ORIGINAL ARTICLE   **Open Access**

Typical curve with G1 constraints for curve completion

Chuan He[1,2,3], Gang Zhao[1,3], Aizeng Wang[1,3*], Fei Hou[4,5], Zhanchuan Cai[6] and Shaolin Li[2]

Abstract

This paper presents a novel algorithm for planar G1 interpolation using typical curves with monotonic curvature. The G1 interpolation problem is converted into a system of nonlinear equations and sufficient conditions are provided to check whether there is a solution. The proposed algorithm was applied to a curve completion task. The main advantages of the proposed method are its simple construction, compatibility with NURBS, and monotonic curvature.

Keywords: Typical curves, Monotonic curvature, G1 interpolation, Curve completion, Euler spiral

Introduction

G1 interpolation is an essential problem in many applications such as path planning and curve completion. The goal is to find a transition curve that matches the positions and associated unit tangent vectors at two given endpoints [1]. Curve completion aims to find a pleasing contour to fill in an object boundary that is partially occluded [2, 3]. This process differs significantly for humans and computers. For example, the visual system of humans can automatically complete missing parts of curves (Fig. 1), which is called visual curve completion [4–6]. For computers, it is necessary to identify a fairing curve matching "boundary conditions" among potential transition curves, which is called curve completion.

In computer-aided design (CAD) and computer-aided geometric design, the measurement of the fairness of curves typically depends on the curvature distribution. Farin and Sapidis [7] assumed that a

fairing curve should have relatively few monotonic curvature variation segments. By virtue of linear curvature variation and the ability to minimize changes in the total curvature, the Euler spiral is considered to be the most pleasing curve for shape completion [2], so it has been widely used in various applications such as path planning and curve completion [2, 8–11]. However, the Euler spiral is defined in the form of a transcendental function that is computationally intensive and inefficient. Furthermore, the Euler spiral is not compatible with the NURBS methods, which are the standard methods in existing CAD software [12].

To overcome these shortcomings, some methods using polynomial curves or arc splines to approximate Euler curves have been proposed [3, 13–16]. Another solution is to use a polynomial curve as a design object directly. In highway design, a transition curve is typically defined between a line and a circular curve or between two circular curves. Walton et al. [17–19] used planar cubic Bézier spirals to construct transition curves with G2 continuity for
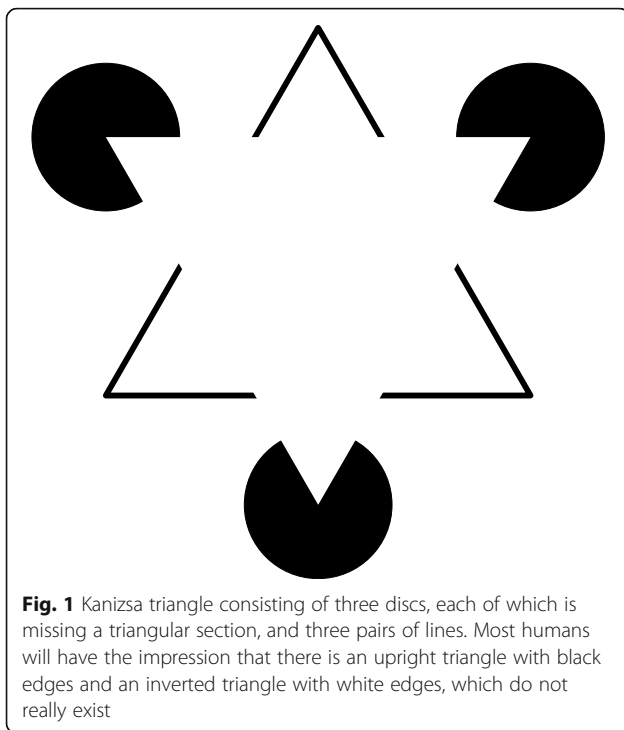
* Correspondence: azwang@buaa.edu.cn
[1]School of Mechanical Engineering and Automation, Beihang University, Beijing 100191, China
[3]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China
Full list of author information is available at the end of the article

Fig. 1 Kanizsa triangle consisting of three discs, each of which is missing a triangular section, and three pairs of lines. Most humans will have the impression that there is an upright triangle with black edges and an inverted triangle with white edges, which do not really exist

different boundary constraints and the results of refs. [17, 19] were extended in refs. [20, 21]. Traditional methods for designing Bézier curves with monotonic curvature focused on adjusting the positions of control points until Mineur proposed the concept of a typical curve [22], which indicates a special subset of planar Bézier curves with control edges that maintain a specific geometric constraint. Sánchez-Reyes [23] pointed out that these constrained typical curves belong to the family of offset rational sinusoidal spirals and can also be expressed as a subset of rational Bézier curves called p-Bézier curves [24]. Recently, Wang et al. [25] proposed a geometric proof of the necessary and sufficient conditions for the curvature monotonicity of typical curves. In 2006, Farin [26] extended typical curves to 3D class-A Bézier curves, whose control edges were generated from a class-A matrix and initial control edge. However, the conditions of the class-A matrix have been proven to be incomplete [27] and several counterexamples have been reported [28]. Yoshida et al. [29] proposed an interactive method to generate general class-A Bézier curves by perturbing the elements of a planar typical class-A matrix. Furthermore, typical curves converge to logarithmic spiral segments as the degree is elevated and the drawable regions of typical

curves for boundary constraints under different degrees can be obtained.

In recent years, geometric continuity has also been discussed based on trigonometric Bézier curves with shape parameters [30–33]. Generalized trigonometric Bézier curves are flexible and can achieve G2 continuity. However, these methods are more complex than polynomials and have not provided a way to obtain a monotonic variation of curvature. This study analyzed the sufficient conditions for the G1 interpolation problem using typical curves and developed a novel algorithm to find a fairing solution, which is applied to curve completion. For a given set of constraints, multiple solutions may exist, so a suitable criterion must be defined to find the optimal solution automatically. Furthermore, a designer can modify data manually to obtain a fair solution to fit different situations. The main contributions of this work can be summarized as follows. The sufficient conditions for G1 interpolation based on typical curves are provided and curve completion using Bézier curves with monotonic curvature is realized.

The remainder of this paper is organized as follows. Typical curves Section introduces typical curves comprehensively. Methods Section discusses the G1 interpolation problem, provides sufficient conditions for a typical curve solution, and presents a novel algorithm to solve G1 boundary constraints. In Results and discussion Section, some examples are presented to demonstrate the practicability and superiority of the proposed algorithm. The method is then applied in the application of curve completion. Finally, this paper is concluded in Conclusions Section.
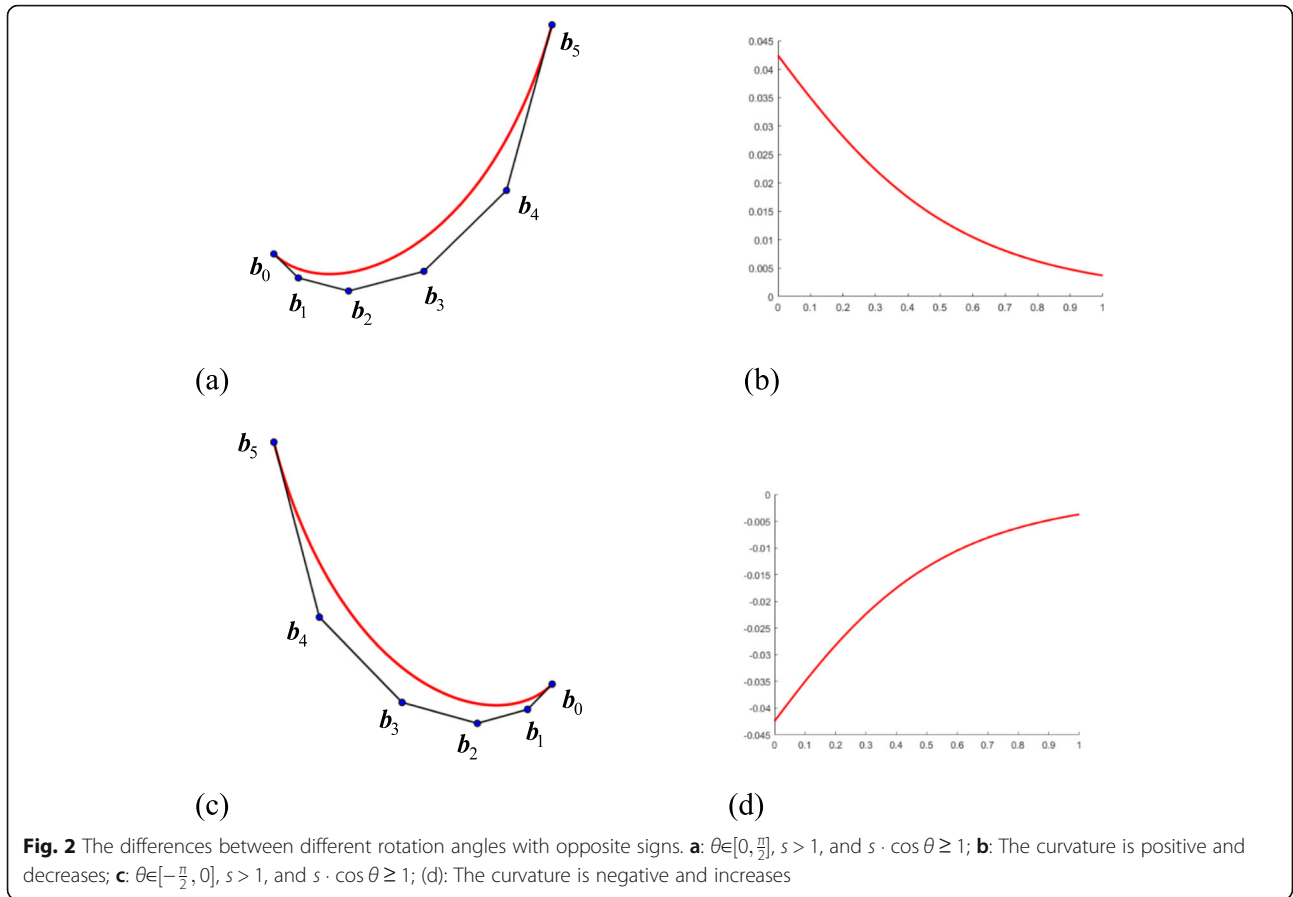
Typical curves

