition 2.** If the number of data nodes \( A_0, A_1, \ldots, A_n \) is equal to the number of data nodes \( B_0, B_1, \ldots, B_m \), then the transform is called an equal number transform.

3. Spline interpolation of degree 0, 1, and 2

We construct a golden interpolation function by adding a transform to the traditional method. In this paper, the following traditional methods have been used.

Give interpolation data nodes \( A_0(x_0, y_0), A_1(x_1, y_1), \ldots, A_n(x_n, y_n) \), where \( x_0 < x_1 < \cdots < x_n \), \( n \geq 1 \). And for quadratic spline interpolation, we also need to give the derivative \( k_0 \) at the node \( A_0 \).

(1) Zero-degree spline interpolation

The interpolation function is

\[
p(x) = y_i,
\]

where \( x \in [x_i, x_{i+1}], i = 0, 1, \ldots, n-1 \).

(2) Piecewise linear interpolation

The interpolation function is

\[
p(x) = \begin{cases} x - x_{i+1} & \text{if } x \in [x_i, x_{i+1}], i = 0, 1, \ldots, n-1, \\ \frac{x - x_i}{x_{i+1} - x_i} y_i + \frac{x - x_i}{x_{i+1} - x_i} y_{i+1}, & \text{otherwise} \end{cases}
\]

where \( x \in [x_i, x_{i+1}], i = 0, 1, \ldots, n-1 \). It is also the one-degree spline interpolation.

(3) Quadratic spline interpolation

The interpolation function is

\[
p(x) = \begin{cases} \left( \frac{x - x_{i+1}}{x_i - x_{i+1}} \right)^2 y_i + \left( \frac{x - x_i}{x_{i+1} - x_i} \right)^2 y_{i+1} + \left( \frac{x - x_i}{x_{i+1} - x_i} \right) \sum_{j=0}^{i-1} (-1)^{j+i} y_{j+1} - y_j \cdot \frac{2 y_j}{x_{i+1} - x_i} & \text{if } x \in [x_i, x_{i+1}], i = 0, 1, \ldots, n-1, \\ -k_0 x + \frac{2 y_j}{x_i - x_{i+1}} \end{cases}
\]

where \( x \in [x_i, x_{i+1}], i = 0, 1, \ldots, n-1 \). It is also the two-degree spline interpolation.
A quadratic spline is composed of piecewise quadratic polynomials with continuous first-order derivatives at spline knots. In order to write its interpolation function according to interpolation data nodes directly, we create the above formula referring to the thought of barycentric rational Hermite interpolation [30]. Here we elaborate how we deduce the formula above.

Consider the interval \([x_i, x_{i+1}]\). We all know the two basis functions:

\[
b_i(x) = \frac{x - x_{i+1}}{x_i - x_{i+1}}, \quad b_{i+1}(x) = \frac{x - x_i}{x_{i+1} - x_i}.
\]

They have the properties:

\[
b_i(x) = \begin{cases} 1, & x = x_i, \\ 0, & x = x_{i+1}, \\ 1, & x = x_{i+1}, \end{cases} \quad b_{i+1}(x) = \begin{cases} 0, & x = x_i, \\ 1, & x = x_{i+1}. \end{cases}
\]

It is well-known that the linear combination \(b_i(x)y_i + b_{i+1}(x)y_{i+1}\) is a one-degree polynomial that passes through \(A_i\) and \(A_{i+1}\). Then we can easily think out a two-degree polynomial that passes through \(A_i\) and \(A_{i+1}\):

\[
r_i(x) = b_i(x)^2y_i + b_{i+1}(x)^2y_{i+1}.
\]

But for the quadratic spline, we also need to restrict the derivative at \(x_i\), so that the first-order derivative of the whole function at \(x_i\) is continuous. Referring to the Hermite interpolation [30], we can think out the following expression:

\[
p_i(x) = r_i(x) + (x - x_i)b_i(x)[p_i'(x_i) - r_i'(x_i)].
\]

It’s easy to work out:

\[
p_i(x_i) = y_i,
\]

\[
p_i(x_{i+1}) = y_{i+1},
\]

\[
p_i'(x_i) = p_i'(x_i),
\]

\[
p_i'(x_{i+1}) = \frac{2(y_{i+1} - y_i)}{x_{i+1} - x_i} - p_i'(x_i).
\]

We mark \(q_i = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}\). Then \(p_i'(x_{i+1}) = 2q_i - p_i'(x_i)\),

\[
p_i'(x_{i+1}) = 2q_{i-1} - p_i'(x_{i-1})
\]

\[
= 2q_{i-1} - [2k_{i-2} - p_i'(x_{i-2})]
\]

\[
= 2q_{i-1} - 2q_{i-2} + 2q_{i-3} - \cdots + (-)^2q_0 - (+)k_0
\]

\[
= 2 \sum_{j=0}^{i-1} (-1)^{j+1}q_j + (-)^i k_0.
\]

Put all the equations into the equation (1) and we can get the expression of the quadratic spline interpolation function above.

4. Golden step interpolation

Among one-dimensional interpolation methods, the step function interpolation is the simplest and plainest. So this paper first discusses how to apply the golden section principle to the step function interpolation. The graph of the step interpolation function is a set of horizontal line segments that respectively pass through the given data nodes, and the lengths of the line segments are the most obvious geometric feature of the graph. So the basic idea of our golden step interpolation is trying to make the ratio of the lengths of the two adjacent segments satisfy the golden section ratio. Common the step function interpolation methods are various, like the nearest neighbor interpolation, the zero-degree spline interpolation, etc. Considering the zero-degree spline interpolation has the property that we can easily change the function graph into the shape we want by transforming the interpolation data nodes, we use this traditional method as a basis, and add a golden transform for the data nodes to enhance the beauty of the final interpolation graph.

4.1. Golden section step interpolation

We add an extension transform to the zero-degree spline interpolation, in which additional nodes are added to the original interpolation node set, and finally makes the ratio of the lengths of two line segments between two adjacent original interpolation nodes satisfy the golden section ratio. The specific interpolation process includes two steps as follows, and we call this method the golden extension step interpolation.

step 1. Transform \(A_0, A_1, \ldots, A_n\) into \(A_0, A_{0.5}, A_1, A_{1.5}, \ldots, A_{n-0.5}, A_n\) as follows:

Add a data node \(A_{i+0.5}(x_{i+0.5}, y_{i+0.5})\) between every two adjacent nodes \(A_i, A_{i+1}\) \((i = 0, 1, \ldots, n-1)\), in which

\[
x_{i+0.5} = x_{i+1} - (x_{i+1} - x_i)\phi,
\]

and

\[
y_{i+0.5} = \begin{cases} y_{i+1} - (y_{i+1} - y_i)\phi, & y_i \neq y_{i+1}, \\ y_{i+1} + L_i y_i = y_{i+1}, & L_i \neq 0 \end{cases}.
\]

\(L_i \neq 0\) is a longitudinal jump that is given \(A_{i+0.5}\) when \(y_i = y_{i+1}\) and it should be determined according to the actual situation.

step 2. Get the final interpolation function \(p_{se}(x) = y_{0.5}, \) where \(x \in [x_{0.5}, x_{0.5(i+1)}], i = 0, 1, \ldots, 2n - 1.\)

We need to explain that there are two golden section points in an interval, but here we take the left one. We can also take the right point by replacing the \(\varphi\) with 1 - \(\varphi\). So is it for other methods in this paper and we won’t explain it again.

4.2. Golden equal number step interpolation

The above golden extension step interpolation will extend data nodes. Although it can guarantee that the ratio of the lengths of two line segments between two adjacent interpolation data nodes satisfies the golden section ratio in the graph, compared with the direct zero-degree spline interpolation, this method increases the discontinuities of the function, and the number of line segments in the graph is doubled, and the length of each line segment is shortened. However, sometimes we want to make the function graph look more beautiful on condition that the interpolation function has the same number of discontinuities as the direct zero-degree spline interpolation function. Therefore, we also propose adding a golden equal number transform, which makes the ratio of the lengths of two adjacent line segments in the final interpolation function graph satisfy the golden section ratio as far as possible. The specific interpolation process includes two steps as follows, and we call this method the golden equal number step interpolation.
step 1. Transform \( A_0, A_1, \cdots, A_n \) into \( A_0, A'_1, A_2, A'_3, \cdots, A_n \) as follows:

If \( n < 2 \), the data nodes are not changed; that is, the transformed data nodes are the same as the original data nodes.

If \( n \geq 2 \), take out the data nodes \( A_{2i-1}(x_{2i-1}, y_{2i-1}), i = 1, 2, \cdots, \lceil 0.5n \rceil \) whose sequence numbers are odd numbers among the \( n \) data nodes, in which \( \lceil 0.5n \rceil \) represents the integer part of 0.5\( n \). Then transform the \( x \)-coordinate \( x_{2i-1} \) of \( A_{2i-1}(x_{2i-1}, y_{2i-1}) \) into

\[
x'_{2i-1} = \begin{cases} 
  x_i, & x_i < x_{2i-1}, \\
  x_{2i-1}, & x_i \geq x_{2i-1}, 
\end{cases}
\]

where \( x'_i = x_i - (x_i - x_{2i-2})\varphi \).

