Research Article

The Approximation of Generalized Log-Aesthetic Curves with G^2 Cubic Trigonometric Bézier Function

Diya' J. Albayari,[1,2] R. U. Gobithaasan [ID],[3,4] and Kenjiro T. Miura [ID][5]

[1]*Mathematics Department, Arab Evangelical Episcopal Church Council (Ahliyyah & Mutran), Amman 11181, Jordan*
[2]*Mathematics Department, Ibn Rushd National Academy, Amman 17126, Jordan*
[3]*Special Interest Group on Modelling & Data Analytics, Faculty of Ocean Engineering Technology and Informatics,*
 University Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia
[4]*School of Mathematics Sciences, Universiti Sains Malaysia, Gelugor 11800, Penang, Malaysia*
[5]*Graduate School of Science & Technology, Shizuoka University, 3-5-1 Johoku, Naka-Ku, Hamamatsu, Shizuoka 432-8561, Japan*

Correspondence should be addressed to R. U. Gobithaasan; gobithaasan@gmail.com

One of the requirements of curves in computer-aided design (CAD) is a curve with monotonic curvature profiles. Generalized log-aesthetic curves (GLACs) comprise a family of aesthetic curves which possesses a monotonic curvature profile. However, we cannot directly implement GLAC in CAD systems since it is in the form of a transcendental form. In this paper, we used cubic trigonometric Bézier (T-Bézier) curves with two shape parameters to approximate GLAC with G^2 continuity. The final approximation formula inherits the shape parameters of GLAC whereas T-Béziers' shape parameters are utilized to satisfy G^2 constraints. Numerical results indicate that the proposed algorithm is capable of approximating GLAC within the given tolerance in (at least) two iterations.

1. Introduction

In 2010, Gobithasaan and Miura formulated the generalized log-aesthetic curves (GLACs) [1] by extending the formulation of generalized Cornu spiral (GCS) [2], similar to log-aesthetic curves (LACs) [3]. LAC is a curve obtained by a curve synthesis process using linear representation of logarithmic curvature graph (LCG) which involves the third derivative. Due to its capabilities capturing the shape of natural spirals, LCG has been used as shape interrogation tools [4]. The main characteristic of GLAC is that it has an extra curvature shift parameter $(-\nu)$, which increases the flexibility of the curve. In general, the curvature of GLAC is always monotonic and, hence, it produces fair curves where the family is composed of clothoid, logarithmic spiral, circle involute, Neilsen's curve, and GCS.

There are two types of GLAC, the first one is called ρ-shift GLAC which can be constructed by manipulating the radius of curvature of LAC as follows [1]:

$$\kappa_{\rho-\text{GLAC}}(s) = \begin{cases} \dfrac{1}{e^{\Lambda s} + \nu}, & \alpha = 0, \\[3mm] \dfrac{1}{(\Lambda \alpha s + 1)^{1/\alpha} + \nu}, & \text{otherwise.} \end{cases} \tag{1}$$

Similarly, the second type of GLAC is called κ-shift GLAC which can be constructed by manipulating the radius of curvature of LAC as follows [1]:

$$\kappa_{\kappa-\text{GLAC}}(s) = \begin{cases} e^{-\Lambda s} + \nu, & \alpha = 0, \\[2mm] (\Lambda \alpha s + 1)^{-1/\alpha} + \nu, & \text{otherwise.} \end{cases} \tag{2}$$

Thus, the directional angle of GLAC can be represented as follows:

$$\theta_{\mathrm{GLAC}}(s) = \theta_0 + \int_0^s \kappa_{\mathrm{GLAC}}(u)\mathrm{d}u, \qquad (3)$$

and finally, the parametric form of GLAC is as follows:

$$\mathbf{C}_{\mathrm{GLAC}}(s) = \mathbf{P_0} + \begin{pmatrix} \int_0^s \cos\left(\theta_{\mathrm{GLAC}}(u)\right)\mathrm{d}u \\ \int_0^s \sin\left(\theta_{\mathrm{GLAC}}(u)\right)\mathrm{d}u \end{pmatrix}, \qquad (4)$$

where $\mathbf{P_0}$ is the start point, θ_0 is the angle between the tangent at the start point measured anticlockwise from the x-axis, and $\{\Lambda, \alpha, \nu\} \in \mathbb{R}$ are variables related to the shape of the GLAC segment [1]. Fair curves are essential in CAD/CAM environment. However, we cannot implement GLAC in the CAD system directly due to its transcendental form. Thus, one way to implement this curve for design feat is utilizing approximation to traditional curves, for example, quintic Bézier curves approximation using GCS [2] and quintic Bézier curves approximation using LAC [5].

There are numerous works on the representation of splines with trigonometric functions. In 1964, Schoenberg first introduced trigonometric B-splines as a function in space [6]. In 2003, Han introduced piecewise quadratic trigonometric polynomial curves with C^1 continuity analogous to the quadratic B-spline curves [7], and in 2004, he further proposed cubic trigonometric polynomial curves with a shape parameter [8]. A class of quartic trigonometric polynomial curves with a shape parameter was presented by Han in 2011 [9].

Similarly, there were attempts to represent traditional Bézier curves with trigonometric coordinate functions. In 1997, Walz introduced trigonometric Lagrange and Berstein polynomials to form Stancu polynomials [10]. A notable work of Han et al. is a cubic trigonometric Bézier (T-Bézier) curve with two shape parameters similar to traditional Bézier curves which were published in 2009 [9]. The advantage of T-Bézier over traditional Bézier curves is twofold: firstly, it can precisely represent circular arcs, cylinders, cones, tori, etc. Thus, CAD implementation becomes an easy task. Secondly, T-Bézier curves are located closer to the control polygon as compared to traditional Bézier curves which are of use during the designing process. The applications of trigonometric splines were the motivation for introducing various types of the trigonometric spline with features suitable for CAD/CAM applications. Readers are referred to [11–16] for further advantages of trigonometric splines in geometric modelling.

It is essential to consider the geometric constraints at the endpoints for developing splines which possess local control. For example, creating a piecewise curve which satisfies second-order geometric continuity G^2 involves matching the endpoints, its tangent vectors, and curvatures [15, 17–20]. In this paper, we approximate GLAC to cubic T-Bézier curves with G^2 conditions and derived a stable solution region so that GLAC can be approximated efficiently and directly implemented for the CAD environment.

The rest of the paper is arranged as follows. The next section introduces cubic T-Bézier curves. Section 3 elaborates the formulation to approximate GLAC with cubic T-Bézier curves. Section 4 elaborates on the curvature error measure, and it is followed by Section 5 detailing the solution region of the approximation before laying out the approximation algorithm. Finally, numerical examples are depicted before concluding remarks.

2. Preliminaries

For a given four control points P_i (i = 0, 1, 2, 3) in $\mathbb{R}^2$ or $\mathbb{R}^3$, Han et al. presented a cubic trigonometric Bézier curve with two real-valued shape parameters $\lambda, \mu \in [-2, 1]$ as follows [11]:

$$r(t) = \sum_{i=0}^{3} P_i b_i, \quad t \in [0, 1], \qquad (5)$$

where

$$b_0(t) = \left(1 - \sin\frac{\pi}{2}t\right)^2 \left(1 - \lambda \sin\frac{\pi}{2}t\right),$$

$$b_1(t) = \sin\frac{\pi}{2}t\left(1 - \sin\frac{\pi}{2}t\right)\left(2 + \lambda - \lambda \sin\frac{\pi}{2}t\right),$$

$$b_2(t) = \cos\frac{\pi}{2}t\left(1 - \cos\frac{\pi}{2}t\right)\left(2 + \mu - \mu \cos\frac{\pi}{2}t\right), \qquad (6)$$

$$b_3(t) = \left(1 - \cos\frac{\pi}{2}t\right)^2\left(1 - \mu \cos\frac{\pi}{2}t\right),$$

are the basis functions for the cubic T-Bézier curve whenever $\lambda, \mu \neq 0$. To note, when $\lambda, \mu \in [-2, 1]$, essential properties of traditional Bézier curves are preserved, e.g., non-negativity, partition of unity, monotonicity, and symmetry as proven in [11].

The derivatives of $r(t)$ at the endpoints are given as follows:

$$r(0) = P_0,$$

$$r'(0) = \frac{\pi}{2}(\lambda + 2)(P_1 - P_0),$$

$$r''(0) = \frac{\pi^2}{2}((2\lambda + 1)P_0 - 2(\lambda + 1)P_1 + P_2),$$

$$r(1) = P_3, \qquad (7)$$

$$r'(1) = \frac{\pi}{2}(\mu + 2)(P_3 - P_2),$$

$$r''(1) = \frac{\pi^2}{2}(P_1 - 2(\mu + 1)P_2 + (2\mu + 1)P_3).$$

3. The Approximation Method

Let the GLAC be in a normalized form defined over $s \in [0, S]$. We can express a GLAC segment in normalized form by translating it to the origin, then rotate it such that its

initial tangent point resides along the positive x-axis ($\theta_0 = 0$) and finally scale it to make the arc length $S = 1$. The rest of this paper employs κ-shift GLAC for cubic T-Bézier approximation. To note the derivation of ρ-shift approximation [21] is straightforward and similar to the steps shown as follows.