A planar Bézier curve of degree k can be expressed as

$$\boldsymbol{P}(t) = \sum_{i=0}^{k} \boldsymbol{b}_i B_{i,k}(t), t \in [0,1] \tag{1}$$

where $\boldsymbol{b}_i$ are the two-dimensional control points and $B_{i,k}(t)$ is the i-th Bernstein polynomial of degree k. Let the forward difference vector $\boldsymbol{V}_i = \boldsymbol{b}_{i+1} - \boldsymbol{b}_i$ ($i = 0, ..., k-1$) be the control edge. Then, a typical curve is obtained with control edges satisfying

$$\boldsymbol{V}_i = s \cdot \boldsymbol{R}_\theta \cdot \boldsymbol{V}_{i-1} \tag{2}$$

where s is a positive scale factor and $\boldsymbol{R}_\theta$ is a second-order rotation matrix with a rotation angle $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$. A k-degree typical curve exhibits monotonic curvature variation if and only if
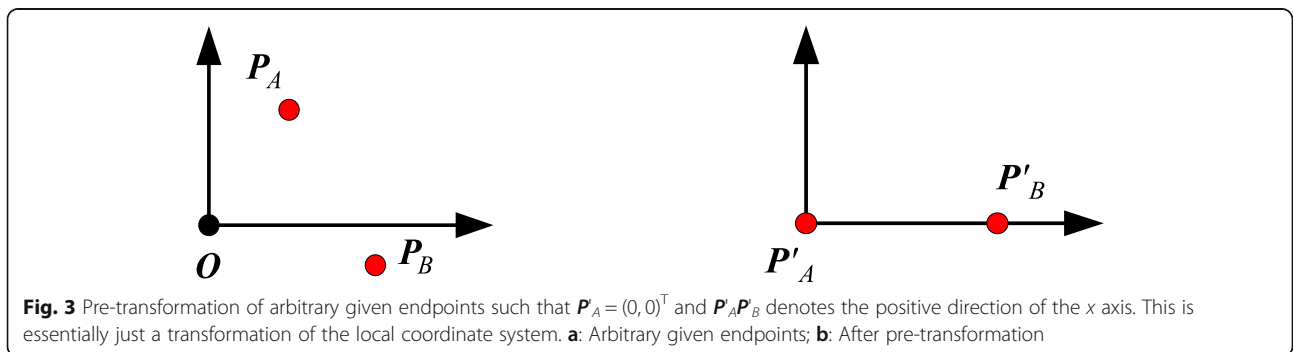
Fig. 2 The differences between different rotation angles with opposite signs. **a**: $\theta \in [0, \frac{\pi}{2}]$, $s > 1$, and $s \cdot \cos \theta \geq 1$; **b**: The curvature is positive and decreases; **c**: $\theta \in [-\frac{\pi}{2}, 0]$, $s > 1$, and $s \cdot \cos \theta \geq 1$; (d): The curvature is negative and increases

$$\begin{cases} s \cdot \cos\theta \geq 1, s \geq 1 \\ s \leq \cos\theta, 0 < s < 1 \end{cases} \tag{3}$$

When $\theta \in [0, \frac{\pi}{2}]$, it is assumed that the control edge rotates counterclockwise (Fig. 2(a)) and the relative curvature is assumed to be positive (Fig. 2(b)). In this case, if the scale factor $s > 1$, then the curve $P(t)$ has a monotonically decreasing curvature. If $0 < s < 1$, then $P(t)$ has a monotonically increasing curvature. When $\theta \in [-\frac{\pi}{2}, 0]$, it is assumed that the control edge rotates clockwise (Fig. 2(c)) and the relative curvature is negative (Fig. 2(d)), so the monotonicity of curvature is opposite to $\theta \in [0, \frac{\pi}{2}]$. As a result of the construction process of control edge vectors, the generated curve can only be 'C' shaped rather than 'S' shaped.

Methods

This section presents the G1 interpolation algorithm based on typical curves. First, the G1 interpolation problem based on typical curves is introduced briefly. The given endpoint-orientation pairs are then divided into

Fig. 3 Pre-transformation of arbitrary given endpoints such that $P'_A = (0, 0)^\top$ and $P'_A P'_B$ denotes the positive direction of the x axis. This is essentially just a transformation of the local coordinate system. **a**: Arbitrary given endpoints; **b**: After pre-transformation
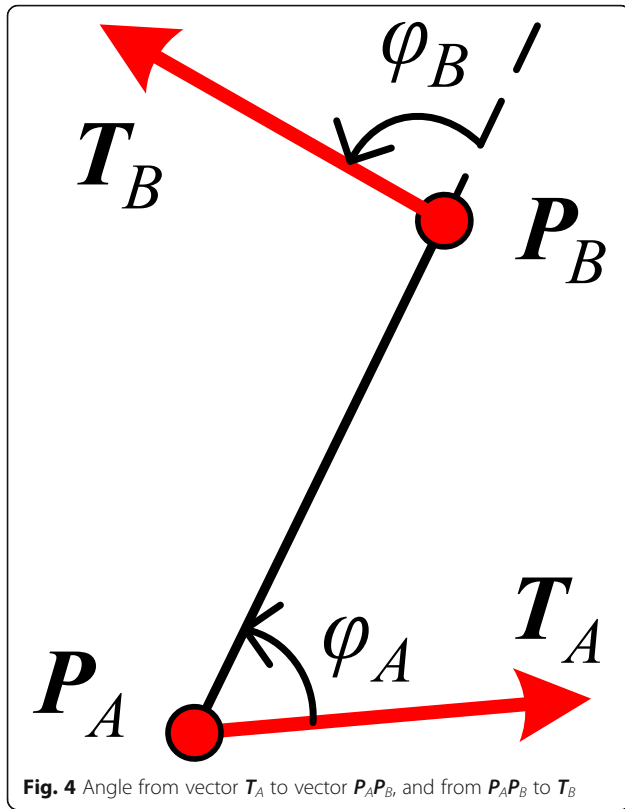
Fig. 4 Angle from vector T_A to vector P_AP_B, and from P_AP_B to T_B

two cases, and a novel algorithm for constructing a typical curve under the given boundary constraints is proposed.

G1 Hermite interpolation problem

In many interpolation problems, not only positions, but also the corresponding derivative values or higher-order derivatives at endpoints are required to be equal, which is called Hermite interpolation. If the interpolation curve matches the given unit tangent vectors at the two endpoints, it is a two-point G1 Hermite interpolation. Additionally, if the given curvatures at the two endpoints are also matched, it is called two-point G2 Hermite interpolation [1].