Then the data node \( A_{2i-1}(x_{2i-1}, y_{2i-1}) \) is transformed into \( A'_{2i-1}(x'_{2i-1}, y_{2i-1}) \). The meaning of this transform will be explained below.

step 2. For convenience, we mark \( A_0, A'_1, A_2, A'_3, \cdots, A_n \) as \( A'_0(x'_{0}, y_{0}), A'_1(x'_{1}, y_{1}), A'_2(x'_{2}, y_{2}), A'_3(x'_{3}, y_{3}), \cdots, A'_n(x'_{n}, y_{n}) \), and get the final interpolation function \( p_i(x) = y_i, \) where \( x \in [x'_{i}, x'_{i+1}], i = 0, 1, \cdots, n - 1 \).

It is necessary to explain the meaning of the transform in step 1. The meaning is that if we can make the ratio of the length of the line segment passing through \( A_{2i-2} \) to the length of the line segment passing through \( A_{2i-1} \) satisfy the golden section ratio in the function graph, then change the data node \( A_{2i-1} \), otherwise keep the data node \( A_{2i-1} \) unchanged. So this transform just try to be as far as possible and the transformed data nodes are not always different from the original data nodes.

4.3. Examples and applications

Give data nodes \( B_0(2, 1), B_1(9, 3), B_2(11, 4), B_3(19, 6), B_4(21, 2), B_5(24, 5) \).

Do the traditional zero-degree spline interpolation and the golden equal number step interpolation for them. The function graphs are Fig. 1(a) and Fig. 1(b). In order to exist a line segment passing through the final interpolation data node, we extend the definition domain of the function, such that for the line segment passing through \( B_3 \), its length equals the length of its previous adjacent line segment in Fig. 1(a) and is \( 1/\varphi \) times the length of its previous adjacent line segment in Fig. 1(b). Then we discuss an application of designing the stumps in the park: If we want to design some cylindrical stumps next to each other with diverse diameters and heights in the park for decorating, exercising and playing, the designer only needs to point some nodes on the panel to decide the approximate locations and heights of the stumps. We also suppose these nodes are the \( B_0, B_1, \cdots, B_5 \) above. Then the computer can automatically generate the line segments in Fig. 1(a) or Fig. 1(b). The side view of these stumps can be constructed by adding the perpendicular lines of the horizontal axis through the ends of these line segments. Rotate each side view of these stumps and the 3D model of the stumps can be built. See Fig. 1(d) and Fig. 1(c). By comparing, we can observe that the appearance generated by the golden method is more harmonious and aesthetic than the one generated by the traditional method. It is owed to the integration of gold ratio in geometric features in the golden method.

For the golden extension step interpolation, the function graph interpolating the above nodes is Fig. 2(a) in which the length of the final line segment is \( 1/\varphi \) times that of the previous one. For this golden method, we can use it to design a picture in which meteorites are streaking across the night sky: The designer only needs to point several nodes on the canvas. Let’s suppose these nodes are the \( B_0, B_1, \cdots, B_5 \) above. Then the computer can automatically generate the tracks of the meteorites passing through these nodes in the night sky. These tracks are the line segments in Fig. 2(a) and the generated picture is shown in Fig. 2(b). We can see the picture Fig. 2(b) looks harmonious and aesthetic.

4.4. Go further to explore criteria

For the aesthetic interpolation problem, most of the traditional thoughts have developed some criteria. For example, strain energy and curvature variation energy have been universally used to measure the fairness of a curve [18, 19, 20, 21], which can provide reference for new interpolation methods and turn the aesthetic interpolation problem into an optimization problem to solve. For our golden interpolation, it is also expected to explore some criteria to measure the golden degree of the function graph, and to analyse the golden methods mathematically and quantificationally. So we do the following exploration for the golden step interpolation.

Suppose the \( n + 1 \) nodes \( (x_0, y_0), (x_1, y_1), \cdots, (x_n, y_n) \) are the nodes directly input to the zero-degree spline interpolation and any two of \( y_0, y_1, \cdots, y_n \) are unequal. Consider the \( m + 1 (m \geq 2) \) adjacent nodes \( (x_d, y_d), (x_{d+1}, y_{d+1}), \cdots, (x_{d+m}, y_{d+m}) \), where \( d = 0, m, 2m, \cdots, lm \) and \( l = \lceil n/m \rceil - 1 \), where the “\( \lceil \cdot \rceil \)” represents taking the integer part. Then there are \( m \) line segments in the function graph, whose lengths are \( x_{d+1} - x_d, x_{d+2} - x_{d+1}, \cdots, x_{d+m} - x_{d+m-1} \) respectively. The following criteria are easy to explore.

1. Left golden error

If every \( m \) line segments constitute a group and we hope the ratio of the lengths of any two adjacent line segments satisfies \( \varphi \) in each group, then the following \( E_{\text{left}} \) can measure the error between present graph and the goal.

\[
E_{\text{left}} = \sum_{i=0}^{l} \sum_{j=0}^{m-2} \frac{|x_{im+j+1} - x_{im+j}|}{x_{im+j+2} - x_{im+j}} (1 - \varphi) +
\sum_{j=m+l+1}^{n-2} \frac{|x_{j+1} - x_j|}{x_{j+2} - x_j} (1 - \varphi).
\]

In this formula, when the upper limit is smaller than the lower limit of \( \Sigma, \) the sum is 0. And so is it in the rest of this paper.

2. Right golden error

If we hope the ratio is \( 1/\varphi \), then the following \( E_{\text{right}} \) can measure the error.

\[
E_{\text{right}} = \sum_{i=0}^{l} \sum_{j=0}^{m-2} \frac{|x_{im+j+1} - x_{im+j}|}{x_{im+j+2} - x_{im+j}} (1 - \varphi) +
\sum_{j=m+l+1}^{n-2} \frac{|x_{j+1} - x_j|}{x_{j+2} - x_j} (1 - \varphi).
\]
Fig. 1. Comparative examples and applications: (a) the graph of the traditional zero-degree spline interpolation function; (b) the graph of the golden equal number step interpolation function; (c) the 3D stumps generated by (a); (d) the 3D stumps generated by (b).

Fig. 2. An example and an application of the golden extension step interpolation: (a) the function graph; (b) an application of designing a picture in which meteorites are streaking across the night sky.
\[ E_{\text{right}} = \sum_{i=0}^{l} \sum_{j=0}^{m-2} \frac{|x_{im+j+1} - x_{im+j}|}{x_{im+j+2} - x_{im+j} - \varphi} + \sum_{j=m(l+1)}^{n-2} \frac{|x_{j+1} - x_j|}{x_{j+2} - x_j - \varphi}. \]

(3) Mixed golden error

If the ratio is smaller than 1, we hope it is \( \varphi \); Otherwise, we hope it is \( 1/\varphi \). Then the following \( E_{\text{mixed}} \) can measure the error.

\[ E_{\text{mixed}} = \sum_{i=0}^{l} \sum_{j=0}^{m-2} \frac{|x_{im+j+1} - x_{im+j}|}{x_{im+j+2} - x_{im+j} - q_{i,j}} + \sum_{j=m(l+1)}^{n-2} \frac{|x_{j+1} - x_j|}{x_{j+2} - x_j - q_{j}}. \]

where

\[ q_{i,j} = \begin{cases} 1 - \varphi, & \text{if } x_{im+j+1} - x_{im+j} < x_{im+j+2} - x_{im+j+1}, \\ \varphi, & \text{if } x_{im+j+1} - x_{im+j} \geq x_{im+j+2} - x_{im+j+1}, \\ 1, & \text{otherwise}. \end{cases} \]

and

\[ q_{j} = \begin{cases} 1 - \varphi, & \text{if } x_{j+1} - x_j < x_{j+2} - x_{j+1}, \\ \varphi, & \text{if } x_{j+1} - x_j \geq x_{j+2} - x_{j+1}, \\ 1, & \text{otherwise}. \end{cases} \]

(4) Alternate golden error

If we hope the ratios are \( \varphi, 1/\varphi, \varphi, 1/\varphi, \cdots \), then the following \( E_{l-r} \) can measure the error. We call it “left-right golden error”.

\[ E_{l-r} = \sum_{i=0}^{l} \sum_{j=0}^{m-2} \frac{|x_{im+j+1} - x_{im+j}| + (-1)^{j} \varphi + \text{sign}(1 - (-1)^{j})}{x_{im+j+2} - x_{im+j}} + \sum_{j=m(l+1)}^{n-2} \frac{|x_{j+1} - x_j| + (-1)^{j} \varphi + \text{sign}(1 - (-1)^{j})}{x_{j+2} - x_j}, \]

where

\[ \text{sign}(x) = \begin{cases} 1, & x > 0, \\ 0, & x = 0, \\ -1, & x < 0, \end{cases} \]

is the sign function.

If we hope the ratios are \( 1/\varphi, \varphi, 1/\varphi, \varphi, \cdots \), then we can easily construct a “right-left golden error” \( E_{r-l} \) by replacing the exponent \( j \) by \( j + 1 \) in the above formula.

The above four kinds of errors will increase with the increase of \( n \). In order to compare the golden degree of different step functions, we can compute the average error by multiplying the error by

\[ \frac{1}{(m-1)(l+1) + (n-1 - m(l+1))}. \]

to eliminate the influence of \( n \), where

\[ (x)_{+} = \begin{cases} x, & x > 0, \\ 0, & x \leq 0. \end{cases} \]

The above errors are convenient for numerical comparison, but when we want to minimize them, we often need to solve their derivatives, in which case we can replace the above absolute value operation with the square operation for convenience.