Let $\kappa_\kappa(s)$ be the curvature function of κ-shift GLAC as stated in equation (2), (x, y) be the endpoint where $x = x(1)$ and $y = y(1)$, and θ be the winding angle at the endpoint such that $\theta = \int_0^1 \kappa(s)ds$. Then, the derivatives of GLAC at the endpoints are given as follows:

$$
\begin{aligned}
C_{GLAC}(0) &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} C_{GLAC}(1), \\[6pt]
C'_{GLAC}(0) &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} C'_{GLAC}(1) = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}, \\[6pt]
C''_{GLAC}(0) &= \begin{pmatrix} 0 \\ (1+\nu) \end{pmatrix} C''_{GLAC}(1) = (\Lambda\alpha + 1)^{-1/\alpha} + \nu) \begin{pmatrix} -\sin\theta \\ \cos\theta \end{pmatrix}, \\[6pt]
C''_{GLAC}(0) &= \begin{pmatrix} 0 \\ (1+\nu) \end{pmatrix} C''_{GLAC}(1) = (\Lambda\alpha + 1)^{-1/\alpha} + \nu) \begin{pmatrix} -\sin\theta \\ \cos\theta \end{pmatrix},
\end{aligned}
\tag{8}
$$

where $'$ is the differentiation with respect to the parameter s.

To guarantee the second-order geometric continuity (G^2) between $r(t)$ and $C_{GLAC}(u)$ at a, it is required to establish $\widetilde{C_{GLAC}}(h(u))$ parameterization equivalent to $C_{GLAC}(u)$ satisfying the following equation [2]:

$$
\frac{d^i}{dt^i} r(a) = \widetilde{C_{GLAC}} \frac{d^i}{du^i}(a), \quad i = 0, 1, 2.
\tag{9}
$$

The accompanying sets of equations are given as follows [2]:

$$
\begin{aligned}
r(a) &= C_{GLAC}(a), \\
r'(a) &= \tau_1 C'_{GLAC}(a), \\
r''(a) &= \tau_2 C'_{GLAC}(a) + \tau_1^2 C''_{GLAC}(a),
\end{aligned}
\tag{10}
$$

where τ_1 and τ_2 are real-valued shape factors. The second-order parametric continuity, C^2, is obtained by letting $\tau_1 = 1$ and $\tau_2 = 0$.

Consider the ith derivative of $C_{GLAC}(s)$ w.r.t. s evaluated at $s = a$ denoted by D_a^j, then to approximate GLAC segment using G^2 cubic T-Bézier curve, we should satisfy the set of equations in equation (9) at the start and endpoints with the shape parameters ξ_1 and ξ_2 used at the start point and the shape parameters ω_1 and ω_2 used at the endpoint.

At the start point, we have the following set of equations:

$$
P_0 = D_0^0,
$$

$$
\frac{\pi}{2}(\lambda + 2)(P_1 - P_0) = \xi_1 D_0^1,
\tag{11}
$$

$$
\frac{\pi^2}{2}((2\lambda + 1)P_0 - 2(\lambda + 1)P_1 + P_2) = \xi_2 D_0^1 + \xi_1^2 D_0^2.
$$

After algebraic simplifications, we get the control points at the start point as follows:

$$
P_0 = D_0^0,
\tag{12}
$$

$$
P_1 = \frac{(2+\lambda)\pi D_0^0 + 2\xi_1 D_0^1}{(2+\lambda)\pi},
\tag{13}
$$

$$
P_2 = \frac{(2+\lambda)\pi^2 D_0^0 + 4(1+\lambda)\xi_1 \pi D_0^1 + 2(2+\lambda)\xi_2 D_0^1 + 2(2+\lambda)\xi_1^2 D_0^2}{(2+\lambda)\pi^2}.
\tag{14}
$$

Similarly, at the endpoint, we have the following set of equations:

$$P_3 = D_1^0,$$ and

$$\frac{\pi}{2}(\mu + 2)(P_3 - P_2) = \omega_1 D_1^1,$$ (15)

$$\frac{\pi^2}{2}(P_1 - 2(\mu + 1)P_2 + (2\mu + 1)P_3) = \omega_2 D_1^1 + \omega_1^2 D_1^2,$$

$$P_1 = \frac{(2 + \mu)\pi^2 D_1^0 - 4(1 + \mu)\omega_1 \pi D_1^1 + 2(2 + \mu)\omega_2 D_1^1 + 2(2 + \mu)\omega_1^2 D_1^2}{(2 + \lambda)\pi^2},$$ (16)

$$P_2 = \frac{(2 + \mu)\pi D_1^0 - 2\omega_1 D_1^1}{(2 + \mu)\pi},$$ (17)

$$P_3 = D_1^0.$$ (18)

For the control points P_1 and P_2, each has two expressions at the start and endpoints. Thus, we have four equations in x and y coordinate functions to be solved for λ and μ, leading to the following equations:

$$\lambda(\xi_1, \omega_1) = \frac{2\pi\xi_1 \sin\theta}{\pi^2(x\sin\theta - y\cos\theta) - 2\left((\Lambda\alpha + 1)^{-1/\alpha} + \nu\right)\omega_1^2} - 2,$$ (19)

$$\mu(\xi_1, \omega_1) = \frac{2\pi\omega_1 \sin\theta}{\pi^2 y - 2(1 + \nu)\xi_1^2} - 2.$$ (20)

Definition 1. G^2 cubic T-Bézier approximation

A G^2 cubic trigonometric approximation curve of GLAC is defined by the following equation:

$$\tilde{r}(t) = \sum_{i=0}^{3} P_i b_i, \quad t \in [0, 1], \xi_1, \omega_1 \in \delta,$$ (21)

where δ is the solution region shown in Section 6, with the definitions of basis functions, b_i, as stated in equation (5), λ and μ as stated in equations (19) and (20). By employing equations (12), (13), (17), and (18), the control points, P_i, are defined as follows:

$$P_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

$$P_1 = \begin{pmatrix} x - y\cot(\theta) - \dfrac{2\omega_1^2 \csc(\theta)\left((\alpha\Lambda + 1)^{-1/\alpha} + \nu\right)}{\pi^2} \\ 0 \end{pmatrix},$$

$$P_2 = \begin{pmatrix} x - y\cot(\theta) + \dfrac{2(\nu + 1)\xi_1^2 \cot(\theta)}{\pi^2} \\ \dfrac{2(\nu + 1)\xi_1^2}{\pi^2} \end{pmatrix},$$

$$P_3 = \begin{pmatrix} x \\ y \end{pmatrix}.$$

(22)

The free shape parameters for the approximated curve are α, Λ, and ν which are inherited from GLAC. λ and μ are calculated by searching for suitable values ξ_1 and ω_1 satisfying G^2 data. Once these shape parameters are given values by the designer, then we would be able to identify their corresponding control points as stated in Definition 1.

Definition 2. An alternative to G^2 cubic T-Bézier approximation

One can use the equations (14) and (16) instead of equations (13) and (17) to define the control points P_1 and P_2

while employing the basis functions, b_i, as stated in equation (5), λ and μ as stated in equations (19) and (20). Then, the G^2 cubic trigonometric approximation of GLAC is defined as follows:

$$\bar{r}(t) = \sum_{i=0}^{3} P_i b_i, \quad t \in [0,1], \xi_1, \omega_1 \in \delta, \tag{23}$$

where

$$\xi_2(\xi_1, \omega_1) = (\nu + 1)\xi_1^2 \cot(\theta) - 2\left(\omega_1^2 \csc(\theta)\left((\alpha\Lambda + 1)^{-1/\alpha} + \nu\right) + \pi\left(3\pi x - 4\xi_1 - 3\pi y \cot(\theta)\right)\right),$$
$$\omega_2(\xi_1, \omega_1) = -\frac{1}{2}\csc(\theta)\left(2\omega_1^2 \cos(\theta)\left((\alpha\Lambda + 1)^{-1/\alpha} + \nu\right) - 4\pi\omega_1 \sin(\theta) - 4(\nu + 1)\xi_1^2 + 3\pi^2 y\right).$$
$$\tag{24}$$

Since both definitions provide the same results, it is recommended to use $\tilde{r}(t)$ in Definition 1 instead of $\bar{r}(t)$ to avoid extra computations in calculating the values of ξ_2 and ω_2.

4. Curvature Error Measure

It is known that the curvature of GLAC is a smooth monotonic function which guarantees fairness. Thus, we used the curvature error measure as proposed by Cross and Cripps which was used to approximate GCS [2] as follows:

$$\epsilon_{\max} = \max_{s\in[0,1]} \frac{|\kappa_r(s) - \kappa_C(s)|}{\max\{\kappa_C(s), 1\}}, \tag{25}$$

where $\kappa_r(s)$ is the curvature function of the Bézier curve defined in the domain $[0,1]$ and $\kappa_C(s)$ is the curvature function of the GCS. The advantage of this measure is that it can accurately reflect the curvature profile and produce smooth approximation results. However, the disadvantage is that it is computationally costly due to the arc length reparameterization of the Bézier curve. Thus, to reduce the high computational cost, Albayari [22] proposed another curvature error measure as follows:

$$\bar{\epsilon}_{\max} = \max_{t\in[0,1]} \frac{|\kappa_r(t) - \kappa_C(t)|}{\max\{\kappa_C(t), 1\}}, \tag{26}$$

where $\kappa_r(t)$ is the curvature function of the Bézier curve and $\kappa_C(t)$ is the curvature function of the GCS, and to calculate the error, one has to sample a sequence of equally spaced parameter values $\{t_i\}_{i=0}^{N}$. Even though Lu's curvature error measure takes less computation than Cross and Cripps's curvature measure, we still must calculate Cross and Cripps's error measure to exactly identify the quality of an approximation. Hence, one may end up calculating both $\epsilon_{\max}$ and $\bar{\epsilon}_{\max}$ for an efficient approximation [23].