Although G2 interpolation has better continuity, it also introduces stronger constraints for the interpolation curve and the high-order continuity at endpoints is sometimes difficult to guarantee. In some applications, G1 interpolation is preferable to G2 interpolation when considering cost effectiveness, such as curve completion or path planning using Euler spirals [1]. Consider two different points $P_A = (x_A, y_A)^{\mathrm{T}}$ and $P_B = (x_B, y_B)^{\mathrm{T}}$ with the corresponding unit tangent vectors $T_A = (\cos\theta_A, \sin\theta_A)^{\mathrm{T}}$ and $T_B = (\cos\theta_B, \sin\theta_B)^{\mathrm{T}}$, where the three vectors T_A, T_B, and $P_AP_B = P_B - P_A$ are not all parallel. G1 Hermite interpolation is used to find a fairing curve that joins P_A and P_B on the condition that the tangent vectors at the endpoints match T_A and T_B, respectively.

Walton et al. [1, 34] proposed an improved Euler spiral algorithm for shape completion. However, the Euler spiral is not compatible with NURBS. Therefore, the paper proposes a more intuitive algorithm for the G1 interpolation problem based on a typical curve, which is easier to construct.

G1 interpolation problem based on a typical curve
Proposition 1

Consider the positions of two endpoints $P_A = (x_A, y_A)^{\mathrm{T}}$ and $P_B = (x_B, y_B)^{\mathrm{T}}$, and their associated unit tangent vectors $T_A = (\cos\theta_A, \sin\theta_A)^{\mathrm{T}}$ and $T_B = (\cos\theta_B, \sin\theta_B)^{\mathrm{T}}$, where θ_A and θ_B are orientation angles with multiple values. When θ_A or θ_B are increased or reduced by $2m \cdot \pi (m \in \mathbb{Z})$, the unit tangent vector remains
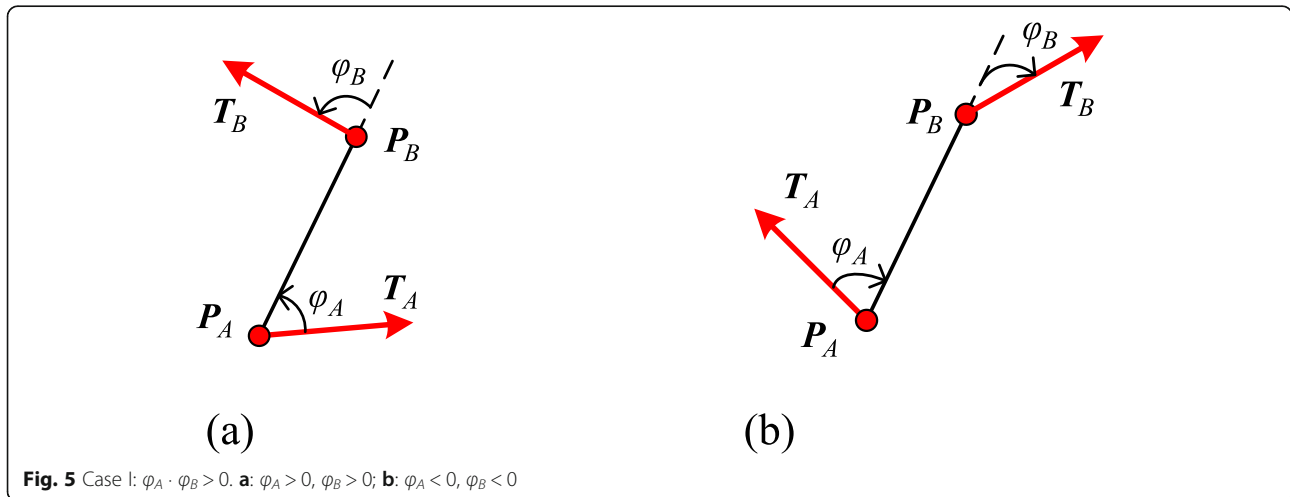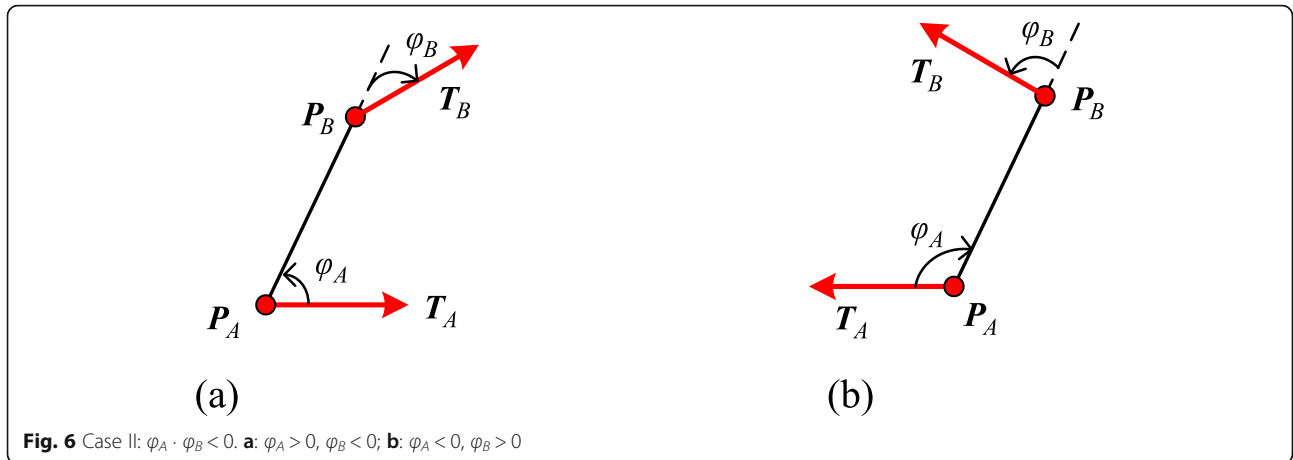

Fig. 5 Case I: $\varphi_A \cdot \varphi_B > 0$. **a**: $\varphi_A > 0$, $\varphi_B > 0$; **b**: $\varphi_A < 0$, $\varphi_B < 0$

Fig. 6 Case II: $\varphi_A \cdot \varphi_B < 0$. **a**: $\varphi_A > 0$, $\varphi_B < 0$; **b**: $\varphi_A < 0$, $\varphi_B > 0$

unchanged. The sufficient condition for the existence of a k-degree typical curve matching the boundary constraints is that the following equation has real number solutions satisfying $s \cdot \cos\theta \geq 1$ $(s \geq 1)$ or $s \leq \cos\theta$ $(0 < s < 1)$:

$$
\begin{cases}
x_A + \sum_{i=1}^{k} \|V_0\| \cdot s^{i-1} \cdot \cos[\theta_A + (i-1) \cdot \theta] = x_B \\
y_A + \sum_{i=1}^{k} \|V_0\| \cdot s^{i-1} \cdot \sin[\theta_A + (i-1) \cdot \theta] = y_B
\end{cases}
\tag{4}
$$

where $\|V_0\|$ and s are free variables with positive values and $\theta = (\theta_B - \theta_A)/(k-1)$.

Proof

When the degree k is fixed, $\theta = (\theta_B - \theta_A)/(k-1)$ is the rotation angle of the typical curve. If there are real number solutions such that $s \cdot \cos\theta \geq 1$ $(s \geq 1)$ or $s \leq \cos\theta$ $(0 < s < 1)$,

which are the sufficient and necessary conditions for the monotonic curvature of a typical curve, the curvature of the Bézier curve constructed using this method must be monotonous.