For the golden step interpolation, suppose the original interpolation nodes \( A_0(x_0, y_0), A_1(x_1, y_1), \cdots, A_n(x_n, y_n) \) are transformed into \( A'_0(x'_0, y'_0), A'_1(x'_1, y'_1), \cdots, A'_k(x'_k, y'_k) \), then the interpolation problem can be turned into the following optimization problem.

\[ \min E(x'_0, x'_1, \cdots, x'_k) \]

s.t. the final function passes through the original nodes;

other constraints on \( x'_0, x'_1, \cdots, x'_k \).

where \( E \) represents one of the above five errors: \( E_{l-r}, E_{r-l}, E_{\text{mixed}}, E_{l-r} \) and \( E_{r-l} \).

When \( m < n \), it is a local golden method, in which the golden optimization only happens in every group consisting of \( m + 1 \) nodes; When \( m = n \), it becomes a global golden method, in which the optimization target is to make the ratio of every two adjacent line segments satisfies \( \varphi \) or \( 1/\varphi \) in the whole function graph.

It is easy to work out that the golden extension step interpolation proposed in Section 4.1 is the optimum solution of the following optimization problem when \( m = 2 \) and \( k = 2n \).

\[ \min E_{l-r}(x'_0, x'_1, \cdots, x'_k) \]

s.t. \( x'_0 = x_0; \)

\( x_0 < x'_1 < x_1; \)

\( x'_1 = x_1; \)

\( x_1 < x'_2 < x_2; \)

\( x'_2 = x_2; \)

\( x_2 < x'_3 < x_3; \)

\( \cdots; \)

\( x_{n-1} < x'_{k-1} < x_n; \)

\( x'_k = x_n. \)

And the golden equal number step interpolation proposed in Section 4.2 is the optimum solution of the following optimization problem when \( m = 2 \) and \( k = n \).

\[ \min E_{l-r}(x'_0, x'_1, \cdots, x'_k) \]

s.t. \( x'_0 = x_0; \)

\( x_0 < x'_1 \leq x_1; \)

\( x'_1 = x_1; \)

\( x_1 < x'_2 \leq x_2; \)

\( x'_2 = x_2; \)

\( x_2 < x'_3 \leq x_3; \)

\( x'_3 = x_3; \)

\( x_3 < x'_4 \leq x_4; \)

\( \cdots; \)

\( x_2[0.5n-2] < x'_{k-1} \leq x_2[0.5n-1]; \)

If \( n \) is odd, add a constraint: \( x'_{k-1} = x_{n-1}; \)

\( x'_k = x_n. \)

5. Golden piecewise linear interpolation

The step function is discontinuous. In this section, we go more complex to the simplest continuous interpolation function: the piecewise linear interpolation, and we discuss how to
apply the golden section principle to it. The graph of a piecewise linear function is composed of connected line segments end to end. If we use the traditional piecewise linear interpolation method, the interpolation data nodes are exactly the ends of the line segments, so it is convenient to change the function graph by transforming the nodes. Considering that, we add a golden transform to the traditional method, struggling to make the interpolation graph look more beautiful.

5.1. Golden cuspidal hill

When thinking how to apply the golden section principle to the piecewise linear interpolation, we may firstly call to mind the golden triangles that are universally acknowledged the most beautiful among all triangles. The ratio of the lengths of a golden triangle’s sides conforms to the golden section ratio. Therefore, we naturally consider making the lengths’ ratio of two adjacent line segments of the piecewise line satisfy the golden section ratio. In our study, we have indeed done this first of all. But unfortunately, we find this strategy is bad and the interpolation graph looks even uglier. Thinking the reason carefully, it may be that compared with the triangle, the piecewise line consisting of two segments lacks of an edge, which leads to essential difference between the piecewise line and the triangle. So the golden principle in triangles is not applicable to piecewise lines.

Now that the idea inspired by the golden triangles fails, we propose another strategy in this paper. In our strategy, we first define the concept of the golden cuspidal hill as follows.

Definition 3. See Fig. 2 A piecewise line ABC consists of two segments: AB and BC. Then ABC is called cuspidal hill, and the point B is called the hilltop. Connect AC and construct a perpendicular line BH intersecting AC at the point H. If H is a golden section point of AC, then we call ABC a golden cuspidal hill and call B the golden hilltop. If H is the left(right) golden point, then ABC is called the left(right) golden cuspidal hill and B is called the left(right) golden hilltop.

![Fig. 3. Golden cuspidal hill.](image)

Fig. 3. Golden cuspidal hill.

We argue that golden cuspidal hills have beautiful elements among all piecewise lines with two segments. It is inspired by a golden application about the location of a host: It is suggested that a host should stand at the golden section point of the stage, because standing in the middle of the stage seems dull and standing too far from the middle seems disharmony. The location of the golden section point is considered natural and aesthetic. The golden cuspidal hills refer to the similar principle: The hilltop B is a break point of the graph. When this break happens in the golden section point of AC, the cuspidal hill will seem natural, harmonious and aesthetic. The basic idea of our golden piecewise linear interpolation is to make the interpolation nodes connected by golden cuspidal hills.

5.2. Golden extension piecewise linear interpolation

We add an extension transform to the piecewise linear interpolation, which adds an extra data node between each two adjacent interpolation nodes, and finally make every two adjacent interpolation nodes connected by a golden cuspidal hill. The specific interpolation process includes two steps as follows, and we call this method the golden extension piecewise linear interpolation.

step 1. Transform \( A_0, A_1, \ldots, A_n \) into \( A_0, A_{0.5}, A_1, A_{1.5}, \ldots, A_{n-0.5}, A_n \) as follows:

Add an extra data node \( A_{i+0.5}(x_{i+0.5},y_{i+0.5}) \) between each two adjacent interpolation nodes \( A_i, A_{i+1}(i = 0, 1, \ldots, n-1) \), making the piecewise line connecting \( A_i, A_{i+0.5}, A_{i+1} \) become a golden cuspidal hill. That is to say, we should make the node \( A_{i+0.5} \) become the golden hilltop of \( A_iA_{i+0.5}A_{i+1} \). See Fig. 4. There are the nodes \( A_i \) and \( A_{i+1} \). Connect \( A_iA_{i+1} \). Take the right golden section point \( H \) of the line segment \( A_iA_{i+1} \). Construct \( A_{i+0.5}HLA_{i+1} \) and make \( |A_{i+0.5}H| = q|A_iA_{i+1}| \) where \( q = 0.2 \) whose value is given by us. Then \( A_{i+0.5} \) is the added node that we want. The coordinates of \( A_{i+0.5} \) can be solved by elementary mathematics and we put the solving process into Appendix A. Here we just list the important results.

![Fig. 4. Add a node \( A_{i+0.5} \) between \( A_i \) and \( A_{i+1} \).](image)

Fig. 4. Add a node \( A_{i+0.5} \) between \( A_i \) and \( A_{i+1} \).

The slope of \( A_iA_{i+1} \) is

\[
k = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}
\]

The x-coordinate and the y-coordinate of \( A_{i+0.5} \) can be solved by the following formulas:

\[
x_{i+0.5} = \begin{cases} x_i + (x_{i+1} - x_i)\varphi + q(y_{i+1} - y_i), |k| \geq 1, \\ x_i + (x_{i+1} - x_i)\varphi - q(y_{i+1} - y_i), |k| < 1, \end{cases}
\]

\[
y_{i+0.5} = \begin{cases} y_i + (y_{i+1} - y_i)\varphi - q(x_{i+1} - x_i), |k| \geq 1, \\ y_i + (y_{i+1} - y_i)\varphi + q(x_{i+1} - x_i), |k| < 1, \end{cases}
\]

where

\[
q = \frac{|A_{i+0.5}H|}{|A_iA_{i+1}|} = 0.2
\]
whose value is given by us. Finally we should limit the scope of $x_{i+0.5}$ in order to avoid $x_{i+0.5}$ is not in $(x_i, x_{i+1})$ or too close to the endpoint of $(x_i, x_{i+1})$. We let $t = 0.5(1 - \varphi(x_{i+1} - x_i))$. If the $x_{i+0.5}$ calculated according to (2) and (3) is not in the interval $[x_i + t, x_{i+1} - t]$, then we revise the value of $q$ by letting

$$q = \frac{t}{|y_{i+1} - y_i|},$$

and recalculate $x_{i+0.5}$ and $y_{i+0.5}$ according to (2) and (5).

Step 2. Get the final interpolation function

$$p_n(x) = \frac{x - x_{0.5(i+1)}}{x_{0.5(i+1)} - x_{0.5(i)}}y_{0.5(i)} + \frac{x - x_{0.5i}}{x_{0.5(i+1)} - x_{0.5i}}y_{0.5(i+1)},$$

where $x \in [x_{0.5i}, x_{0.5(i+1)}], i = 0, 1, \ldots, 2n - 1$.

### 5.3. Golden equal number piecewise linear interpolation

By adding data nodes, the golden extension piecewise linear interpolation makes any two adjacent interpolation data nodes connected by a golden cuspidal hill. However, compared with the direct piecewise linear interpolation, due to adding data nodes, this method makes the non-differentiable points almost double in the interpolation function. In the function graph, the number of line segments increases and the length of each line segment becomes shorter. But sometimes, we want to make the graph look more beautiful on condition that the interpolation function has the same number of non-differentiable points as the direct piecewise linear interpolation function. Therefore, we also propose a golden equal number transform, in which three adjacent interpolation data nodes are extracted in turn and the data node in the middle of the three is transformed, which is aimed to make the final interpolation function graph not only pass through the original interpolation nodes but also consist of more golden cuspidal hills. The specific interpolation process includes two steps as follows, and we call this method the golden equal number piecewise linear interpolation.