In 2016, Albayari et al. proposed a new error measure that can reduce the computations and determine the acceptability of the approximations accurately as follows [21]:

$$\tilde{\epsilon}_{\max} = \max_{i=0,\cdots,n} \frac{|\kappa_r(t_i) - \kappa_C(s_i)|}{\max\{\text{Abs}[\kappa_C(s_i)], 1\}}, \tag{27}$$

where $\kappa_r(t_i)$ and $\kappa_C(s_i)$, $i = 0, \ldots, n$, are the correspondence curvatures of the curve and the approximation curve at the same arc length, respectively. Although their measure reduces the computation, and it is very accurate to the exact error but the accuracy of the results depends on the numbers of the selected nodes. Since there is no rule to identify an ideal number of nodes which satisfy the exact error for all cases, and the maximum error may occur in between some of the two nodes. Thus, Lu introduced a new error measure as follows [21]:

$$\dot{\epsilon}_{\max} = \max_{t\in[0,1]} \frac{|\kappa_r(s) - \kappa_C(s)|}{\max\{\kappa_C(s), 1\}}, \tag{28}$$

where $\kappa_r(s)$ is the curvature function of $r(s(t))$ normalized to the domain $[0,1]$ and $\kappa_C(s)$ is the curvature function of the GLAC. By using this error measure, one may obtain a parameterized arc length curvature error measure by interpolating the arc length function of GLAC to avoid unnecessary computation. The default tolerance is set to be 0.05 which guarantees a high-quality approximation as proposed by Cross and Cripps [2].

Similarly, we used the error measure proposed by Lu [21] which is tailored to approximate GLAC as follows:

$$\dot{\epsilon}_{\max} = \max_{t\in[0,1]} \frac{|\kappa_{\tilde{r}}(s) - \kappa_{C-\text{GLAC}}(s)|}{\max\{\kappa_{C-\text{GLAC}}(s), 1\}}, \tag{29}$$

where $\kappa_{\tilde{r}}(s)$ is the curvature function of $\tilde{r}(s(t))$ normalized to the domain $[0,1]$ and $\kappa_{C-\text{GLAC}}(s)$ is the curvature function of the GLAC which can be either κ-shift or ρ-shift GLAC.

5. Solution Region

The first step in the approximation is to set an initial value to the shape parameters ξ_1 and ω_1. A plausible set of initial values for shape parameters are $\xi_1 = \omega_1 = 1$ which satisfies

C^2 continuity [2]. However, in some cases, these initial values provide unacceptable errors. Thus, it is essential to find a suitable solution region for the shape parameters ξ_1 and ω_1 to meet the tolerance in a shorter period.

Indeed, there is a geometric effect of the two shape parameters ξ_1 and ω_1 on the initial and end tangent vectors. From equations (12), (13), (17), and (18), it is evident that ξ_1 relates to the magnitude of $\|P_1 - P_0\|$ and ω_1 relates to the

magnitude of $\|P_3 - P_2\|$. Since $\lambda, \mu > -2$, the shape parameters ξ_1 and ω_1 should be positive; otherwise, the tangents will point in the opposite direction. Hence, we set the lower bound of ξ_1 and ω_1 as 0. Additionally, the values of the shape parameters should not be large otherwise the control points will be far away from each other. Since the total arc length of the curve $\int_0^1 \|V'(s)\| ds$ is approximately 1, so using the trapezoid rule, we get the following equation [2]:

$$1 \approx \int_0^1 \|r'(s)\| ds \approx \frac{1}{4}\|r'(0)\| + \frac{1}{2}\left\|r'\left(\frac{1}{2}\right)\right\| + \frac{1}{4}\|r'(1)\| = \frac{1}{4}\xi_1 + \frac{1}{2}\left\|r'\left(\frac{1}{2}\right)\right\| + \frac{1}{4}\omega_1 > \frac{1}{4}(\xi_1 + \omega_1). \tag{30}$$

Hence, an appropriate upper bound for the shape parameters ξ_1 and ω_1 is 4. It can be concluded that the entire solution region of the shape parameters ξ_1 and ω_1 is $\Delta = \{(\xi_1, \omega_1) \in [0,4] \times [0,4]\}$.

Referring to equations (13) and (17), the value of λ becomes -2 when $\xi_1 = 0$. Similarly, the value of μ becomes -2 at $\omega_1 = 0$. These values may lead to undefined control points P_1 and P_2; hence, we should drop 0 from the solution region. Since the shape parameters ξ_1 and ω_1 affect the values of λ and μ so we should consider the values in Δ for which λ and μ are still in $[-2, 1]$. Hence, the restricted solution region becomes as follows:

$$\delta = \{(\xi_1, \omega_1) \in (0,4] \times (0,4]: -2 < \lambda(\xi_1, \omega_1), \mu(\xi_1, \omega_1) \le 1\}. \tag{31}$$

6. The Basis Functions' Properties of G^2 GLAC T-Bézier Approximated Curves

The proposed G^2 basis functions for all GLAC approximated curves possess various properties. The following theorems elucidate these properties. Without loss of generality, the case when $\alpha = 0$ for κ-shift GLAC is discussed in Theorems 3 and 4. However, all other 3 different cases are discussed in detail in [21].

Theorem 3. *The proposed G^2 basis functions of the G^2 GLAC cubic trigonometric approximated curve satisfy the partition of unity property.*

Proof. After algebraic simplification of the basis functions, the approximated curve basis functions stated in equation (5) can be rewritten as follows:

$$b_{\kappa,0}(t, \lambda(\xi_1, \omega_1), \mu(\xi_1, \omega_1)) = \frac{2\pi[ee]^\Lambda \xi_1 \sin(\theta)\sin^3(\pi t/2)}{2\omega_1^2([ee]^\Lambda \nu + 1) + \pi^2[ee]^\Lambda(y\cos(\theta) - x\sin(\theta))}$$

$$- \frac{4\pi[ee]^\Lambda \xi_1 \sin(\theta)\sin^2(\pi t/2)}{2\omega_1^2([ee]^\Lambda \nu + 1) + \pi^2[ee]^\Lambda(y\cos(\theta) - x\sin(\theta))} \tag{32}$$

$$+ \frac{2\pi[ee]^\Lambda \xi_1 \sin(\theta)\sin(\pi t/2)}{2\omega_1^2([ee]^\Lambda \nu + 1) + \pi^2[ee]^\Lambda(y\cos(\theta) - x\sin(\theta))} + 2\sin^3\left(\frac{\pi t}{2}\right) - 3\sin^2\left(\frac{\pi t}{2}\right) + 1,$$

$$b_{\kappa,1}(t, \lambda(\xi_1, \omega_1), \mu(\xi_1, \omega_1)) = -\frac{2\pi[ee]^\Lambda \xi_1 \sin(\theta)\sin^3(\pi t/2)}{2\omega_1^2([ee]^\Lambda \nu + 1) + \pi^2[ee]^\Lambda(y\cos(\theta) - x\sin(\theta))}$$

$$+ \frac{4\pi[ee]^\Lambda \xi_1 \sin(\theta)\sin^2(\pi t/2)}{2\omega_1^2([ee]^\Lambda \nu + 1) + \pi^2[ee]^\Lambda(y\cos(\theta) - x\sin(\theta))} \tag{33}$$

$$- \frac{2\pi[ee]^\Lambda \xi_1 \sin(\theta)\sin(\pi t/2)}{2\omega_1^2([ee]^\Lambda \nu + 1) + \pi^2[ee]^\Lambda(y\cos(\theta) - x\sin(\theta))} - 2\sin^3\left(\frac{\pi t}{2}\right) + 2\sin^2\left(\frac{\pi t}{2}\right),$$

$$b_{\kappa,2}(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1)) = \frac{2\pi\omega_1\sin(\theta)\cos^3(\pi t/2)}{\pi^2 y - 2(\nu+1)\xi_1^2} - \frac{4\pi\omega_1\sin(\theta)\cos^2(\pi t/2)}{\pi^2 y - 2(\nu+1)\xi_1^2}$$

$$+ \frac{2\pi\omega_1\sin(\theta)\cos(\pi t/2)}{\pi^2 y - 2(\nu+1)\xi_1^2} - 2\cos^3\left(\frac{\pi t}{2}\right) + 2\cos^2\left(\frac{\pi t}{2}\right), \tag{34}$$

$$b_{\kappa,3}(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1)) = -\frac{2\pi\omega_1\sin(\theta)\cos^3(\pi t/2)}{\pi^2 y - 2(\nu+1)\xi_1^2} + \frac{4\pi\omega_1\sin(\theta)\cos^2(\pi t/2)}{\pi^2 y - 2(\nu+1)\xi_1^2}$$

$$- \frac{2\pi\omega_1\sin(\theta)\cos(\pi t/2)}{\pi^2 y - 2(\nu+1)\xi_1^2} + 2\cos^3\left(\frac{\pi t}{2}\right) - 3\cos^2\left(\frac{\pi t}{2}\right) + 1. \tag{35}$$