Now we are going to analyze the sufficient conditions for the existence of solutions for Eq. (4). Considering the degree k as a free variable in Eq. (4), there are three free variables $k \in Z^+$, $s > 0$, and $\|V_0\| > 0$. Eliminating $\|V_0\|$ yields

$$
\sum_{i=1}^{k} s^{i-1} \cdot [(y_B - y_A) \cdot \cos(\theta_A + (i-1) \cdot \theta) - (x_B - x_A) \cdot \sin(\theta_A + (i-1) \cdot \theta)] = 0 \tag{5}
$$

Because Bézier curves are invariant under affine transformations, one can place the first endpoint $P_A = (x_A, y_A)^T$ at the origin and the second endpoint $P_B = (x_B, y_B)^T$ at the positive half of the x axis using a pretransformation, which leads to $x'_A = 0$, $y'_A = 0$, $x'_B > 0$ and $y'_B = 0$ (Fig. 3). Then, Eq. (5) is converted into
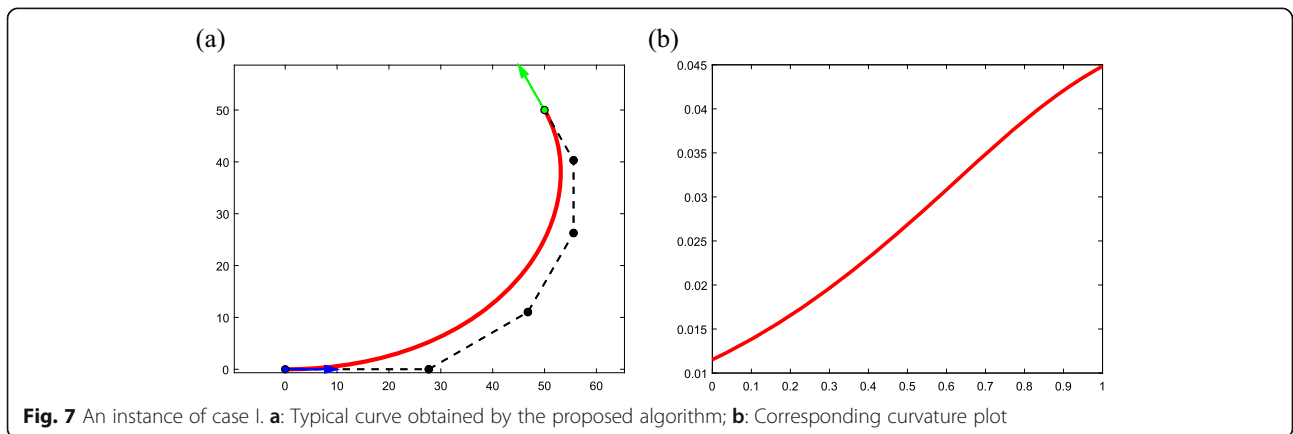

Fig. 7 An instance of case I. **a**: Typical curve obtained by the proposed algorithm; **b**: Corresponding curvature plot

Table 1 The specific data of typical curve for Example 1

k	s	$\|V_0\|$	θ
5	0.797364	27.680619	$\pi/6$

$$\sum_{i=1}^{k} s^{i-1} \cdot [-x'_B \cdot \sin(\theta'_A + (i-1) \cdot \theta)] = 0 \qquad (6)$$

The superscript ' indicates a new variable obtained from the pre-transformation. The pre-transformation can be used to construct a new local coordinate system. Equation (6) is a polynomial of degree k and one can infer the number of positive roots from the Descartes rule of signs [35]. The number of sign changes in the coefficients is equal to the number of sign changes in $\sin[\theta'_A + (i-1) \cdot \theta]$, $i = 1, ..., k$. Therefore, a sufficient condition to guarantee that Eq. (6) has at least one positive solution is that $\sin[\theta'_A + (i-1) \cdot \theta]$ changes signs an odd number of times, meaning that after pre-transformation, the sign of the sine angle for all control edges changes an odd number of times.

Angle sign change condition

Consider P_A as the origin and $P_A P_B$ as the positive direction of the x axis. In this new local coordinate system, the function $\sin\alpha$ ($\theta'_A \leq \alpha \leq \theta'_B$) only changes its sign an odd number of times.

Proposition 2

If the ASC condition is satisfied, then one can always find an appropriate degree k such that the solutions of Eq. (4) or (6) satisfy Eq. (3).

Proof

Because the affine transformation does not change the solution of the curve, Eqs. (4, 6) have the same solution and the ASC condition guarantees a positive solution for Eq. (4). When the degree k gradually increases in Eq. (6) and the rotation angle $\theta = \frac{\theta_B - \theta_A}{k-1} \to 0$ $\cos\theta \to 1$, there is an $s_k > 0$ such that $s_k \cdot \cos\theta \geq 1$ ($s_k \geq 1$) or $s_k \leq \cos\theta$ ($0 < s_k < 1$), meaning the solutions of Eq. (4) or (6) satisfy Eq. (3).

Corollary 1

For Eq. (4) or (6), if there is a typical curve satisfying the constraints when $k = m \geq 2$, then there are also typical curve solutions when $k \geq m + 1$. This means that the G1 interpolation problem has multiple solutions consisting of typical curves and the limit of the solutions increases $\theta \to 0$ with an increase in the degree k.

The ASC condition is a sufficient condition for the positive solution of Eq. (6). Even if the ASC condition is not satisfied, Eq. (4) or (6) may still have a positive root. In contrast, the ASC condition cannot guarantee solutions satisfying Eq. (3) under a fixed degree.

Two cases of G1 constraints

For convenience of expression, homogeneous coordinates will be used to represent points and vectors for the remainder of this paper. For a 2D point $(x, y)^T$, its homogeneous coordinate can be written as $(\omega x, \omega y, \omega)^T$, where $\omega \neq 0$. This study used $\omega = 1$. For a 2D vector $(x, y)^T$, the homogeneous coordinate can only be expressed as $(x, y, 0)^T$.

For an arbitrary $P_A = (x_A, y_A, 1)^T$, $P_B = (x_B, y_B, 1)^T$, $T_A = (\cos\theta_A, \sin\theta_A, 0)^T$, and $T_B = (\cos\theta_B, \sin\theta_B, 0)^T$, one can divide the G1 constraints into two cases according to the relationship between the relative positions of the three vectors $T_A = (\cos\theta_A, \sin\theta_A, 0)^T$, $T_B = (\cos\theta_B, \sin\theta_B, 0)^T$, and $P_A P_B = (x_B - x_A, y_B - y_A, 0)^T$. This analysis does
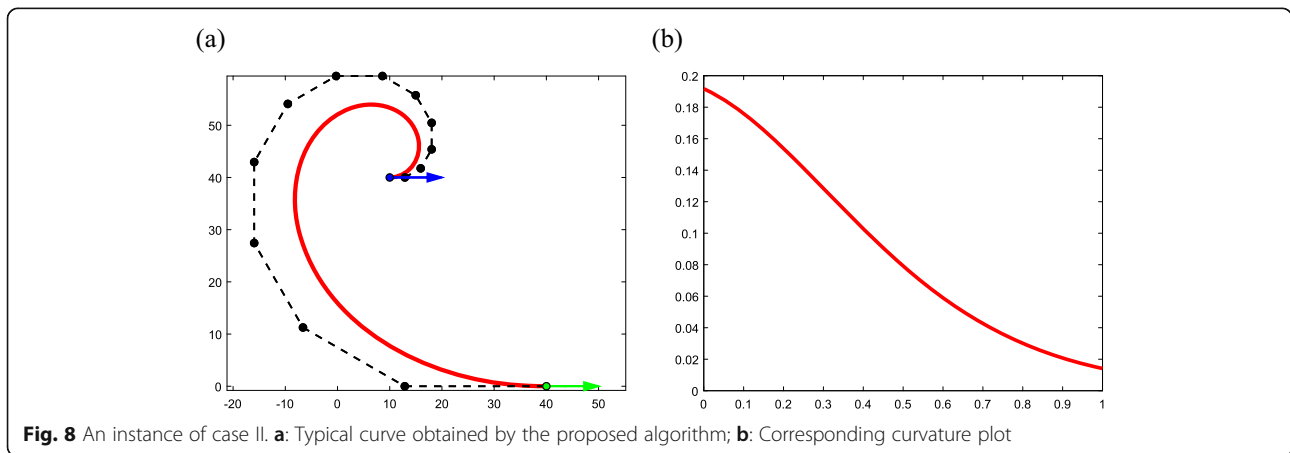

Fig. 8 An instance of case II. **a**: Typical curve obtained by the proposed algorithm; **b**: Corresponding curvature plot

not consider the degenerate case in which all three vectors are parallel.

Let $-\pi \le \phi_A \le \pi$ denote the angle from vector $\boldsymbol{T}_A$ to vector $\boldsymbol{P}_A\boldsymbol{P}_B$ (Fig. 4). The sign of ϕ_A indicates the direction and $\phi_A > 0$ indicates that $\boldsymbol{T}_A$ is on the right side of $\boldsymbol{P}_A\boldsymbol{P}_B$. Similarly, $-\pi \le \phi_B \le \pi$ denotes the angle from vector $\boldsymbol{P}_A\boldsymbol{P}_B$ to vector $\boldsymbol{T}_B$.

Considering the signs of these two angles, one can divide all potential constraints into two cases as follows:

Case I $\phi_A \cdot \phi_B > 0$, meaning $\boldsymbol{T}_A$ and $\boldsymbol{T}_B$ are located on opposite sides of $\boldsymbol{P}_A\boldsymbol{P}_B$ (Fig. 5).