Step 1. Transform $A_0, A_1, \ldots, A_n$ into $A_0, A'_1, A_2, A'_3, \ldots, A_n$ as follows:

If $n < 2$, the data nodes are not changed; that is, the transformed data nodes are the same as the original data nodes.

If $n \geq 2$, take out the data nodes $A_{2i-1}(x_{2i-1}, y_{2i-1}), i = 1, 2, \ldots, [0.5n]$ whose sequence numbers are odd numbers among the $n$ data nodes, in which $[0.5n]$ represents the integer part of $0.5n$. Transform $A_{2i-1}(x_{2i-1}, y_{2i-1})$ into $A'_{2i-1}(x'_{2i-1}, y'_{2i-1})$. The process of solving $A'_{2i-1}(x'_{2i-1}, y'_{2i-1})$ is illustrated in Fig[3] There are data nodes $A_{2i-1}, A_{2i-1}$, and $A_{2i}$. Construct $A_{2i-1}P \perp A_{2i-1}A_{2i}$ where $P$ is the foot point. If $P$ is not on the line segment $A_{2i-1}A_{2i}$ or is an endpoint of the line segment $A_{2i-1}A_{2i}$, then we let $A'_{2i-1}$ the same as $A_{2i-1}$; otherwise, we find the golden section point $H$ of $A_{2i-1}A_{2i}$ where $H$ is the the golden section point that is closer to $P$. Construct $A'_{2i-1}H \perp A_{2i-1}A_{2i}$ where $H$ is the foot point. If the x-coordinate of $H$ is greater than the x-coordinate of $P$, then extend the line segment $A_{2i-1}A_{2i-1}$, making the extended line and $A'_{2i-1}H$ intersect at the point $A'_{2i-1}$; if the x-coordinate of $H$ is equal to or smaller than the x-coordinate of $P$, then extend the line segment $A_{2i-1}A_{2i-1}$, making the extended line and $A'_{2i-1}H$ intersect at the point $A'_{2i-1}$. Then, the piecewise line $A_{2i-2}A'_{2i-2}A_{2i}$ not only passes through the original nodes $A_{2i-2}, A_{2i-1}$ and $A_{2i}$, but also is a golden cuspidal hill. However, the x-coordinate of $A'_{2i-1}$ may not be in the interval $(x_{2i-2}, x_{2i})$, in which condition the x-coordinates of the transformed data nodes may not be monotonically increasing so we can’t construct interpolation function for them. When this happens, we make $A'_{2i-1}$ the same as $A_{2i-1}$; that is, we don’t change $A_{2i-1}$. Therefore, this transform just tries to do better but can’t guarantee that the piecewise line $A_{2i-2}A'_{2i-2}A_{2i}$ becomes a golden cuspidal hill. The specific calculation steps and formulas to solve $A'_{2i-1}$ are as follows.

![Fig. 5. Transform the node $A_{2i-1}$ into the node $A'_{2i-1}$.](image)

**step a.** Solve $P$. Let $k$ stand for the slope of $A_{2i-2}A_{2i}$. Then

$$k = \frac{y_{2i} - y_{2i-2}}{x_{2i} - x_{2i-2}}.$$

Using the knowledge of elementary mathematics, we can calculate that the coordinates of $P$ are

$$\begin{cases}
  x_p = \frac{k(y_{2i-1} - y_{2i-2}) + k^2x_{2i-2} + x_{2i-1}}{1 + k^2}, \\
y_p = \frac{k(x_{2i-1} - x_{2i-2}) + k^2y_{2i-1} + y_{2i-2}}{1 + k^2}.
\end{cases}$$

If $x_p \notin (x_{2i-2}, x_{2i})$, then make $A'_{2i-1}$ the same as $A_{2i-1}$; that is,

$$\begin{cases}
x'_{2i-1} = x_{2i-1}, \\
y'_{2i-1} = y_{2i-1}.
\end{cases}$$

If $x_p \in (x_{2i-2}, x_{2i})$, then continue the following steps.

**step b.** Solve $H$. The x-coordinate of the midpoint of the line segment $A_{2i-2}A_{2i}$ is

$$x_c = \frac{x_{2i-2} + x_{2i}}{2}.$$

If $x_c > x_z$, then the coordinates of $H$ are

$$\begin{cases}
x_h = x_{2i-2} + (x_z - x_{2i-2})\varphi, \\
y_h = y_{2i-2} + (y_z - y_{2i-2})\varphi;
\end{cases}$$

if $x_c \leq x_z$, then the coordinates of $H$ are

$$\begin{cases}
x_h = x_z - (x_z - x_{2i-2})\varphi, \\
y_h = y_z - (y_z - y_{2i-2})\varphi.
\end{cases}$$
step c. Solve $A'_{2i-1}$. Let

$$c = \begin{cases}  
\frac{y_{2i-1} - y_{2i-2}}{x_{2i-1} - x_{2i-2}}, & x_h > x_p, \\
\frac{y_{2i-1} - y_2}{x_{2i-1} - x_2}, & x_h \leq x_p. 
\end{cases}$$

Then the coordinates of $A'_{2i-1}$ are

$$\begin{align*}
    x'_{2i-1} &= \frac{x_h + k(cy_{2i-1} + y_h - y_{2i-1})}{1 + ck}, \\
y'_{2i-1} &= \frac{y_{2i-1} + c(ky_h + x_h - x_{2i-1})}{1 + ck}.
\end{align*}$$

step d. Check and decide the value. If $x'_{2i-1} \notin (x_{2i-2}, x_{2i})$, then make $A'_{2i-1}$ the same as $A_{2i-1}$; that is,

$$\begin{align*}
x'_{2i-1} &= x_{2i-1}, \\
y'_{2i-1} &= y_{2i-1}.
\end{align*}$$

step 2. Mark $A_0, A'_1, A_2, A'_3, \ldots, A_n$ as $A'_0(x_{0'}, y_{0'}), A'_1(x'_1, y'_1), A'_2(x'_2, y'_2), A'_3(x'_3, y'_3), \ldots, A'_n(x'_n, y'_n)$, and then get the final interpolation function

$$p_i(x) = \frac{x - x'_{i+1}}{x'_i - x'_{i+1}} y'_i + \frac{x - x'_i}{x'_{i+1} - x'_i} y'_{i+1},$$

where $x \in [x'_i, x'_{i+1}], i = 0, 1, \ldots, n-1$.

5.4. Examples and applications

We demonstrate comparative examples and applications about the shape design of landscape lights: Given design nodes $C_0(6, 13), C_1(20, 18), C_2(24, 13), C_3(58, 18), C_4(42, 13), C_5(56, 18), C_6(60, 13)$, construct the profiles of the landscape lights by doing the traditional piecewise linear interpolation, the golden extension piecewise linear interpolation and the golden equal number piecewise linear interpolation respectively for them. The graphs of these three interpolation functions are shown in Fig. 6(a), Fig. 6(b) and Fig. 6(c), which are the three profiles constructed by the three different methods. It should be noted that the unit of the $x$-axis should be equal to the unit of the $y$-axis when drawing, so as to avoid the distortion of the graph in the visual effect. Rotate the profiles around the line $y = 10$, and then the 3D models of the landscape lights can be obtained, which are shown in Fig. 6(d), Fig. 6(e) and Fig. 6(f). Compared with Fig. 6(d), Fig. 6(e) and Fig. 6(f) integrate the golden section into the geometric features, which makes the designed model look more pleasant.

5.5. Go further to explore criteria

We can explore criteria for the golden piecewise linear interpolation in the similar way of exploring the golden step interpolation criteria.

Suppose the $n + 1$ nodes $A_0(x_0, y_0), A_1(x_1, y_1), \ldots, A_n(x_n, y_n)$ are directly input to the piecewise linear interpolation. For a node $A_{i+1}(i = 0, 1, \ldots, n-2)$, it is the hilltop of the cuspidal hill $A_iA_{i+1}A_{i+2}$. Construct $A_{i+1}C \perp A_iA_{i+2}$ where $C(x_c, y_c)$ is the foot point.

Then

$$x_c = \frac{k(y_{i+1} - y_i) + k^2x_i + x_{i+1}}{1 + k^2}, \quad \text{where } k = \frac{y_{i+2} - y_i}{x_{i+2} - x_i}.$$ 

Get the ratio

$$t_{i+1} = \frac{x_c - x_i}{x_{i+2} - x_i} = \frac{(y_{i+2} - y_i)(y_{i+1} - y_i) + (x_{i+2} - x_i)(x_{i+1} - x_i)}{(x_{i+2} - x_i)^2 + (y_{i+2} - y_i)^2}.$$ 

Replace the

$$\frac{x_{im+j+1} - x_{im+j}}{x_{im+j+2} - x_{im+j}} \quad \text{and} \quad \frac{x_{j+1} - x_j}{x_{j+2} - x_j}$$

in Section 4.4 with $t_{im+j+1}$ and $t_{j+1}$, and then we can get the errors $E_{left}, E_{right}, E_{mixed}, E_{left}^r$ and $E_{right}^r$ for the the golden piecewise linear interpolation with a value of $m$ given. What’s different from the step interpolation is that the ratios $t_{im+j+1}$ and $t_{j+1}$ may be negative when $\angle A_{i+1}A_iA_{i+2}$ is an obtuse angle, but it doesn’t matter for the evaluation of the errors. The meanings of these kinds of errors are as follows.