The first four terms of $b_{\kappa,0}(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1))$ stated in equation (32) are the negation terms of the first four terms of $b_{\kappa,1}(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1))$ stated in equation (33). Similarly, for the first four terms of $b_{\kappa,2}(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1))$ stated in equation (34) with the first four terms of $b_{\kappa,3}(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1))$ stated in equation (35), the summation of the basis functions is as follows:

$$\sum_{i=0}^{3} b_{\kappa,i}(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1)) = -\sin^2\left(\frac{\pi t}{2}\right) + 1 - \cos^2\left(\frac{\pi t}{2}\right) + 1 = 2 - \left(\sin^2\left(\frac{\pi t}{2}\right) + \cos^2\left(\frac{\pi t}{2}\right)\right) = 1. \tag{36}$$

□

Theorem 4. *The proposed G^2 basis functions of the G^2 GLAC cubic trigonometric approximated curves satisfy the symmetric property.*

Proof. The symmetric property for the G^2 basis functions holds if the following equations hold for $i = 0, 1, 2, 3$.

$$b_i(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1)) = b_{3-i}(1-t,\mu(\xi_1,\omega_1),\lambda(\xi_1,\omega_1)). \tag{37}$$

From equations (5), (19), and (20), $b_{\kappa,0}(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1))$ for the κ-shift GLAC approximated curve can be written as follows:

$$b_{\kappa,0}(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1)) = \left(\sin\left(\frac{\pi t}{2}\right) - 1\right)^2 * \left(2\sin\left(\frac{\pi t}{2}\right)\left(\frac{\pi[ee]^\Lambda \xi_1 \sin(\theta)}{2\omega_1^2([ee]^\Lambda \nu + 1) + \pi^2[ee]^\Lambda(y\cos(\theta) - x\sin(\theta))} + 1\right) + 1\right). \tag{38}$$

Similarly, the value of $b_3(1-t,\mu(\xi_1,\omega_1),\lambda(\xi_1,\omega_1))$ can be calculated as follows:

$$b_{\kappa,3}(1-t,\mu(\xi_1,\omega_1),\lambda(\xi_1,\omega_1)) = 4\cos^4\left(\frac{1}{4}\pi(t+1)\right) * \left(2\sin\left(\frac{\pi t}{2}\right)\left(\frac{\pi[ee]^\Lambda \xi_1 \sin(\theta)}{2\omega_1^2([ee]^\Lambda \nu + 1) + \pi^2[ee]^\Lambda(y\cos(\theta) - x\sin(\theta))} + 1\right) + 1\right). \tag{39}$$

It is evident from equations (38) and (39) that it is sufficient to show $4\cos^4((1/4)\pi(t+1)) = (\sin((1/2)\pi t) - 1)^2$ to prove that $b_0(t,\lambda(\xi_1,\omega_1),\mu(\xi_1,\omega_1)) = b_3(1-t,\mu(\xi_1,\omega_1),\lambda(\xi_1,\omega_1))$.

Based on the trigonometric identities in equations (40) and (41), we can complete the proof by equation (42).

$$\cos^2\left(\frac{\theta}{2}\right) = \frac{1}{2}\left(\cos(\theta) + 1\right), \tag{40}$$

$$\cos(x + y) = \cos(x)\cos(y) - \sin(x)\sin(y), \tag{41}$$

$$4\cos^4\left(\frac{1}{4}\pi(t+1)\right) = 4\left(\cos^2\left(\frac{1/2\pi\,(t+1)}{2}\right)\right)^2 = 4\left(\frac{1}{2}\left(\cos\left(\frac{1}{2}\pi(t+1)\right)+1\right)\right)^2 = \left(\cos\left(\frac{1}{2}\pi(t+1)\right)+1\right)^2$$
$$= \left(\left(\cos\left(\frac{1}{2}\pi t\right)\cos\left(\frac{\pi}{2}\right) - \sin\left(\frac{1}{2}\pi t\right)\sin\left(\frac{\pi}{2}\right)\right)+1\right)^2 = \left(1 - \sin\left(\frac{1}{2}\pi t\right)\right)^2 = \left(\sin\left(\frac{1}{2}\pi t\right)-1\right)^2. \tag{42}$$

To prove the symmetric property for the remaining basis functions, it is required to show that $b_1(t, \lambda(\xi_1, \omega_1), \mu(\xi_1, \omega_1)) = b_2(1 - t, \mu(\xi_1, \omega_1), \lambda(\xi_1, \omega_1))$, which recall from equations (5), (19), and (20) that the value of $b_1(t, \lambda(\xi_1, \omega_1), \mu(\xi_1, \omega_1))$ is as follows:

$$b_{\kappa,1}\left(t, \lambda(\xi_1, \omega_1), \mu(\xi_1, \omega_1)\right) = -\left(2\left(\sin\left(\frac{\pi t}{2}\right)-1\right)\sin\left(\frac{\pi t}{2}\right)\right)\left(\frac{\pi[ee]^{\Lambda}\xi_1\sin(\theta)(\sin(\pi t/2)-1)}{2\omega_1^2\left([ee]^{\Lambda}\nu+1\right)+\pi^2[ee]^{\Lambda}(y\cos(\theta)-x\sin(\theta))}\right.$$
$$\left. + \frac{\sin(\pi t/2)\left(2\omega_1^2\left([ee]^{\Lambda}\nu+1\right)+\pi^2[ee]^{\Lambda}(y\cos(\theta)-x\sin(\theta))\right)}{2\omega_1^2\left([ee]^{\Lambda}\nu+1\right)+\pi^2[ee]^{\Lambda}(y\cos(\theta)-x\sin(\theta))}\right), \tag{43}$$

where the value of $b_{\kappa,2}(1 - t, \mu(\xi_1, \omega_1), \lambda(\xi_1, \omega_1))$ can be calculated as follows:

$$b_{\kappa,2}\left(1 - t, \mu(\xi_1, \omega_1), \lambda(\xi_1, \omega_1)\right) = \left(2\left(1-\sin\left(\frac{\pi t}{2}\right)\right)\times\sin\left(\frac{\pi t}{2}\right)\right)\left(\frac{\pi[ee]^{\Lambda}\xi_1\sin(\theta)(\sin(\pi t/2)-1)}{2\omega_1^2\left([ee]^{\Lambda}\nu+1\right)+\pi^2[ee]^{\Lambda}(y\cos(\theta)-x\sin(\theta))}\right.$$
$$\left. + \frac{\sin(\pi t/2)\left(2\omega_1^2\left([ee]^{\Lambda}\nu+1\right)+\pi^2[ee]^{\Lambda}(y\cos(\theta)-x\sin(\theta))\right)}{2\omega_1^2\left([ee]^{\Lambda}\nu+1\right)+\pi^2[ee]^{\Lambda}(y\cos(\theta)-x\sin(\theta))}\right). \tag{44}$$

Multiplying the constant -1 by the first parenthesis of equation (43) makes it identically equal to equation (44). Therefore, the symmetric property holds. $\square$

Theorem 5. *The proposed G^2 basis functions are non-negative functions.*

Proof. The proposed G^2 basis functions are constructed by substituting the proposed G^2 formulas of the shape parameters λ and μ in Han's basis functions [7] stated in equation (5) as follows:

$$
\begin{aligned}
b_0(t, \lambda(\xi_1, \omega_1)) &= \left(1 - \sin\frac{\pi}{2}t\right)^2 \left(1 - \lambda(\xi_1, \omega_1)\sin\frac{\pi}{2}t\right), \\
b_1(t, \lambda(\xi_1, \omega_1)) &= \sin\frac{\pi}{2}t\left(1 - \sin\frac{\pi}{2}t\right)\left(2 + \lambda(\xi_1, \omega_1) - \lambda(\xi_1, \omega_1)\sin\frac{\pi}{2}t\right), \\
b_2(t, \mu(\xi_1, \omega_1)) &= \cos\frac{\pi}{2}t\left(1 - \cos\frac{\pi}{2}t\right)\left(2 + \mu(\xi_1, \omega_1) - \mu(\xi_1, \omega_1)\cos\frac{\pi}{2}t\right), \\
b_3(t, \mu(\xi_1, \omega_1)) &= \left(1 - \cos\frac{\pi}{2}t\right)^2 \left(1 - \mu(\xi_1, \omega_1)\cos\frac{\pi}{2}t\right).
\end{aligned}
\tag{45}
$$

The search region, δ, of the shape parameters ξ_1 and ω_1 for all GLAC approximated curves restricts ξ_1 and ω_1 over a domain which encloses the corresponding shape parameters $\lambda(\xi_1, \omega_1)$ and $\mu(\xi_1, \omega_1)$ in the interval $[-2, 1]$. Therefore, taking into consideration that the general parameter, t, varies over the interval $[0, 1]$, it becomes clear that $(1 - \lambda(\xi_1, \omega_1)\sin \pi/2t) \geq 0$, and $(1 - \sin \pi/2t) \geq 0$, which follows that $b_0(t, \lambda(\xi_1, \omega_1)) \geq 0$. Similarly, $(2 + \lambda(\xi_1, \omega_1) - \lambda(\xi_1, \omega_1)\sin \pi/2t) \geq 0$, $\sin \pi/2t > 0$, and $(1 - \sin \pi/2t) > 0$, which follows that $b_1(t, \lambda(\xi_1, \omega_1)) > 0$, $b_2(t, \mu(\xi_1, \omega_1))$, and $b_3(t, \mu(\xi_1, \omega_1))$. $\square$

Theorem 6. *For a given parameter t, the proposed G^2 basis functions preserve monotonicity property for the corresponding shape parameters $\lambda(\xi_1, \omega_1)$ and $\mu(\xi_1, \omega_1)$ as follows.*

(1) $b_0(t, \lambda(\xi_1, \omega_1))$ monotonically decreasing for the corresponding shape parameter $\lambda(\xi_1, \omega_1)$.