Case II $\phi_A \cdot \phi_B < 0$, meaning that $\boldsymbol{T}_A$ and $\boldsymbol{T}_B$ are located on the same side of $\boldsymbol{P}_A\boldsymbol{P}_B$ (Fig. 6).

Because θ_A and θ_B can be increased or decreased by $2m \cdot \pi$ $(m \in \mathbb{Z})$ while keeping the unit tangent vectors $\boldsymbol{T}_A$ and $\boldsymbol{T}_B$ unchanged, the rotation angle $\theta = (\theta_B - \theta_A)/(k-1)$ can be either positive or negative, and the labeling of the given endpoint-orientation pairs can be swapped. Based on these features, there are different constructions for typical curves under the given conditions. This paper presents a practical method for choosing a suitable curve based on the position relationship between the unit tangent vectors $\boldsymbol{T}_A$ $\boldsymbol{T}_B$ and the vector $\boldsymbol{P}_A\boldsymbol{P}_B$.

Proposed algorithm

Given two endpoint-orientation pairs, the G1 interpolation algorithm is used to find the optimal solution to Eq. (4), where the degree $k \ge 2$ is an integral and the rotation angle θ may have multiple values. $s > 0$ and $\|V_0\| > 0$ are the variables to be determined. The value of k can be increased from two to a set maximum value k_{max}. For a fixed k, one must determine the value of $\theta = (\theta_B - \theta_A)/(k-1)$, which is equivalent to determining $\Delta\theta = \theta_B - \theta_A$. The proposed algorithm provides a rule for determining $\Delta\theta$ automatically. For case I, let $\Delta\theta = \phi_A + \phi_B$. For case II, let $\Delta\theta = \phi_A + \phi_B - \text{sign}(\phi_A) \cdot 2\pi$, where $\text{sign}(\phi_A)$ is the sign of ϕ_A. In this manner, the algorithm can find the typical curve with the lowest degree and smallest total angle between $\boldsymbol{T}_A$ and $\boldsymbol{T}_B$.

Once k and $\Delta\theta$ are determined, one can use an optimization process to obtain s and $\|V_0\|$ such that

$$f(s, \|V_0\|) = \left\{ x_A + \sum_{i=1}^{k} \|V_0\| \cdot s^{i-1} \cdot \cos[\theta_A + (i-1) \cdot \theta] - x_B \right\}^2 + \left\{ y_A + \sum_{i=1}^{k} \|V_0\| \cdot s^{i-1} \cdot \sin[\theta_A + (i-1) \cdot \theta] - y_B \right\}^2 \tag{7}$$

becomes zero. If $s > 0$, $\|V_0\| > 0$, and Eq. (3) is satisfied, then a typical curve segment can be constructed using

these values. Otherwise, the steps of the iteration should be repeated by increasing the degree k.

The pseudo code for the proposed algorithm is presented in Algorithm 1.

Algorithm 1 G1 interpolation based on typical curves

Input: $\boldsymbol{P}_A$, $\boldsymbol{P}_B$, $\boldsymbol{T}_A$, $\boldsymbol{T}_B$

Output: Typical curve with degree k matching G1 Hermite data.

$\boldsymbol{P}_A\boldsymbol{P}_B = \boldsymbol{P}_B - \boldsymbol{P}_A$, $\boldsymbol{b}_0 = \boldsymbol{P}_A$, isSuccess = FALSE;

φ_A = angle from $\boldsymbol{T}_A$ to $\boldsymbol{P}_A\boldsymbol{P}_B$;

φ_B = angle from $\boldsymbol{P}_A\boldsymbol{P}_B$ to $\boldsymbol{T}_B$;

$\Delta\theta = \varphi_A + \varphi_B$;

IF $\boldsymbol{T}_A$ and $\boldsymbol{T}_B$ locate on the same side of $\boldsymbol{P}_A\boldsymbol{P}_B$ THEN

    $\Delta\theta = \Delta\theta - \text{sign}(\varphi_A) \cdot 2\pi$;

END IF

FOR $k = 2 : k_{max}$ DO

    $\theta = \Delta\theta / (k-1)$;

    $f = \left\| \boldsymbol{P}_A + \sum_{i=1}^{k} \|V_0\| \cdot s^{i-1} \cos[\theta_A + (i-1) \cdot \theta] - \boldsymbol{P}_B \right\|^2$;

    Minimize f such that $f \le \varepsilon$ to find s and $\|V_0\|$;

    IF $f > \varepsilon$ THEN

        CONTINUE;

    END IF

    IF s and $\cos\theta$ satisfy $s \cdot \cos\theta \ge 1$ ($s \ge 1$) or $s < \cos\theta$ ($0 < s < 1$) THEN

        save the current k, s, $\|V_0\|$, θ;

        isSuccess = TRUE;

        BREAK

    END IF

END FOR

IF isSuccess == TRUE THEN

    $V_0 = \|V_0\| \cdot \boldsymbol{T}_A$;

    $\boldsymbol{R} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$;

    FOR $i = 1 : k-1$ DO

        $V_i = s \cdot \boldsymbol{R} \cdot V_{i-1}$;

    END FOR

    FOR $i = 1 : k$ DO

        $\boldsymbol{b}_i = V_{i-1} + \boldsymbol{b}_{i-1}$;

    END FOR

END IF

Construct Bézier curve by control points $\boldsymbol{b}_i$, $i = 0, \ldots, k$.

The algorithm first calculates an angle $\Delta\theta$, which is determined by the position relationship between, $\boldsymbol{T}_A$, $\boldsymbol{T}_B$, and $\boldsymbol{P}_A\boldsymbol{P}_B$. Then, Eq. (7) is minimized to a value of zero iteratively by the optimization process. This step was implemented using the interior point method, which can find the minimum value of a constrained nonlinear multivariable function. For the obtained $s > 0$ and $\|V_0\| > 0$, the algorithm checks whether Eq. (3) is satisfied. Once

Table 2 The specific data of typical curve for Example 2

k	s	‖V_0‖	θ
13	1.204772	2.900292	π/6

Table 3 The specific data of typical curves with different degree

Color	k	s	‖V_0‖	θ
Red	6	1.264905	8.780173	π/5
Purple	9	1.189868	4.753024	π/8
Carmine	20	1.090728	1.704246	π/19

the solutions satisfy Eq. (3), iteration can be stopped and a typical curve can be generated from the obtained data.

The algorithm proposed above avoids outputting multiple solutions by choosing an appropriate rotation angle $\theta = (\theta_B - \theta_A)/(k-1)$, but this does not mean that there is only one typical curve solution under the given constraints. The main principle of the proposed algorithm is to select the minimum rotation angle $\Delta\theta = \theta_B - \theta_A$ under the given conditions and then choose the lowest degree k to satisfy the constraints. Additional solutions can be obtained by increasing the value of the degree k, modifying the angle value $\Delta\theta = \theta_B - \theta_A$ while leaving the unit tangents T_A and T_B unchanged, or even by exchanging the labels of P_A and P_B or T_A and T_B. These methods are discussed in detail in Results and discussion Section.

Results and discussion

This section presents some numerical experiments based on the proposed algorithm and discusses the existence of multiple solutions.

Examples of two cases of constraints

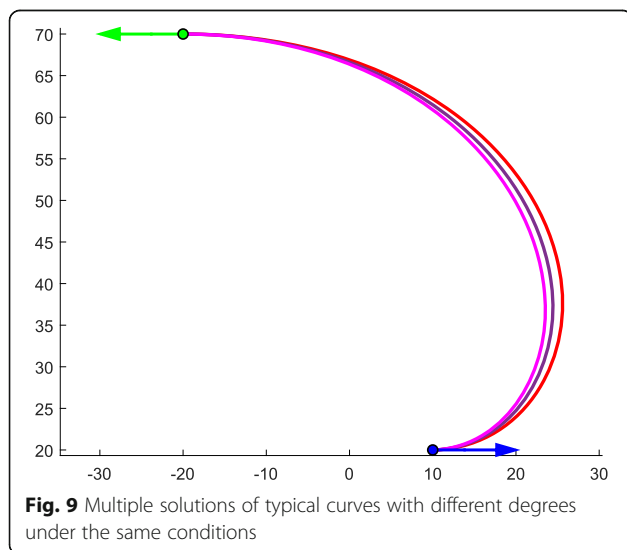This section presents different results for two cases of constraints.