(1) The left(right) golden error $E_{left}(E_{right})$

Every $m + 1$ nodes forms a group. If we hope every cuspidal hill in each group is a left(right) golden cuspidal hill, then the error $E_{left}(E_{right})$ can measure the distance between the present graph and the goal.

(2) The mixed golden error $E_{mixed}$

If we hope every cuspidal hill in each group is a golden cuspidal hill, no matter what kind of golden cuspidal hill it is, then the error $E_{mixed}$ can be a measure.

(3) The alternate golden error $E_{left}^r$ and $E_{right}^r$

If we hope the cuspidal hills in each group are “the left golden cuspidal hill, the right golden cuspidal hill, the left golden cuspidal hill, the right golden cuspidal hill...”, then the “left-right golden error” $E_{left}^r$ can be a measure. Similarly, the meaning of the “right-left golden error” $E_{right}^r$ is easy to known.

As for the average error, the square error and the relevant optimization problems, they are similar to those in Section 4.4. However, for the golden piecewise linear interpolation, the optimization problems and their constraints are more complex.

Suppose the original interpolation nodes $A_0(x_0, y_0), A_1(x_1, y_1), \ldots, A_n(x_n, y_n)$ are transformed into $A'_0(x'_0, y'_0), A'_1(x'_1, y'_1), \ldots, A'_n(x'_n, y'_n)$.
The golden extension piecewise linear interpolation proposed in Section 5.2 is one of the optimum solutions of the following optimization problem when \( m = 2 \) and \( k = 2n \).

\[
\begin{align*}
\text{min} & \quad E_{right}(x'_0, x'_1, \ldots, x'_k, y'_0, y'_1, \ldots, y'_k) \\
\text{s.t.} & \quad x'_0 = x_0 \text{ and } y'_0 = y_0; \\
& \quad x_0 + 0.5(1-\varphi)(x_1 - x_0) \leq x'_1 \leq x_1 - 0.5(1-\varphi)(x_1 - x_0); \\
& \quad x'_2 = x_1 \text{ and } y'_2 = y_1; \\
& \quad x_1 + 0.5(1-\varphi)(x_2 - x_1) \leq x'_3 \leq x_2 - 0.5(1-\varphi)(x_2 - x_1); \\
& \quad x'_4 = x_2 \text{ and } y'_4 = y_2; \\
& \quad x_2 + 0.5(1-\varphi)(x_3 - x_2) \leq x'_5 \leq x_3 - 0.5(1-\varphi)(x_3 - x_2); \\
& \quad \ldots \\
& \quad x_{n-1} + 0.5(1-\varphi)(x_n - x_{n-1}) \leq x'_k \leq x_n - 0.5(1-\varphi)(x_n - x_{n-1}); \\
& \quad x'_k = x_n \text{ and } x'_k = x_n.
\end{align*}
\]

The above optimization problem has countless optimum solutions. The setting of \( q \) and the choice of the above node or the below node according to the slope \( k \) in Section 5.2 are to select a solution from all the optimum solutions.

The golden equal number piecewise linear interpolation proposed in Section 5.3 is related to the following optimization problem when \( m = 2 \) and \( k = n \). But it isn’t the optimal solution and needs to be improved.

\[
\begin{align*}
\text{min} & \quad E_{mixed}(x'_0, x'_1, \ldots, x'_k, y'_0, y'_1, \ldots, y'_k) \\
\text{s.t.} & \quad x'_0 = x_0 \text{ and } y'_0 = y_0; \\
& \quad \begin{cases} 
(x'_1, y'_1) \mid x_0 < x'_1 \leq x_1 & = y_2 - y_1 \\
\quad x'_1 < x_1 & = y_2 - y_1 \\
\end{cases} \cup \\
& \quad \begin{cases} 
(x'_1, y'_1) \mid x_1 < x'_2 \leq x_2 & = y_0 - y_1 \\
\quad x'_1 < x_2 & = y_0 - y_1 \\
\end{cases} \\
& \quad x'_2 = x_2 \text{ and } y'_2 = y_2; \\
& \quad \begin{cases} 
(x'_3, y'_3) \mid x_2 < x'_3 \leq x_3 & = y_4 - y_3 \\
\quad x'_3 < x_3 & = y_4 - y_3 \\
\end{cases} \cup \\
& \quad \begin{cases} 
(x'_3, y'_3) \mid x_3 < x'_4 \leq x_4 & = y_2 - y_3 \\
\quad x'_3 < x_4 & = y_2 - y_3 \\
\end{cases} \\
& \quad x'_4 = x_4 \text{ and } y'_4 = y_4; \\
& \quad \begin{cases} 
(x'_5, y'_5) \mid x_4 < x'_5 \leq x_5 & = y_6 - y_5 \\
\quad x'_5 < x_5 & = y_6 - y_5 \\
\end{cases} \cup \\
& \quad \begin{cases} 
(x'_5, y'_5) \mid x_5 < x'_6 \leq x_6 & = y_4 - y_5 \\
\quad x'_5 < x_6 & = y_4 - y_5 \\
\end{cases} \\
& \quad \ldots \\
& \quad \begin{cases} 
(x'_k, y'_k) \mid x_{n-1} \leq x'_k \leq x_n & = y_{n-1} - y_n \\
\quad x_{n-1} < x'_k & = y_{n-1} - y_n \\
\end{cases} \cup \\
& \quad \begin{cases} 
(x'_k, y'_k) \mid x_k < x'_k \leq x_{k+1} & = y_{n+1} - y_k \\
\quad x_k < x'_k & = y_{n+1} - y_k \\
\end{cases}.
\end{align*}
\]

If \( n \) is odd, add constraints: \( x'_{k-1} = x_{n-1} \) and \( y'_{k-1} = y_n \); \( x'_{k} = x_n \) and \( y'_{k} = y_n \).
The above optimization problem is rough. It may have more than one optimal solutions, so we need further strategies to select one. The error of the optimal solution may be very large, so some constraints on errors need be further added (So is it in Section 5.3). All like these need to be further considered in practical methods.

6. Golden curve interpolation

In previous sections, we have discussed the discontinuous step function and the $C^0$ smooth piecewise linear function. In this section, we go further to the $C^1$ smooth function. We’ll discuss how to apply the golden section principle to a smooth interpolation curve with a continuous first-order derivative for aesthetic. Spline functions are made up of piecewise polynomials with a certain degree of smoothness. They have both smoothness and flexibility. Among them, the quadratic spline has a continuous first-order derivative, and it also has a good convexity-preserving. Lots of work studies the shape-preserving interpolation curve based on the quadratic spline [31][32][33]. That is because there will be no excess inflection points when we interpolate with a quadratic spline. For these reasons, we choose the quadratic spline interpolation as a basis to control shape conveniently, and we add a golden extension transform to it, aiming to integrate the golden aesthetic into the interpolation curve.

6.1. Golden domed hill

We have defined the cuspidal hill in section 5.1. Similarly, in this section, we propose the concept of the golden domed hill for introducing our golden curve interpolation.

Definition 4. See Fig.8 In the plane right-angle coordinate system, there is a curve with $A(x_a, y_a)$ and $C(x_c, y_c)$ as the end points and $x_a < x_c$. Suppose that the functional expression of the curve is $f(x)$. $f(x)$ has a continuous first-order derivative $f'(x)$ and has no inflection points in the interval $(x_a, x_c)$. According to the Lagrange mean value theorem, there exists $x_b \in (x_a, x_c)$ which can make

$$f'(x_b) = \frac{y_c - y_a}{x_c - x_a};$$

that is, there exists a node $B(x_b, f(x_b))$ on the curve where the tangent slope at the node $B$ equals the slope of the line $AC$. Construct $BH \perp AC$ where $H$ is the foot point. If the node $H$ is a golden section point of the line segment $AC$, then the curve $ABC$ is called a golden domed hill or a golden domed hill curve. And the node $B$ is called the golden hilltop of the curve $ABC$. If $H$ is the left(right) golden point, then $ABC$ is called the left(right) golden domed hill and $B$ is called the left(right) golden hilltop.

Fig. 8. Golden domed hill.

We argue that golden domed hill curves are the aesthetic curves. The reason is similar to the golden cuspidal hill. Therefore, for the golden curve interpolation, our basic idea is to make the function graph contain more golden domed hills.

6.2. Golden extension curve interpolation

Given interpolation data nodes $A_0(x_0, y_0), A_1(x_1, y_1), \cdots, A_n(x_n, y_n)$ and the derivative $k_0$ at the node $A_0$, then their quadratic spline interpolation function is uniquely determined. If a spline knot $x_{i+0.5}$ is added between every two adjacent data nodes $A_i, A_{i+1}(i = 0, 1, \cdots, n - 1)$ but the value of the function at $x_{i+0.5}$ is not given, then the quadratic spline interpolation function is not unique and increases $n$ degrees of freedom. The basic idea of our golden extension curve interpolation is to take advantage of the $n$ degrees of freedom to make the interpolation data nodes connected by more golden domed hills.