(2) $b_1(t, \lambda(\xi_1, \omega_1))$ monotonically increasing for the corresponding shape parameter $\lambda(\xi_1, \omega_1)$.

(3) $b_2(t, \mu(\xi_1, \omega_1))$ monotonically increasing for the corresponding shape parameter $\mu(\xi_1, \omega_1)$.

(4) $b_3(t, \mu(\xi_1, \omega_1))$ monotonically decreasing for the corresponding shape parameter $\mu(\xi_1, \omega_1)$.

Proof. This can be proved by considering the sign of the first derivatives for the basis functions over the entire domain $t \in [0, 1]$.

From equations in equation (45), the first derivatives of the basis functions with respect to the corresponding shape parameters can be calculated as follows:

$$
\begin{aligned}
b_0'(\lambda(\xi_1, \omega_1)) &= -\left(\sin\left(\frac{\pi t}{2}\right) - 1\right)^2 \sin\left(\frac{\pi t}{2}\right), \\
b_1'(\lambda(\xi_1, \omega_1)) &= \left(\sin\left(\frac{\pi t}{2}\right) - 1\right)^2 \sin\left(\frac{\pi t}{2}\right), \\
b_2'(\mu(\xi_1, \omega_1)) &= 4\sin^4\left(\frac{\pi t}{4}\right)\cos\left(\frac{\pi t}{2}\right), \\
b_3'(\mu(\xi_1, \omega_1)) &= -4\sin^4\left(\frac{\pi t}{4}\right)\cos\left(\frac{\pi t}{2}\right).
\end{aligned}
\tag{46}
$$

The derivatives' functions $b_0'(\lambda(\xi_1, \omega_1))$ and $b_3'(\mu(\xi_1, \omega_1))$ are negative over the entire domain $t \in [0, 1]$, where the derivatives' function $b_1'(\lambda(\xi_1, \omega_1))$ and $b_2'(\mu(\xi_1, \omega_1))$ are positive over the entire domain $t \in [0, 1]$ which proves the monotonicity of the proposed basis functions, as mentioned in Theorem 6.

To proof that the approximation curve $\mathbf{r}(t)$ is invariant under translation, consider the translation $\mathbf{m} = (x^*, y^*)$, then

$$\mathbf{r}^*(t) = \begin{bmatrix} b_0(t) & b_1(t) & b_2(t) & b_2(t) \end{bmatrix} \begin{bmatrix} P_0^* \\ P_1^* \\ P_2^* \\ P_3^* \end{bmatrix} = \begin{bmatrix} b_0(t) & b_1(t) & b_2(t) & b_2(t) \end{bmatrix} \begin{bmatrix} P_0 + (x^*, y^*) \\ P_1 + (x^*, y^*) \\ P_2 + (x^*, y^*) \\ P_3 + (x^*, y^*) \end{bmatrix}$$

$$= \left\{ \left(\sin\frac{\pi}{2}t \left(1 - \sin\frac{\pi}{2}t \right) \left(2 + \left(\frac{2\pi\xi_1 \sin\theta}{\pi^2(x \quad \sin\theta - y \quad \cos\theta) - 2\left((\Lambda\alpha + 1)^{-1/\alpha} + \nu \right)\omega_1^2} - 2 \right) \right. \right.$$

$$- \left(\frac{2\pi\xi_1 \sin\theta}{\pi^2(x \quad \sin\theta - y \quad \cos\theta) - 2\left((\Lambda\alpha + 1)^{-1/\alpha} + \nu \right)\omega_1^2} - 2 \right) \quad \sin\frac{\pi}{2}t \right) \bigg)$$

$$\cdot \left(x - y\cot(\theta) - \frac{2\omega_1^2 \csc(\theta)\left((\alpha\Lambda + 1)^{-1/\alpha} + \nu \right)}{\pi^2} + x^* \right) \left(x - y\cot(\theta) - \frac{2\omega_1^2 \csc(\theta)\left((\alpha\Lambda + 1)^{-1/\alpha} + \nu \right)}{\pi^2} + x^* \right)$$

$$+ \left(\cos\frac{\pi}{2}t \left(1 - \cos\frac{\pi}{2}t \right) \left(2 + \left(\frac{2\pi\omega_1 \sin\theta}{\pi^2 y - 2(1 + \nu)\xi_1^2} - 2 \right) - \left(\frac{2\pi\omega_1 \sin\theta}{\pi^2 y - 2(1 + \nu)\xi_1^2} - 2 \right) \quad \cos\frac{\pi}{2}t \right) \right) (x - y\cot(\theta)$$

$$+ \frac{2(\nu + 1)\xi_1^2 \cot(\theta)}{\pi^2} + x^* \right) + \left(\left(1 - \cos\frac{\pi}{2}t \right)^2 \left(1 - \left(\frac{2\pi\omega_1 \sin\theta}{\pi^2 y - 2(1 + \nu)\xi_1^2} - 2 \right) \quad \cos\frac{\pi}{2}t \right) \right) (x + x^*),$$

$$\left(\cos\frac{\pi}{2}t \left(1 - \cos\frac{\pi}{2}t \right) \left(2 + \left(\frac{2\pi\omega_1 \sin\theta}{\pi^2 y - 2(1 + \nu)\xi_1^2} - 2 \right) - \left(\frac{2\pi\omega_1 \sin\theta}{\pi^2 y - 2(1 + \nu)\xi_1^2} - 2 \right) \quad \cos\frac{\pi}{2}t \right) \right) \left(\frac{2(\nu + 1)\xi_1^2}{\pi^2} + y^* \right)$$

$$+ \left(\left(1 - \cos\frac{\pi}{2}t \right)^2 \left(1 - \left(\frac{2\pi\omega_1 \sin\theta}{\pi^2 y - 2(1 + \nu)\xi_1^2} - 2 \right) \quad \cos\frac{\pi}{2}t \right) \right) (y + y^*) \right\}$$

$$= \mathbf{r}(t) + \left(b_0(t) + b_1(t) + b_2(t) + b_3(t) \right) (x^*, y^*) = \mathbf{r}(t) + \left(\sum_{i=0}^{3} b_i(t) \right) \mathbf{m}.$$

(47)

For the approximation curve to be invariant, the Cauchy's relation $\sum_{i=0}^{3} b_i(t) = 1$, must be satisfied [24], which is proved in Theorem 3.   □

7. Approximation Algorithm

This section illustrates the overall algorithm to approximate GLAC with G^2 cubic T-Bézier as shown in Algorithm 1. The numerical method used to search for suitable ξ_1 and ω_1 in the algorithm is the interior point method utilizing the FindMinimum [] command found in Mathematica. Based on numerical results discussed in the next section, the values of ξ_1 and ω_1 is always around (1, 1). This is because the length of normalized GLAC where the parameter s is equal to 1.

8. Numerical Examples

As stated in the previous section, we may generate LAC with the formulation of GLAC by letting $\nu = 0$ [1]. Table 1 depicts various types of GLAC segments (including LAC) with different values of shape parameters to show the efficiency of the proposed method in approximating the GLAC family with cubic T-Béziers.

Numerous approximation examples of LACs ($\nu = 0$) and GLACs are depicted in Table 1. The First 1–5 examples show the results of approximating various GLACs. The next 6–13 examples depict special cases of GLAC and LAC ($\nu = 0$). Most cases terminate after the first run which indicates the efficiency of the proposed algorithm. However, two cases (examples 14 and 15) have error of the approximation which

---

INPUT: GLAC Shape parameters Λ, α, ν; by default *tol* is set as: tol = 0.05; the maximum number of iterations n.
OUTPUT: G^2 cubic T-Bézier curve approximating the predetermined GLAC with curvature error ≤ 0.05.
Begin
Step 1 Set $i = 1$;
Step 2 Compute the solution region δ as stated in equation (31)
   Set the initial values as $(\xi_1, \omega_1) = (1, 1)$ whenever $(1, 1) \in \delta$, otherwise set any random values satisfy δ close to $(1, 1)$.
   While $i \leq n$ do step 3–7
Step 3 Compute the control points as stated in equation (22)
Step 4 Compute $\lambda(\xi_1, \omega_1)$ as stated in (19);
   $\mu(\xi_1, \omega_1)$ as stated in (20)
Step 5 Use equation (21) to compute the G^2 cubic trigonometric Bézier curve basis functions using the values of the shape parameters
λ and μ gained in step 4.
Step 6 Calculate the curvature error measure as stated in equation (29)
Step 7 If $\dot{e}_{\max} \leq 0.05$ then
   output $r(t)$; (Procedure completed successfully.)
     stop.
   else
     Select random initial values for ξ_1 and ω_1 satisfy the solution region δ and set them as new initial values in step 2, where we
     highly recommended choosing values close to C^2 initial value, 1.
     Return to step 3 and search using the Interior Point method for values to ξ_1 and ω_1 which minimize $\dot{e}_{\max}$ around the maximum
   error point in the previous iteration.
Step 8 Output ("Method failed after n iterations split the GLAC or increase the tolerance 0.05, $n =$ ", n).
   Stop.

ALGORITHM 1: G^2 cubic T-Bézier approximation of generalized log-aesthetic curves.

is large on the first attempt of approximation, where the next iteration itself minimizes the error with tol = 0.05.

Figures 1–3 are related to rows 3, 7, and 8 in Table 1, respectively. The GLAC and its G^2 cubic T-Bézier approximation is shown in 1(a) where the curvature error measure appears in 1(b), the graph of the dependent shape parameters $\lambda(\xi_1, \omega_1)$ and $\mu(\xi_1, \omega_1)$ over the solution region δ are shown in 1(c) and 1(d), respectively.