Example 1

Consider data $P_A = (0, 0, 1)^{\mathrm{T}}$, $\theta_A = 0$, $T_A = (1, 0, 0)^{\mathrm{T}}$, $P_B = (50, 50, 1)^{\mathrm{T}}$, $\theta_B = \frac{2\pi}{3}$, and $T_B = \left(-\frac{1}{2}, \frac{\sqrt{3}}{2}, 0\right)^{\mathrm{T}}$ (Fig. 7). It is easy to verify that this is an instance of case I that satisfies the ASC condition. In the proposed algorithm, $\Delta\theta = \theta_B - \theta_A$ is chosen as $\Delta\theta = \frac{2\pi}{3}$ and a solution is obtained when the degree $k = 5$ (Fig. 7). Figure 7(a) presents the boundary constraints and resulting curve, where P_A and T_A are marked in blue, and P_B and T_B are marked in green. Figure 7(b) presents an increasing curvature plot. The data obtained from the typical curve are presented in Table 1.

Example 2

Consider data $P_A = (10, 40, 1)^{\mathrm{T}}$, $\theta_A = 0$, $T_A = (1, 0, 0)^{\mathrm{T}}$, $P_B = (40, 0, 1)^{\mathrm{T}}$, $\theta_B = 0$, and $T_B = (1, 0, 0)^{\mathrm{T}}$ (Fig. 8). This is an instance of case II, where $\Delta\theta$ is chosen as $\Delta\theta = 2\pi$ and one solution is found when $k = 13$ (Fig. 8). The resulting curve satisfies $s > 1$ (Fig. 8(a)), so its curvature plot decreases (Fig. 8(b)). The specific data are presented in Table 2.

Discussion of multiple solutions

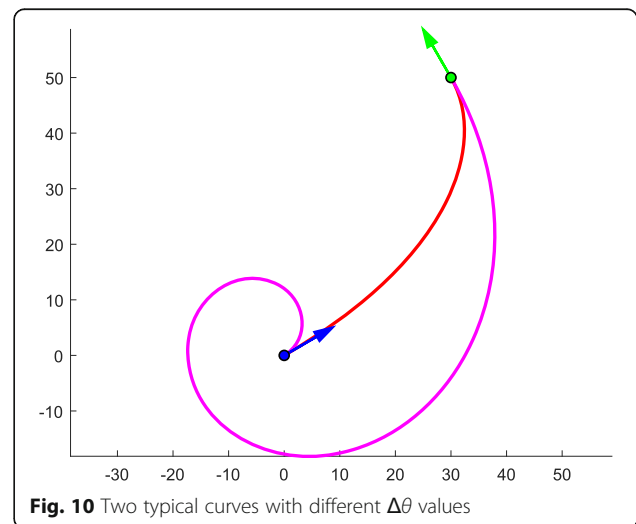Under the given constraints, there are three main reasons for generating multiple solutions: an increase in the

Fig. 9 Multiple solutions of typical curves with different degrees under the same conditions

Fig. 10 Two typical curves with different Δθ values

Table 4 The specific data of typical curves with different $\Delta\theta$

Color	$\Delta\theta$	k	s	$\|V_0\|$	θ
Red	$\pi/2$	3	0.604212	35.719770	$\pi/4$
Carmine	$5\pi/2$	29	1.081436	1.759416	$5\pi/56$

Table 5 The specific data of typical curve by exchanging the two endpoint-orientation pairs

Color	$\Delta\theta$	k	s	$\|V_0\|$	θ
Red	$-5\pi/6$	8	1.091423	8.905500	$-5\pi/42$
Carmine	$-7\pi/6$	17	1.026836	6.522758	$-7\pi/96$

degree k, different chosen values of $\Delta\theta$, and the exchange of labels of two endpoint-orientation pairs. Examples of these cases are discussed below.

Increasing the degree k

According to Corollary 1, multiple solutions with different degrees exist. Figure 9 presents three typical Bézier curves with different degrees k under the given boundary conditions, where $\boldsymbol{P}_A = (10, 20, 1)^T$, $\theta_A = 0$, $\boldsymbol{T}_A = (1, 0, 0)^T$, $\boldsymbol{P}_B = (-20, 70, 1)^T$, $\theta_B = \pi$, and $\boldsymbol{T}_B = (-1, 0, 0)^T$. The three curves in different colors represent three typical curves with different degrees k. With an increase in k, the arguments of the typical curves trend toward the limits $s \to 1$ and $\theta \to 0$ (Table 3).

Different values of $\Delta\theta$

The proposed algorithm provides a criterion for the determination $\Delta\theta$. This is a practical rule. In fact, one can add $2m \cdot \pi (m \in \mathbb{Z})$ to the tangent vector angle while keeping the tangent direction unchanged. Therefore, one can obtain new solutions that satisfy G1 interpolation by adding $2m \cdot \pi (m \in \mathbb{Z})$ to the $\Delta\theta$ selected by the previously proposed algorithm. Figure 10 presents two typical curves with different $\Delta\theta$, both meeting the given conditions of $\boldsymbol{P}_A = (0, 0, 1)^T$, $\theta_A = \frac{\pi}{6}$, $\boldsymbol{T}_A = (\frac{\sqrt{3}}{2}, \frac{1}{2}, 0)^T$, $\boldsymbol{P}_B = (30, 50, 1)^T$, $\theta_B = \frac{2\pi}{3}$, and $\boldsymbol{T}_B = (-\frac{1}{2}, \frac{\sqrt{3}}{2}, 0)^T$. The $\Delta\theta$

Fig. 11 Different typical curves with the exchange of two endpoint-orientation pairs

value of the carmine curve is 2π greater than that of the red curve (Table 4).

Exchange of two endpoint-orientation pairs

The construction process of typical curves depends on the sequence of endpoints. If one exchanges the labels of endpoint-orientation pairs, one can obtain different solutions that meet the given conditions. As shown in Fig. 11, the red and carmine curves are two different typical curves that match the same G1 constraint. The difference between the two results is that the red curve takes the blue endpoint-orientation as the starting point and the green endpoint-orientation as the ending point. This means that $\boldsymbol{P}_A = (10, 0, 1)^T$, $\theta_A = \frac{5\pi}{6}$, $\boldsymbol{T}_A = (-\frac{\sqrt{3}}{2}, \frac{1}{2}, 0)^T$, $\boldsymbol{P}_B = (40, 60, 1)^T$, $\theta_B = 0$, $\boldsymbol{T}_B = (1, 0, 0)^T$, and $\Delta\theta = -\frac{5\pi}{6}$. The labels of the carmine curve are reversed, meaning $\boldsymbol{P}_A = (40, 60, 1)^T$, $\theta_A = 0$, $\boldsymbol{T}_A = (1, 0, 0)^T$, $\boldsymbol{P}_B = (10, 0, 1)^T$, $\theta_B = \frac{5\pi}{6}$, $\boldsymbol{T}_B = (-\frac{\sqrt{3}}{2}, \frac{1}{2}, 0)^T$, and $\Delta\theta = -\frac{7\pi}{6}$. The specific data for these two typical curves are presented in Table 5.

Curve completion

Curve completion is a geometric continuation of the boundaries of objects that are temporarily interrupted by occlusion [2]. Curve completion requires a visually pleasing shape and is similar to G1 interpolation, but more expansive. The principles for finding an optimal completion curve include minimizing the total curvature square [36] and minimizing the total curvature variation [2]. In this section, these rules are simplified to realize curve completion by achieving monotonic curvature. Two sets of experimental results are presented to demonstrate the application of the proposed method to curve completion and discuss the differences between the proposed approach and other interpolation methods.

Figure 12(a) presents a banana that is partially covered by a piece of tape. Figure 12(b) presents the completed curve drawn by the proposed method and Fig. 12(c) presents a local zoomed-in view of the completed curve. Figure 12(d) presents the increasing curvature.