To unify all of our golden methods in one framework: the “node transform + traditional interpolation” pattern, so that we can edit the shape of the function graph by editing the data nodes, we do the golden extension curve interpolation according to the following two steps.

step 1. Transform $A_0, A_1, \cdots, A_n$ into $A_0, (A_{0.5}), A_1, (A_{1.5}), \cdots, (A_{n-0.5}), A_n$ as follows. Here the nodes in brackets “()” may exist and may not exist, which is determined by the following specific computation.

Suppose we add a node $A_{i+0.5}(x_{i+0.5}, y_{i+0.5})$ between the node $A_i$ and $A_{i+1}(i = 0, 1, \cdots, n - 1)$, and we try to make the section of the curve in the interval $[x_i, x_{i+1}]$ become a golden domed hill curve and make the data node $A_{i+0.5}$ become the golden hilltop of this section of the curve. Here we take the right domed hill for example, and the left domed hill in a similar way.

Let $H$ be the right golden point of the line segment $A_iA_{i+1}$. Then the coordinates of $H$ are

$$\begin{aligned}
    x_h &= x_i + (x_{i+1} - x_i)\varphi, \\
    y_h &= y_i + (y_{i+1} - y_i)\varphi.
\end{aligned}$$

We have got the transformed nodes $A_0, (A_{0.5}), A_1, (A_{1.5}), \cdots, A_i$ in the previous iterations, and for convenience we mark them as $B_0(x'_0, y'_0), B_1(x'_1, y'_1), \cdots, B_d(x'_d, y'_d)$, where $d + 1$ is the actual number of the nodes $A_0, (A_{0.5}), A_1, (A_{1.5}), \cdots, A_i$.

Suppose the expression of the quadratic spline interpolation function is $p(x)$. Referring to the Section 5, we
The following equations:
\[
p'(x_{i+0.5}) = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} + \sum_{j=0}^{d-1} \left(-1\right)^{d+j} \frac{y'_{j+1} - y'_j}{x'_{j+1} - x'_j} + \left(-1\right)^{d+1} k_0.
\]

If \(A_{i+0.5}\) is the right golden hilltop, we can list the following equations:

When \(y_{i+1} = y_i\),
\[
p'(x_{i+0.5}) = \frac{y_{i+1} - y_i}{x_{i+1} - x_i},
\]
\[
x_{i+0.5} = x_i.
\]

When \(y_{i+1} \neq y_i\),
\[
p'(x_{i+0.5}) = \frac{y_{i+1} - y_i}{x_{i+1} - x_i},
\]
\[
\frac{y_{i+0.5} - y_i}{x_{i+0.5} - x_i} = \frac{y_{i+0.5} - y_i}{x_{i+0.5} - x_i} = -1.
\]

Then the equations \(5\) above can be solved according to the following formulae \(6\).
\[
\begin{cases}
x_{i+0.5} = x_i, \\
y_{i+0.5} = y_i + t(x_{i+0.5} - x_i).
\end{cases}
\]

And the equations \(5\) above can be solved according to the following formulae \(7\).
\[
\begin{cases}
x_{i+0.5} = (y_{i+1} - y_i)(y_i - y_i + Ix_i) + x_k(x_{i+1} - x_i) \\
\frac{t(y_{i+1} - y_i) + x_k(x_{i+1} - x_i)}{I(x_{i+1} - x_i)}. \\
y_{i+0.5} = y_i + t(x_{i+0.5} - x_i).
\end{cases}
\]

We can observe that the formulæ \(6\) and \(7\) can be unified to the one formulæ \(7\).

Finally, we should judge whether \(x_{i+0.5}\) is valid. If \(x_{i+0.5} \in (x_i, x_{i+1})\), then add the node \(A_{i+0.5}\) (that is, set \(A_{i+0.5}\) as \(A_{i+0.5}\)); if \(x_{i+0.5} \notin (x_i, x_{i+1})\), then don’t add the node \(A_{i+0.5}\) (that is, set \(A_{i+0.5}\) to be empty).

Step 2. Mark the nodes \(A_0, (A_3), A_1, (A_5), \cdots, (A_n-0.5), A_n\) as \(B_0(x', y'), B_1(x', y'), \cdots, B_d(x', y')\), where \(d + 1\) is the actual number of the transformed nodes. Then get the final interpolation function
\[
p(x) = \left(\frac{x - x'_{i+1}}{x'_{i+1} - x'_{i+0.5}}\right)^2 y'_{i+0.5} + \left(\frac{x - x'_{i}}{x'_{i+1} - x'_{i}}\right)^2 y'_{i+1} + \frac{(x - x')(x - x'_{i+1})}{x'_{i+1} - x'_{i}} \left[(x - x'_{i+1}) + \sum_{j=0}^{d-1} \left(-1\right)^{j+c+1} \frac{y'_{j+1} - y'_j}{x'_{j+1} - x'_j} + \left(-1\right)^{d+1} k_0 \right],
\]
where \(x \in [x', x'_{i+1}], i = 0, 1, \cdots, d - 1\).
Fig. 9. Comparative examples and applications: (a) the graph of the traditional quadratic spline interpolation; (b) the graph of the golden extension curve interpolation; (c) the 3D vase model generated by (a); (d) the 3D vase model generated by (b).

Fig. 10. Comparative examples and applications: (a) the graph of the traditional quadratic spline interpolation; (b) the graph of the golden extension curve interpolation; (c) the 3D headboard model generated by (a); (d) the 3D headboard model generated by (b).
points in the interval \((a, b)\). See Fig. 11. It is a domed hill. How much far away is it from a golden domed hill?

We have discussed how to measure the golden degree of a curve and we can observe: The golden measure of a curve is related to how to partition the definition domain of the curve function. For the same curve, the golden errors may vary with different partitions. So the partition is important and it indicates which curve segments we want to become golden domed hills. For the golden extension curve interpolation proposed in Section 6.2, the interpolation data nodes play two roles: (1) interpolation constrains; (2) partition the definition domain. And we can understand that we choose to use the quadratic spline is to make the curve segments in the partitioned intervals become domed hills without inflection points.

After the problem of golden measure, then we will discuss how to convert the golden curve interpolation problem into a more general optimization problem.

Reconsider the golden extension curve interpolation in Section 6.2. We abstract the problem into the following form.

Give interpolation conditions \(A_0(x_0, y_0), A_1(x_1, y_1), \ldots, A_r(x_r, y_r)\).

Give partition points \(x_1, x_2, \ldots, x_{n-1}\) in the definition domain \([x_0, x_n]\).

Add nodes \(A_{0.5}(x_{0.5}, y_{0.5}), A_{1.5}(x_{1.5}, y_{1.5}), \ldots, A_{n-0.5}(x_{n-0.5}, y_{n-0.5})\). They are the unknowns.

Then the quadratic spline interpolation function \(p(x)\) can be constructed for the nodes \(A_0, A_{0.5}, A_{1.5}, \ldots, A_{n-0.5}, A_n\). And the golden error \(E\) of \(p(x)\) can be calculated as mentioned above. Then the golden extension curve interpolation in Section 6.2 is related to the following optimization problem but it isn’t the optimum solution.

\[
\begin{align*}
\min \quad & E_{\text{right}}(x_{0.5}, x_{1.5}, \ldots, x_{n-0.5}, y_{0.5}, y_{1.5}, \ldots, y_{n-0.5}) \\
\text{s.t.} \quad & A_{0.5}, A_{1.5}, \ldots, A_{n-0.5} \text{ are golden hilltops; } x_{0.5}, x_{1.5}, \ldots, x_{n-0.5} \text{ are in the intervals } (x_0, x_1), (x_1, x_2), \ldots, (x_{n-1}, x_n) \text{ respectively; } \\
& y_{0.5}, y_{1.5}, \ldots, y_{n-0.5} \text{ into the "(" of the } E_{\text{right}}(\cdot) \text{"}. \\
\end{align*}
\]

The above optimization problem requires that the added nodes should be golden hilltops. A more general problem is removing this condition as follows.

\[
\begin{align*}
\min \quad & E(x_{0.5}, x_{1.5}, \ldots, x_{n-0.5}, y_{0.5}, y_{1.5}, \ldots, y_{n-0.5}) \\
\text{s.t.} \quad & x_{0.5}, x_{1.5}, \ldots, x_{n-0.5} \text{ are in the intervals } (x_0, x_1), (x_1, x_2), \ldots, (x_{n-1}, x_n) \text{ respectively.} \\
\end{align*}
\]

But solving this problem is more complicated.

Similarly, other optimization problems of the golden curve interpolation can be established. For example, here the interpolation data nodes play a role in partitioning the definition domain, but we can give another partition to specially express which curve segments we want to become golden domed hills. For another example, here we only propose a kind of node transform, but more various node transforms are worth exploring.
7. Conclusion and prospect

For the classic aesthetic interpolation problem, this paper has proposed a novel solving thought that is applying the golden section principle. But how should we apply the golden section to interpolation methods, making the interpolation graph not only show the personality characteristics to the given nodes, but also be integrated into the aesthetic characteristics of the golden section? To answer this question, this paper starts from the simplest case, and gradually goes to more complex, which gives the examples of applying the golden section to the interpolation of degree 0, 1 and 2. And in the last two kinds of examples, we have proposed the novel and interesting concepts of the golden cuspidal hill and the golden domed hill. The related ideas, methods, criteria, comparative examples and applications are also presented. The entirely new idea for the classic interpolation problem is promising. This paper is to do the initial exploration and provide the reference for further researches and better methods.