For the case of example 14, which shown in Figure 4 below, the error was 0.078 obtained with the initial values of $\xi_1 = \omega_1 = 1$, where their corresponding shape parameters are $\lambda = 0.587$ and $\mu = -0.15$. To reduce the number of iterations, we recommended searching around the parameter s where the maximum error occurs. In this example, the maximum error of 0.078 occurs at $s = 0.5054$. By inputting this information during the search with the interior point method, we obtained the desired result on the next run itself where the error is 0.045 and the corresponding shape parameters are $\xi_1 = 0.989$ and $\omega_1 = 0.989$, which led to the calculation of cubic T-Béziers' shape parameters as $\lambda = 0.471$ and $\mu = -0.194$.

For example 15, the error was 0.063 with the initial values of $\xi_1 = \omega_1 = 1$, and its corresponding shape parameters were $\lambda = 0.034$ and $\mu = 0.43$. The maximum error occurs 0.063 occurs at $s = 0.489$. Similar to example 14, we also obtained the desired result on the next run itself where the error is 0.025 and the corresponding shape parameters are $\xi_1 = 0.984$ and $\omega_1 = 0.988$, which led to the calculation of cubic T-Béziers' shape parameters as $\lambda = -0.028$ and $\mu = 0.31$.

9. Comparing the Proposed T-Bézier Method with the Standard Bézier Method

In this section, we are comparing the proposed T-Bézier method with the standard Bézier method in approximating several GLAC with various arc lengths.

Table 2 shows the results of approximating several normalized κ-shift GLAC while Table 3 depicts the results of approximating numerous examples of κ-shift GLAC with several arc lengths. Table 2 shows that the cubic Bézier approximation method does not reach satisfactory approximations without the need of searching for other values of the shape parameters (ξ_1, ω_1). As a consequence, this method constantly replaces the control points of the cubic Bézier approximated curve with better positions to provide increased accuracy. However, the proposed cubic trigonometric Bézier approximation method completes most of the examples in Table 2 with the given initial values without any change in values. This, then, indicates that the proposed cubic trigonometric Bézier approximation method is more efficient than the cubic Bézier approximation method. Furthermore, although both of the cubic methods require 1 iteration to reach an acceptable approximation for GLAC $(\alpha, \Lambda, \nu) = (0.7, -0.6, 0.1)$, the proposed cubic trigonometric Bézier approximation method provides an error of 0.008955 which is more accurate when compared to the cubic Bézier approximation method's error of 0.017444.

For aesthetic curves with various arc lengths, the cubic trigonometric Bézier approximation method needs iteration for more cases compared to the normalized curves.

TABLE 1: The error in approximation for various types of GLAC tested by the error measure $\dot{\epsilon}_{max}$.

| #	(α, Λ, ν)	δ	Final (ξ_1, ω_1)	$(\lambda(\xi_1, \omega_1), \mu(\xi_1, \omega_1))$	$\dot{\epsilon}_{max}$	No of Itrs	Remarks
1	$(0.1, -0.3, 0.4)$	$\{((0, 1.17] \times (0, 1.36]) \parallel ([1.17, 1.47] \times (0, 1.07])\}$	$(1, 1)$	$(0.124, -0.085)$	0.007	0	
2	$(-0.4, -0.8, -0.3)$	$\{((0, 1.36] \times (0, 1.30]) \parallel ([1.36, 1.78] \times (0, 0.96])\}$	$(1, 1)$	$(0.434, -0.107)$	0.014	0	
3	$(0.8, 0.4, 0.7)$	$\{((0, 1.08] \times (0, 1.47]) \parallel ([1.08, 1.37] \times (0, 1.16])\}$	$(1, 1)$	$(-0.057, 0.115)$	0.011	0	
4	$(3, 0.1, 0.4)$	$\{((0, 1.11] \times (0, 1.47]) \parallel ([1.11, 1.44] \times (0, 1.14])\}$	$(1, 1)$	$(0.017, 0.062)$	0.003	0	
5	$(4, 0.5, 0)$	$\{((0, 1.06] \times (0, 1.58]) \parallel ([1.06, 1.44] \times (0, 1.18])\}$	$(1, 1)$	$(0.081, 0.224)$	0.013	0	
6	$(2, 0.5, 1.5)$	$\{((0, 1.05] \times (0, 1.27]) \parallel ([1.05, 1.2] \times (0, 1.11])\}$	$(1, 1)$	$(-0.285, 0.04)$	0.042	0	Circle involute
7	$(2, -0.2, 0)$	$\{((0, 1.18] \times (0, 1.42]) \parallel ([1.18, 1.55] \times (0, 1.07])\}$	$(1, 1)$	$(0.176, 0.023)$	0.008	0	Circle involute
8	$(-1, 0.1, 1)$	$\{((0, 1.10] \times (0.1.35]) \parallel ([1.1, 1.32] \times [0, 1.12])\}$	$(1, 1)$	$(-0.113, -0.039)$	0.008	0	Clothoid
9	$(-1, 0.5, 0)$	$\{((0, 1.01] \times (0, 1.77]) \parallel ([1.01, 1.4] \times (0, 1.31])\}$	$(1, 1)$	$(-0.008, 0.249)$	0.019	0	Clothoid
10	$(1, -0.2, -1)$	$\{((0, 3.2] \times (0, 1.24])\}$	$(1, 1)$	$(0.643, -0.018)$	0.015	0	Logarithmic spiral
11	$(1, 0.5, 0)$	$\{((0, 1.05] \times (0, 1.63]) \parallel ([1.05, 1.42] \times (0, 1.21])\}$	$(1, 1)$	$(0.061, 0.243)$	0.007	0	Logarithmic spiral
12	$(0, 1, 1)$	$\{((0, 1.03] \times (0, 1.49]) \parallel ([1.03, 1.3] \times (0, 1.2])\}$	$(1, 1)$	$(-0.13, 0.286)$	0.038	0	Neilsen's spiral
13	$(0, 0.7, 0)$	$\{((0, 1.00] \times (0, 1.74]) \parallel ([1.0, 1.37] \times (0, 1.28])\}$	$(1, 1)$	$(0.045, 0.331)$	0.006	0	Neilsen's spiral
14	$(1, -0.5, 0.2)$	$\{((0, 1.25] \times (0, 1.25]) \parallel ([1.25, 1.55] \times (0, 0.98])\}$	$(0.989, 0.989)$	$(0.471, -0.194)$	0.045	1	Logarithmic spiral
15	$(1, 1.5, 0.5)$	$\{((0, 1.00] \times (0, 1.59]) \parallel ([1.00, 1.33] \times (0, 1.2\ 2])\}$	$(0.984, 0.988)$	$(-0.028, 0.31)$	0.025	1	Logarithmic spiral
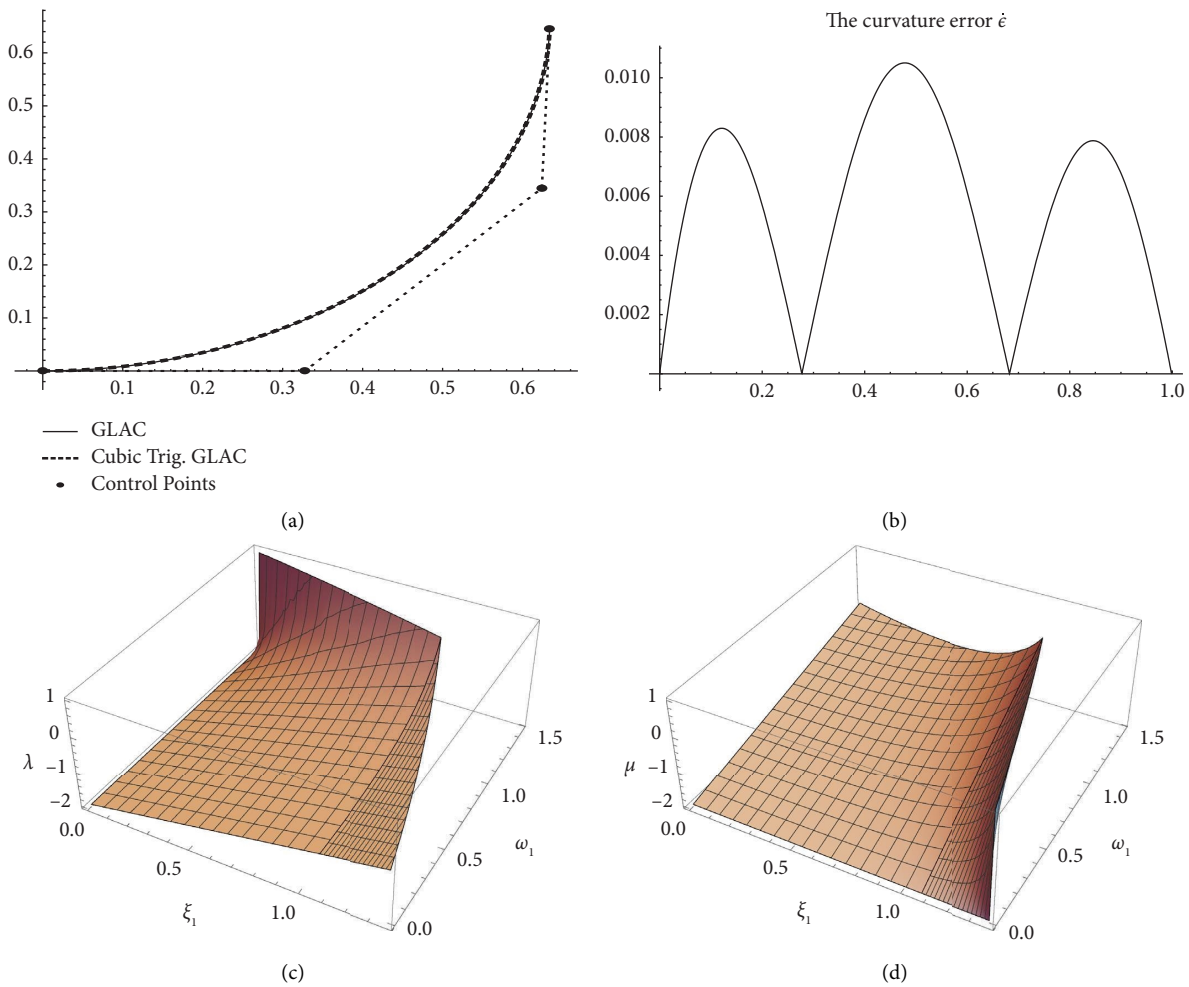
FIGURE 1: (a) The GLAC $(\Lambda, \alpha, \nu) = (0.8, 0.4, 0.7)$ and its approximation. (b) The curvature error. (c) The restricted solution region of (ξ_1, ω_1) with corresponding λ values. (d) The restricted solution region of (ξ_1, ω_1) with corresponding μ values.