Figure 13(a) presents two overlapping leaves, one of which is partially occluded by the other. The proposed method completes the occluded boundary using a typical curve, as shown in Fig. 13(b). The corresponding monotonic curvature is presented in Fig. 13(c).
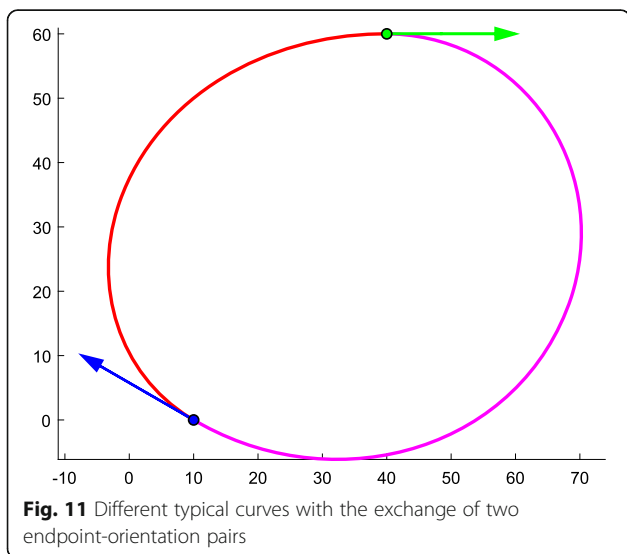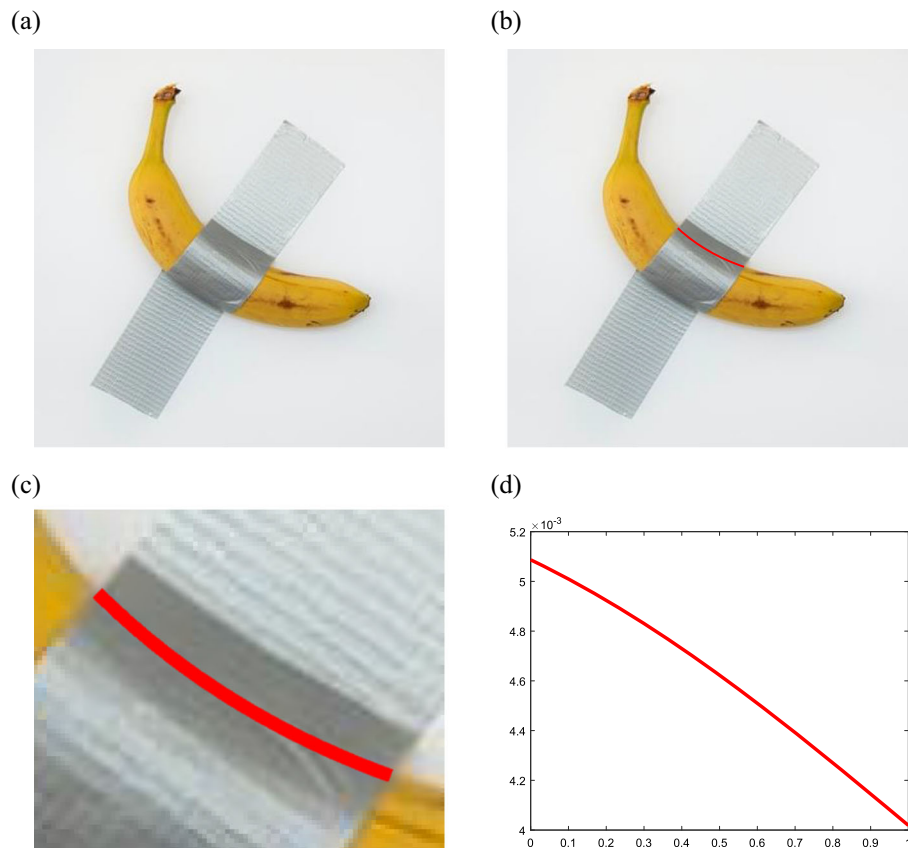
Fig. 12 Banana border completion. **a**: Partially occluded banana; **b**: Typical curve; **c**: Local zoomed-in view; **d**: Curvature plot

For the curve completion problem, many studies and methods have been discussed, including methods based on Euler spirals [2, 8, 10], Euler arc splines [3], and cubic B-spline curves [37], but these methods have their own disadvantages. The Euler spiral is expressed by transcendent equations, which may lead to low computational efficiency. Another shortcoming of the Euler spiral is incompatibility with NURBS. A typical curve is a polynomial curve in the Bézier form and the G1 interpolation problem can be transformed into a simple system of nonlinear equations.

An Euler arc spline [3], which consists of several arcs with the same arc length, is considered to be an extension of the Euler spiral. Although it can be represented by NURBS, the continuity at the internal connecting points is only G1 continuous, rather than continuous curvature. In contrast, the curvature of a typical curve is continuous and monotonous over the entire curve.

For the cubic B-spline curve with monotonic curvature mentioned in ref. [37], with an increase in the number of control vectors, the B-spline curve gradually converges to a straight line as the parameter trends toward one. This feature makes the bending curvature

inadequate and leads to poor control flexibility. Furthermore, the algorithm for curve completion using a monotonically cubic B-spline curve is a brutal algorithm and the cases of boundary conditions have not been fully elucidated. In the proposed method, the bending curvature can be controlled well based on the rotation angle θ or degree k, and the construction of a typical curve is simpler than the cubic B-spline curve constructed in ref. [37]. Additionally, the proposed algorithm can handle more constrained cases, which makes it more widely applicable.

Conclusions

This paper proposed a novel algorithm for G1 interpolation based on typical curves. G1 interpolation was converted into a system of nonlinear equations and a sufficient condition was provided to determine whether there is a typical curve solution with monotonic curvature in advance. The solution is obtained by means of an optimization process and numerical examples demonstrated the effectiveness and practicability of the proposed algorithm.
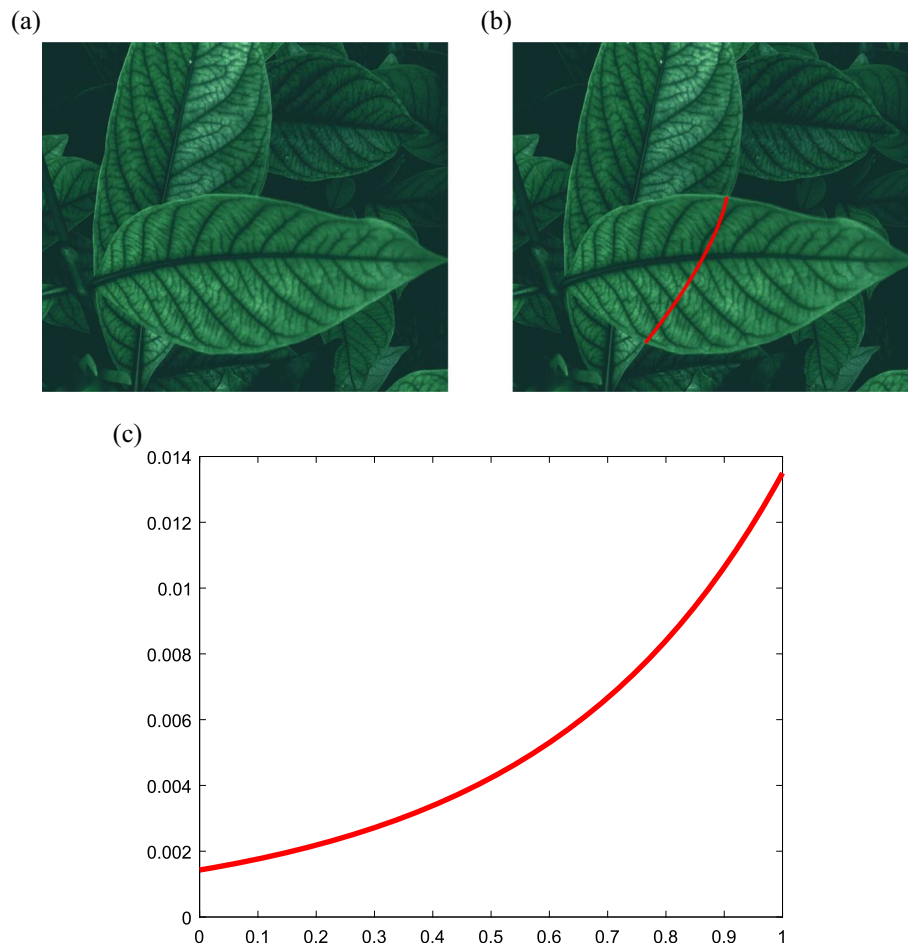
Fig. 13 Occluded leaf edge completion. **a**: Two overlapping leaves; **b**: Typical curve; **c**: Curvature plot

A single typical curve can only be 'C' shaped, but it would be useful to extend the proposed method to 'S'-shaped curves and apply it to the G2 interpolation problem. In industrial production, applications with 3D curves are valuable. Future work will involve constructing 3D curves with monotonic curvatures.

Abbreviations
CAD: Computer-aided design

Authors' contributions
AW provide the conceptualization and methodology; CH, GZ, AW, FH, ZC and SL wrote the original draft; CH and AW contributed to review and editing; all authors read and approved the final manuscript.