At the end of this paper, we also point out what needs to be improved in our work and what is worth exploring in the future: (1) If the philosophy of pursuing interpolation which meet “golden” principle would be accompanied by more analyses, validation and user study, the novel thought will be more convincing. (2) The parametric curve interpolation is more widely used, so it is very meaningful to explore how to apply the golden section principle to these parametric curves. (3) This paper only presents one-dimensional golden interpolation methods, but the higher dimensional golden interpolation is worth exploring in the future. (4) For applying the golden section principle to interpolation, we only initially explore three examples, but more applications and better methods need to be further explored.

Acknowledgments

We would like to thank anonymous reviewers for their insightful comments and suggestions which greatly improve the quality of this paper. We thank Professor Liu Ligang from University of Science and Technology of China for his concern and guidance for this paper. We thank Zhao Wang for helping to modify the format of this paper. We are grateful for the Numerical Methods course of North China University of Science and Technology which is a national level excellent course. This work was partially supported by the National Nature Science Foundation of China (51674121, 61702184) and the Scientific and Technological Project for the Returned Overseas Scholars in HeBei (C2015005014).

Appendix A. The solving process of $A_{i+0.5}$ in Section 5.2

To solve the $A_{i+0.5}$ in section 5.2, we add some auxiliary lines in Fig 5.2 construct $A_{i+0.5}F \perp FH$ where $F$ is the foot point; make $A_{i}D \perp DA_{i+1}$ where $D$ is the foot point.

$H$ is the golden section point that is closer to $A_{i+1}$, so the coordinates of $H$ are

\[
\begin{align*}
    x_h &= x_i + (x_{i+1} - x_i)\phi, \\
    y_h &= y_i + (y_{i+1} - y_i)\phi.
\end{align*}
\]

The slope of $A_iA_{i+1}$ is

\[ k = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}. \]

When $k = 0$, the coordinates of $A_{i+0.5}$ can be calculated by the formula

\[
\begin{align*}
    x_{i+0.5} &= x_i + (x_{i+1} - x_i)\phi, \\
    y_{i+0.5} &= y_i + q(x_{i+1} - x_i).
\end{align*}
\]

When $k \neq 0$, it is obvious that $\Delta A_{i+0.5}FH \sim \Delta A_{i}DA_{i+1}$. Then

\[
\frac{|A_{i+0.5}F|}{|A_iD|} = \frac{|FH|}{|DA_{i+1}|} = \frac{|A_{i+0.5}H|}{|A_iA_{i+1}|} = q.
\]

So

\[ |A_{i+0.5}F| = q|A_iD| = q(x_{i+1} - x_i), \]

and

\[ |FH| = q|DA_{i+1}| = q|y_{i+1} - y_i|. \]

We give the regulation that when $|k| > 1$, the node $A_{i+0.5}$ is below $A_{i}A_{i+1}$ and when $|k| < 1$, the node $A_{i+0.5}$ is above $A_{i}A_{i+1}$.

Therefore, when $|k| > 1$, the x-coordinate of $A_{i+0.5}$ is

\[ x_{i+0.5} = x_h + \text{sign}(k) \cdot |FH| = x_i + (x_{i+1} - x_i)\phi + q(y_{i+1} - y_i), \]

and the y-coordinate of $A_{i+0.5}$ is

\[ y_{i+0.5} = y_h - |A_{i+0.5}F| = y_i + (y_{i+1} - y_i)\phi - q(x_{i+1} - x_i); \]

when $|k| < 1$, the x-coordinate of $A_{i+0.5}$ is

\[ x_{i+0.5} = x_h - |FH| = x_i + (x_{i+1} - x_i)\phi - q|y_{i+1} - y_i|, \]

and the y-coordinate of $A_{i+0.5}$ is

\[ y_{i+0.5} = y_h + |A_{i+0.5}F| = y_i + (y_{i+1} - y_i)\phi + q(x_{i+1} - x_i). \]

In summary, the x-coordinate formula of $A_{i+0.5}$ is

\[
\begin{align*}
    x_{i+0.5} &= \begin{cases} 
    x_i + (x_{i+1} - x_i)\phi + q(y_{i+1} - y_i), & |k| \geq 1, \\
    x_i + (x_{i+1} - x_i)\phi - q(y_{i+1} - y_i), & |k| < 1,
    \end{cases}
\end{align*}
\]

and the y-coordinate formula of $A_{i+0.5}$ is

\[
\begin{align*}
    y_{i+0.5} &= \begin{cases} 
    y_i + (y_{i+1} - y_i)\phi - q(x_{i+1} - x_i), & |k| \geq 1, \\
    y_i + (y_{i+1} - y_i)\phi + q(x_{i+1} - x_i), & |k| < 1.
    \end{cases}
\end{align*}
\]
Algorithm 1 Golden extension step interpolation

**Input:** \( A_0(x_0, y_0), A_1(x_1, y_1), \ldots, A_n(x_n, y_n) \), a given \( x \in [x_0, x_n] \)

**Output:** the value of \( p_n(x) \)

1: // Do the golden extension transform
2: for \( i = 0 : n - 1 \)
3: Create a new node \( A_{i+0.5}(x_{i+0.5}, y_{i+0.5}) \);
4: \( x_{i+0.5} \leftarrow x_{i+1} - (x_{i+1} - x_i) \cdot \varphi; \)
5: if \( y_i \neq y_{i+1} \)
6: \( y_{i+0.5} \leftarrow y_i + (y_{i+1} - y_i) \cdot \varphi; \)
7: else
8: \( y_{i+0.5} \leftarrow y_{i+1} \);
9: // Do the traditional interpolation for the transformed nodes
10: for \( i = 0 : 2n - 1 \)
11: if \( x \in [x_{0.5i}, x_{0.5(i+1)}] \)
12: return \( y_{0.5i} \);

Algorithm 2 Golden equal number step interpolation

**Input:** \( A_0(x_0, y_0), A_1(x_1, y_1), \ldots, A_n(x_n, y_n) \), a given \( x \in [x_0, x_n] \)

**Output:** the value of \( p_n(x) \)

1: // Do the golden equal number transform
2: if \( n \geq 2 \)
3: for \( i = 1 : Integer(0.5n) \)
4: \( x_k \leftarrow x_{2i} - (x_{2i} - x_{2i-2}) \cdot \varphi; \)
5: if \( x_k < x_{2i-1} \)
6: \( x_{2i-1} \leftarrow x_k; \)
7: // Do the traditional interpolation for the transformed nodes
8: for \( i = 0 : n - 1 \)
9: if \( x \in [x_i, x_{i+1}] \)
10: return \( y_i \);

Algorithm 3 Golden extension piecewise linear interpolation

**Input:** \( A_0(x_0, y_0), A_1(x_1, y_1), \ldots, A_n(x_n, y_n) \), a given \( x \in [x_0, x_n] \)

**Output:** the value of \( p_n(x) \)

1: // Do the golden extension transform
2: for \( i = 0 : n - 1 \)
3: Create a new node \( A_{i+0.5}(x_{i+0.5}, y_{i+0.5}) \);
4: \( k \leftarrow (y_{i+1} - y_i)/(x_{i+1} - x_i); \)
5: \( q \leftarrow 0.2; \)
6: if \( |k| \geq 1 \)
7: \( x_{i+0.5} \leftarrow x_i + (x_{i+1} - x_i) \cdot \varphi + q \cdot (y_{i+1} - y_i); \)
8: \( y_{i+0.5} \leftarrow y_i + (y_{i+1} - y_i) \cdot \varphi - q \cdot (x_{i+1} - x_i); \)
9: else
10: \( x_{i+0.5} \leftarrow x_i + (x_{i+1} - x_i) \cdot \varphi - q \cdot (y_{i+1} - y_i); \)
11: \( y_{i+0.5} \leftarrow y_i + (y_{i+1} - y_i) \cdot \varphi + q \cdot (x_{i+1} - x_i); \)
12: \( t \leftarrow 0.5 \cdot (1 - \varphi) \cdot (x_{i+1} - x_i) \)
13: if \( x_{i+0.5} \notin [x_i + t, x_{i+1} - t] \)
14: \( q \leftarrow t/(y_{i+1} - y_i); \)
15: Do the previous “if...else...” codes again here;
16: // Do the traditional interpolation for the transformed nodes
17: for \( i = 0 : 2n - 1 \)
18: if \( x \in [x_{0.5i}, x_{0.5(i+1)}] \)
19: return \( (x - x_{0.5i+1})/(x_{0.5i} - x_{0.5(i+1)}) \cdot y_{0.5i} + (x - x_{0.5i})/(x_{0.5(i+1)} - x_{0.5i}) \cdot y_{0.5(i+1)}; \)
Algorithm 4 Golden equal number piecewise linear interpolation