However, examples with iterations propose that the trigonometric method provides errors of 0.005753 and 0.015615 for the GLAC $(\alpha, \Lambda, \nu) = (0, 0.5, 0.2)$ and GLAC $(\alpha, \Lambda, \nu) = (2, 0.3, -0.4)$, respectively. This is more accurate than the errors of the cubic Bézier method which are 0.019160 and 0.022365.

Table 4 displays the results of several normalized ρ-shift GLAC approximated curves. Table 5 shows the results of approximating the various ρ-shift GLAC with different arc lengths.

For the normalized ρ-shift GLACs, the cubic Bézier approximation method [21] still requires more iterations to reach satisfactory approximations. It is evident that the cubic trigonometric basis functions of the cubic trigonometric Bézier approximation method contribute to an increase in accurate results compared to the Bernstein polynomial basis functions of the cubic Bézier approximation method. The last example of Table 4 is another example which shows that even though the cubic trigonometric method does not reach

a satisfactory approximation with the initial values, it still produces a result that is more accurate than the standard cubic Bézier method.

For ρ-shift GLAC with various arc lengths, the pattern remains the same. The cubic trigonometric method requires less iteration than the cubic Bézier method for most of the examples while it provides more accurate results.

In general, the cubic T-Bézier takes more time than the standard cubic Bézier as more time is needed for computations of the trigonometric basis functions as compared to the Bernstein polynomial basis functions.

Concerning accuracy, as the cubic trigonometric Bézier curve is closer to the control polygon compared to the standard cubic Bézier curve, the cubic trigonometric Bézier curve has an extra degree of freedom. This explains the advantage related to the accuracy of the proposed GLAC cubic trigonometric Bézier approximated curves over cubic Bézier approximated curves.
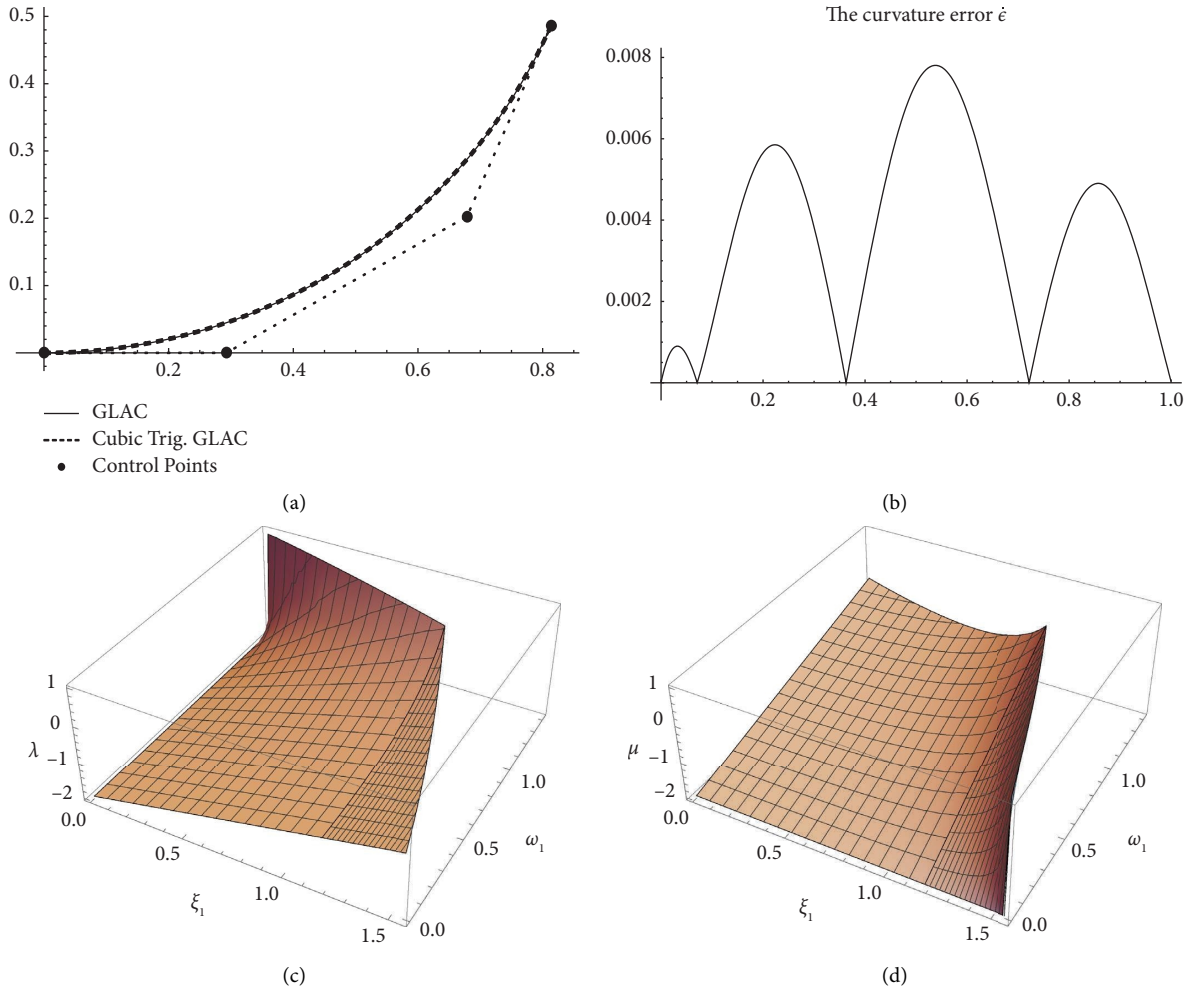
(a)

(b)

(c)

(d)

Figure 2: (a) The GLAC $(\Lambda, \alpha, \nu) = (2, -0.2, 0)$ and its approximation. (b) The curvatures error. (c) The restricted solution region of (ξ_1, ω_1) with corresponding λ values. (d) The restricted solution region of (ξ_1, ω_1) with corresponding μ values.
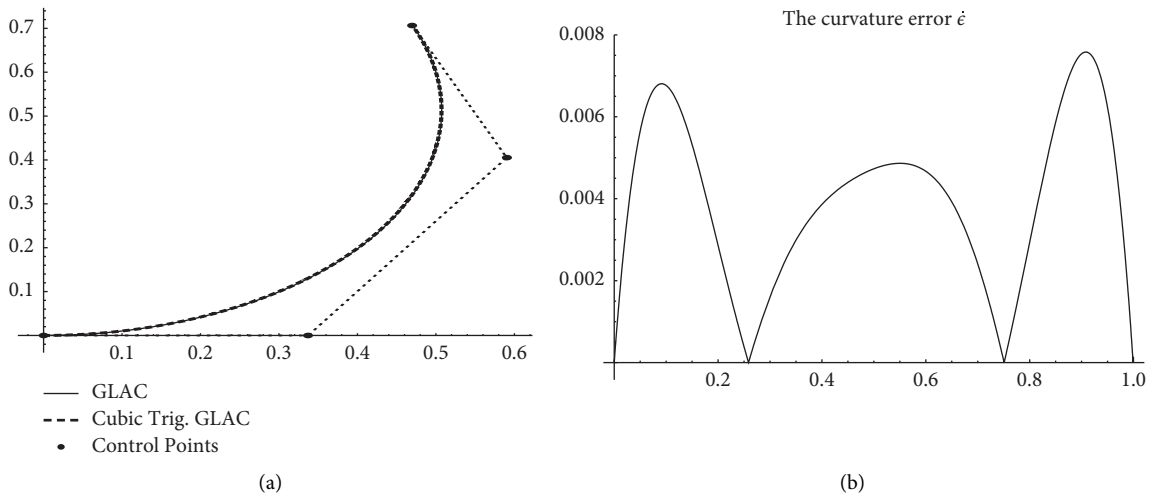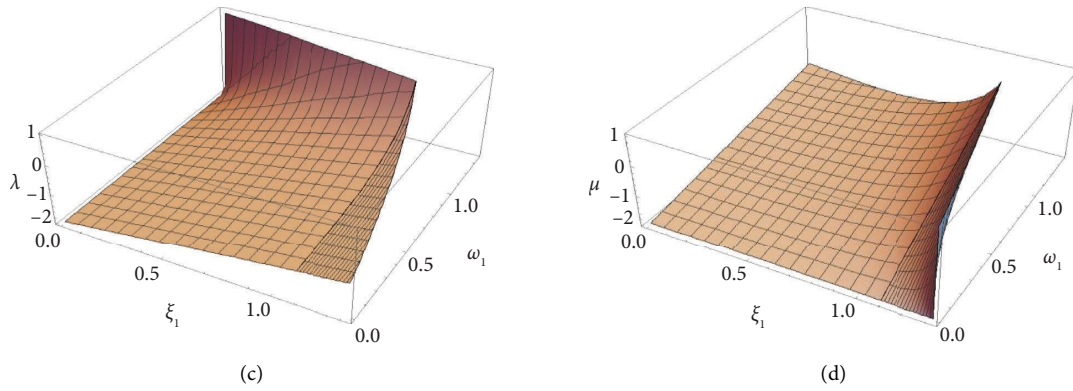

(a)

(b)

Figure 3: Continued.