Availability of data and materials
Not applicable.

Declarations

Competing interests
The authors have no competing interests in the manuscript.

Author details
[1]School of Mechanical Engineering and Automation, Beihang University, Beijing 100191, China. [2]State Key Laboratory of Lunar and Planetary Sciences, Macau University of Science and Technology, Macau 999078, China. [3]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China. [4]State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China. [5]State Key Laboratory of Computer Science, Institute of Software, University of Chinese Academy of Sciences, Beijing 100049, China. [6]Faculty of Information Technology, Macau University of Science and Technology, Macau 999078, China.

References
1. Walton DJ, Meek DS (2009) G1 interpolation with a single Cornu spiral segment. J Comput Appl Math 223(1):86–96. https://doi.org/10.1016/j.cam.2007.12.022

2. Kimia BB, Frankel I, Popescu AM (2003) Euler spiral for shape completion. Int J Comput Vis 54(1–3):159–182

3. Zhou HL, Zheng JM, Yang XN (2012) Euler arc splines for curve completion. Comput Graph 36(6):642–650. https://doi.org/10.1016/j.cag.2012.04.001

4. Lin HW, Wang ZH, Feng PP, Lu XJ, Yu JH (2016) A computational model of topological and geometric recovery for visual curve completion. Comput Vis Media 2(4):329–342. https://doi.org/10.1007/s41095-016-0055-3

5. Ben-Yosef G, Ben-Shahar O (2012) A tangent bundle theory for visual curve completion. IEEE Trans Pattern Anal Mach Intell 34(7):1263–1280. https://doi.org/10.1109/TPAMI.2011.262

6. Ben-Shahar O, Ben-Yosef G (2015) Tangent bundle elastica and computer vision. IEEE Trans Pattern Anal Mach Intell 37(1):161–174. https://doi.org/10.1109/TPAMI.2014.2343214

7. Farin G, Sapidis N (1989) Curvature and the fairness of curves and surfaces. IEEE Comput Graph Appl 9(2):52–57. https://doi.org/10.1109/38.19051

8. Harary G, Tal A (2012) 3D Euler spirals for 3D curve completion. Comput Geom 45(3):115–126. https://doi.org/10.1016/j.comgeo.2011.10.001

9. Harary G, Tal A (2011) The natural 3D spiral. Comput Graph Forum 30(2): 237–246. https://doi.org/10.1111/j.1467-8659.2011.01855.x

10. Connor D, Krivodonova L (2014) Interpolation of two-dimensional curves with Euler spirals. J Comput Appl Math 261:320–332. https://doi.org/10.1016/j.cam.2013.11.009

11. Bertolazzi E, Bevilacqua P, Frego M (2020) Efficient intersection between splines of clothoids. Math Comput Simul 176:57–72. https://doi.org/10.1016/j.matcom.2019.10.001

12. Farin G (2002) Curves and surfaces for CAGD: a practical guide, 5th edn. Morgan Kaufmann, Burlington

13. Montés N, Herraez A, Armesto L, Tornero J (2008) Real-time clothoid approximation by rational bezier curves. In: Abstracts of the 2008 IEEE international conference on robotics and automation. IEEE, Pasadena, 19-23 May 2008. https://doi.org/10.1109/ROBOT.2008.4543548

14. Meek DS, Walton DJ (2004) An arc spline approximation to a clothoid. J Comput Appl Math 170(1):59–77. https://doi.org/10.1016/j.cam.2003.12.038

15. Brezak M, Petrovic I (2014) Real-time approximation of clothoids with bounded error for path planning applications. IEEE Trans Robot 30(2):507–515. https://doi.org/10.1109/TRO.2013.2283928

16. Chen Y, Cai YY, Zheng JM, Thalmann D (2017) Accurate and efficient approximation of clothoids using Bézier curves for path planning. IEEE Trans Robot 33(5):1242–1247. https://doi.org/10.1109/TRO.2017.2699670

17. Walton DJ, Meek DS (1996) A planar cubic Bézier spiral. J Comput Appl Math 72(1):85–100. https://doi.org/10.1016/0377-0427(95)00246-4

18. Walton DJ, Meek DS (1999) Planar G2 transition between two circles with a fair cubic Bézier curve. Comput Aided Des 31(14):857–866. https://doi.org/10.1016/S0010-4485(99)00073-1

19. Walton DJ, Meek DS, Ali JM (2003) Planar G2 transition curves composed of cubic Bézier spiral segments. J Comput Appl Math 157(2):453–476. https://doi.org/10.1016/S0377-0427(03)00435-7

20. Walton DJ, Meek DS (2012) A further generalisation of the planar cubic Bézier spiral. J Comput Appl Math 236(11):2869–2882. https://doi.org/10.1016/j.cam.2012.01.024

21. Walton DJ, Meek DS (2013) Curve design with more general planar Pythagorean-hodograph quintic spiral segments. Comput Aided Geom Des 30(7):707–721. https://doi.org/10.1016/j.cagd.2013.05.002

22. Mineur Y, Lichah T, Castelain JM, Giaume H (1998) A shape controled fitting method for Bézier curves. Comput Aided Geom Des 15(9):879–891. https://doi.org/10.1016/S0167-8396(98)00025-9

23. Sánchez-Reyes J (2007) Offset-rational sinusoidal spirals in Bézier form. Comput Aided Geom Des 24(3):142–150. https://doi.org/10.1016/j.cagd.2007.01.001

24. Sánchez-Reyes J (2002) P-Bézier curves, spirals, and sectrix curves. Comput Aided Geom Des 19(6):445–464. https://doi.org/10.1016/S0167-8396(02)00130-9

25. Wang AZ, He C, Zhao G, Xu HX (2019) A sufficient and necessary criterion for curvature monotone Bézier curves. J Comput Aided Des Comput Graph 31(9):1617–1621

26. Farin G (2006) Class a Bézier curves. Comput Aided Geom Des 23(7):573–581. https://doi.org/10.1016/j.cagd.2006.03.004

27. Cao J, Wang GZ (2008) A note on class a Bézier curves. Comput Aided Geom Des 25(7):523–528. https://doi.org/10.1016/j.cagd.2007.10.001

28. Wang AZ, Zhao G (2018) Counter examples of "class A Bézier curves". Comput Aided Geom Des 61:6–8. https://doi.org/10.1016/j.cagd.2018.02.001

29. Yoshida N, Hiraiwa T, Saito T (2008) Interactive control of planar class a Bézier curves using logarithmic curvature graphs. Comput Aided Des Appl 5(1–4):121–130. https://doi.org/10.3722/cadaps.2008.121-130

30. Maqsood S, Abbas M, Hu G, Ramli ALA, Miura KT (2020) A novel generalization of trigonometric Bézier curve and surface with shape parameters and its applications. Math Probl Eng 2020:4036434–4036425. https://doi.org/10.1155/2020/4036434

31. BiBi S, Abbas M, Misro MY, Hu G (2019) A novel approach of hybrid trigonometric Bézier curve to the modeling of symmetric revolutionary curves and symmetric rotation surfaces. IEEE Access 7:165779–165792. https://doi.org/10.1109/ACCESS.2019.2953496

32. BiBi S, Abbas M, Miura KT, Misro MY (2020) Geometric modeling of novel generalized hybrid trigonometric Bézier-like curve with shape parameters and its applications. Mathematics 8(6):967. https://doi.org/10.3390/math8060967

33. Maqsood S, Abbas M, Miura KT, Majeed A, Iqbal A (2020) Geometric modeling and applications of generalized blended trigonometric Bézier curves with shape parameters. Adv Differ Equ 2020(1):550. https://doi.org/10.1186/s13662-020-03001-4

34. Walton DJ, Meek DS (2008) An improved euler spiral algorithm for shape completion. In: Abstracts of the 2008 Canadian conference on computer and robot vision. IEEE, Windsor, 28-30 May 2008. https://doi.org/10.1109/CRV.2008.11

35. Henrici P (1988) Applied and computational complex analysis, volume 1: power series integration conformal mapping location of zero. Wiley, New York

36. Sharon E, Brandt A, Basri R (2002) Completion energies and scale. IEEE Trans Pattern Anal Mach Intell 22(10):1117–1131. https://doi.org/10.1109/34.879792

37. Wang AZ, He C, Hou H, Cai ZC, Zhao G (2020) Designing planar cubic B-spline curves with monotonic curvature for curve interpolation. Comput Vis Media 6(3):349–354. https://doi.org/10.1007/s41095-020-0182-8

Publisher's Note