**Input:** $A_0(x_0,y_0), A_1(x_1,y_1), \ldots, A_n(x_n,y_n)$, a given $x \in [x_i, x_{i+1}]$.

**Output:** the value of $p(x)$.

1: // Do the golden extension transform
2: if $n \geq 2$ then
3: for $i = 1 : Integer(0.5n)$ do
4: // Step a. Solve $P$
5: $k \leftarrow (y_{2i} - y_{2i-2})/(x_{2i} - x_{2i-2});$
6: $x_p \leftarrow (k*x_{2i-1} + k*y_{2i-2} + x_{2i-2})/(1 + k*k);$
7: $y_p \leftarrow (k*(y_{2i-1} - y_{2i-2}) + k*y_{2i-1} + y_{2i-2})/(1 + k*k);$
8: if $x_p \in (x_{2i-2}, x_{2i})$ then
9: // Step b. Solve $H$
10: $x_i \leftarrow (x_{2i-2} + x_{2i})/2;$
11: if $x_p > x_i$ then
12: $x_h \leftarrow x_{2i} + (x_{2i} - x_{2i-2})*\varphi;$
13: $y_h \leftarrow y_{2i} + (y_{2i} - y_{2i-2})*\varphi;$
14: else
15: $x_h \leftarrow x_{2i} - (x_{2i} - x_{2i-2})*\varphi;$
16: $y_h \leftarrow y_{2i} - (y_{2i} - y_{2i-2})*\varphi;$
17: // Step c. Solve $A'_{2i-1}$
18: if $x_h > x_p$ then
19: $c \leftarrow (y_{2i-1} - y_{2i-2})/(x_{2i-1} - x_{2i-2});$
20: else
21: $c \leftarrow (y_{2i-1} - y_{2i})/(x_{2i-1} - x_{2i});$
22: $x_{2i-1} \leftarrow (x_h + k*c*x_{2i-1} + y_h - y_{2i-1})/(1 + c*k);$
23: $y_{2i-1} \leftarrow (y_{2i-1} + c*k*(y_h/y_i - x_h - x_{2i-1}))/(1 + c*k);$
24: // Step d. Check and decide the value
25: if $x_{2i-1} \in (x_{2i-2}, x_{2i})$ then
26: //Change the node $A_{2i-1}$
27: $x_{2i-1} \leftarrow x_{2i-1}';$
28: $y_{2i-1} \leftarrow y_{2i-1}';$
29: // Do the traditional interpolation for the transformed nodes
30: for $i = 0 : n - 1$ do
31: if $x \in [x_i, x_{i+1}]$ then
32: return $(x - x_{i+1})/(x_i - x_{i+1}) * y_i + (x - x_i)/(x_{i+1} - x_i) * y_{i+1};$

Algorithm 5 Golden curve interpolation

**Input:** $A_0(x_0,y_0), A_1(x_1,y_1), \ldots, A_n(x_n,y_n)$, the derivative $k_0$ at the node $A_0$, a given $x \in [x_i, x_{i+1}]$

**Output:** the value of $p(x)$.

1: // Do the golden extension transform
2: Create a sequential node list $L = \{A_0\}$;
3: for $i = 0 : n - 1$ do
4: $x_h \leftarrow x_i + (x_{i+1} - x_i) * \varphi;$
5: $y_h \leftarrow y_i + (y_{i+1} - y_i) * \varphi;$
6: $d \leftarrow Length(L) - 1;$
7: $sum \leftarrow 0;$
8: for $j = 0 : d - 1$ do
9: // The $L(j).x$ represents the x-coordinate of the node whose sequence number is $j$ in $L$(the sequence number of the nodes in $L$ starts with 0)
10: $sum \leftarrow sum + (-1)^d-j * L(j+1).y - L(j).y$;
11: $t \leftarrow \frac{1}{2}((y_{i+1} - y_i) - (-1)^d+1 * k_0) - sum;$
12: $x_{i+0.5} \leftarrow \frac{(y_{i+1} - y_i) * (x_{i+1} - x_i) + x_h * (y_{i+1} - x_i)}{t * (y_{i+1} - y_i) + x_{i+1} - x_i};$
13: if $x_{i+0.5} \in (x_i, x_{i+1})$ then
14: $y_{i+0.5} \leftarrow y_i + t * (x_{i+0.5} - x_i);$
15: Add the node $(x_{i+0.5}, y_{i+0.5})$ to the end of the list $L$;
16: // Do the traditional interpolation for the transformed nodes
17: $d \leftarrow Length(L) - 1;$
18: for $i = 0 : d - 1$ do
19: if $L(i).x \leq x \leq L(i + 1).x$ then
20: return $L(i).y + \frac{(x - L(i+1).x)}{L(i).x - L(i+1).x} * L(i).y + \frac{(x - L(i).x)}{L(i+1).x - L(i).x} * L(i+1).y + \left(\sum_{j=0}^{i-1} (-1)^{i-j+1} * \left[\frac{L(j+1).y - L(j).y}{L(j+1).x - L(j).x}\right]^2 * \frac{2 * L(j).y}{L(j).x - L(i+1).x}\right)$;

References

[1] Herz-Fischer, R. A mathematical history of division in extreme and mean ratio. Wilfrid Laurier University Press; 1986.
[2] Olsen, S. The golden section: nature’s greatest secret. Bloomsbury Publishing USA; 2006.
[3] Ozturk, S, Kenan, Y, Yetkin, E. Golden ratio: A subtle regulator in our body and cardiovascular system? International Journal of Cardiology 2016:223:143–145.
[4] Shimog, M, Tobisa, M, Nakano, Y. An application of the golden ratio and fibonacci numbers to analyses of canopy structure of forage plants. Journal of the Faculty of Agriculture Kyushu University 2011:56(1):77–78.
[5] Xie, Z. The golden ratio and super central configurations of the n-body problem. Journal of Differential Equations 2011:251(1):58–72.
[6] Wang, RH, Chang, JC. Bivariate splines and golden section based on theory of elasticity. Journal of Mathematical Research with Applications 2011;31(1):1–11.

[7] Seghers, MJ, Longacre, JJ, Destefano, GA. The golden proportion and beauty. Plastic and Reconstructive Surgery 1964;34(4):382–386.

[8] Dio, CD, Macaluso, E, Rizzolatti, G. The golden beauty: Brain response to classical and renaissance sculptures. Plos One 2007;2(11):e1201.

[9] Green, CD. All that glitters: a review of psychological research on the aesthetics of the golden section. Perception 1995;24(6):937–968.

[10] Deuflhard, P, Hohmann, A. Interpolation and approximation. Springer New York; 2003.

[11] Phillips, GM. Interpolation and approximation by polynomials. Springer Science and Business Media; 2003.

[12] Schumaker, L. Spline functions : basic theory. Cambridge University Press; 2007.

[13] Farin, GE. Curves and surfaces for CAGD: a practical guide. Morgan Kaufmann; 2002.

[14] Schweikert, DG. An interpolation curve using a spline in tension. Journal of Mathematics and Physics 1966;45(1):33.

[15] Foley, TA. A shape preserving interpolant with tension controls. Computer Aided Geometric Design 1988;5(2):105–118.

[16] Celniker, G, Gossard, D. Deformable curve and surface finite-elements for free-form shape design. ACM SIGGRAPH Computer Graphics 1991;25(4):257–266.

[17] Nowacki, H, Lü, X. Fairing composite polynomial curves with constraints. Computer Aided Geometric Design 1994;11(1):1–15.

[18] Jaklič, G, Žagar, E. Planar cubic G1 interpolatory splines with small strain energy. Journal of Computational and Applied Mathematics 2011;235(8):2758–2765.

[19] Lu, L. Planar quintic g2 Hermite interpolation with minimum strain energy. Journal of Computational and Applied Mathematics 2015;274:109–117.

[20] Lu, L, Jiang, C, Hu, Q. Planar cubic G1 and quintic G2 Hermite interpolations via curvature variation minimization. Computers and Graphics 2018;70:92–98.

[21] Jaklič, G, Žagar, E. Curvature variation minimizing cubic Hermite interpolants. Applied Mathematics and Computation 2011;218(7):3918–3924.

[22] Harada, T. Study of quantitative analysis of the characteristics of a curve. Forma 1997;12(1):55–63.

[23] Miura, KT, Shibuya, D, Usuki, S, Lan, F, Tamai, H, Makino, H. G2 Hermite interpolation with the log-aesthetic curve. Journal of the Japan Society for Precision Engineering 2013;79(3):260–265.

[24] Meek, DS, Saito, T, Walton, DJ, Yoshida, N. Planar two-point \(G_2\) Hermite interpolating log-aesthetic spirals. Journal of Computational and Applied Mathematics 2012;236(17):4485–4493.

[25] Lu, L, Xiang, X. Quintic polynomial approximation of log-aesthetic curves by curvature deviation. Journal of Computational and Applied Mathematics 2016;296:389–396.

[26] Dietz, DA, Piper, B. Interpolation with cubic spirals. Computer Aided Geometric Design 2004;21(2):165–180.

[27] Habib, Z, Sakai, M. Admissible regions for rational cubic spirals matching G2 Hermite data. Computer-Aided Design 2010;42(12):1117–1124.

[28] Li, Z, Haddou, RA, Biard, L. Pythagorean hodograph spline spirals that match Hermite data from circles. Journal of Computational and Applied Mathematics 2015;278:162–180.

[29] Marciniak, K, Putz, B. Approximation of spirals by piecewise curves of fewest circular arc segments. Computer-Aided Design 1984;16(2):87–90.

[30] Cirillo, E, Kai, H. An iterative approach to barycentric rational Hermite interpolation. Numerische Mathematik 2018;138:9;1–24.

[31] Li, A. Convexity preserving interpolation. Computer Aided Geometric Design 1999;16(2):127–147.

[32] Laihinen, A. Shape preserving interpolation by quadratic splines. Journal of Computational and Applied Mathematics 1990;29(1):15–24.

[33] Schumaker, LL. On shape preserving quadratic spline interpolation. Siam Journal on Numerical Analysis 1983;20(4):854–864.