(c)

(d)

Figure 3: (a) The GLAC $(\Lambda, \alpha, \nu) = (-1, 0.1, 1)$ and its approximation. (b) The curvatures error. (c) The restricted solution region of (ξ_1, ω_1) with corresponding λ values. (d) The restricted solution region of (ξ_1, ω_1) with corresponding μ values.
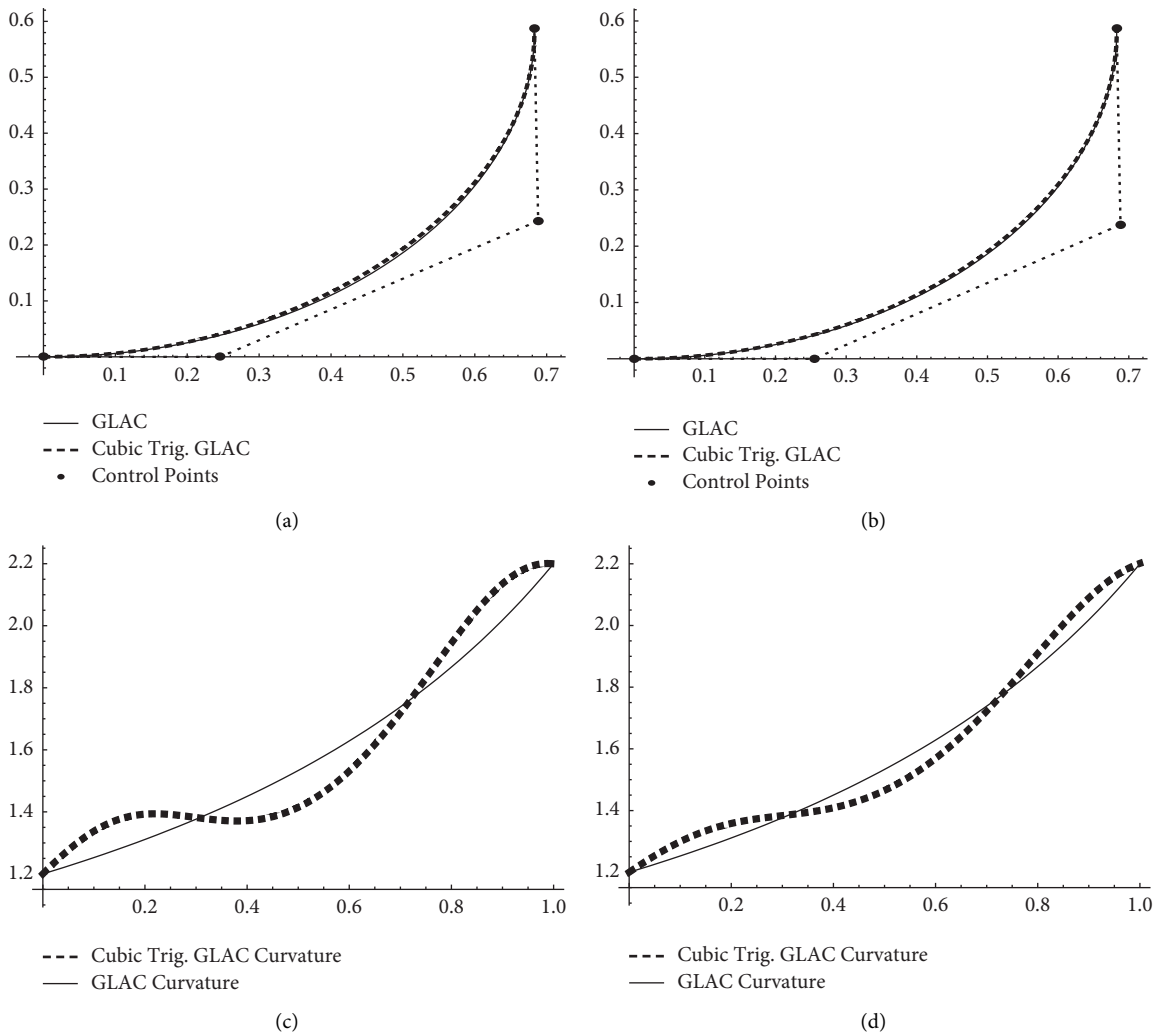

—— GLAC
--- Cubic Trig. GLAC
• Control Points

(a)

—— GLAC
--- Cubic Trig. GLAC
• Control Points

(b)

--- Cubic Trig. GLAC Curvature
—— GLAC Curvature

(c)

--- Cubic Trig. GLAC Curvature
—— GLAC Curvature

(d)

Figure 4: Continued.

Curvature error after 1 iteration
Curvature error calculated at the initial values $\xi_1 = \omega_1 = 1$
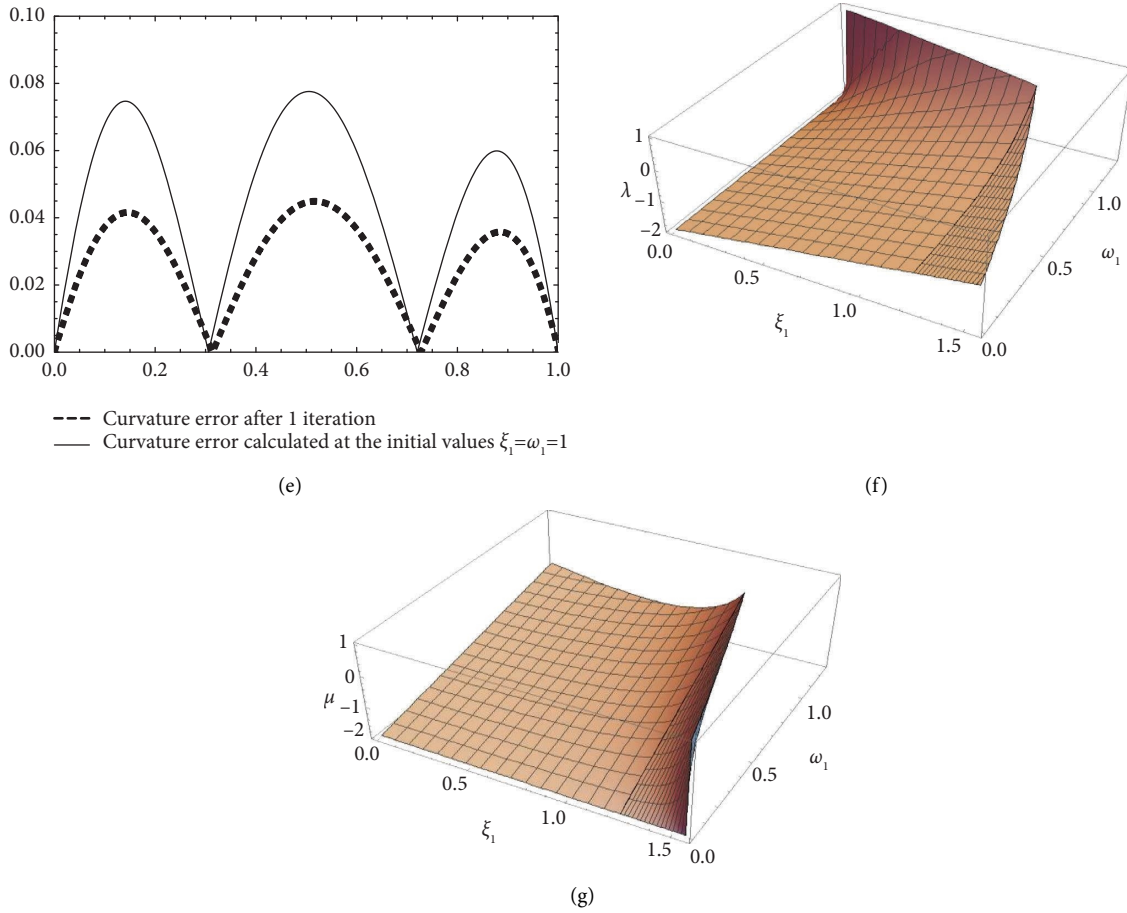
(e)

(f)

(g)

FIGURE 4: (a) The GLAC $(\Lambda, \alpha, \nu) = (1, -0.5, 0.2)$ and its approximation at the first iteration. (b) The same GLAC and its approximation at the second iteration. (c) The curvatures of the GLAC $(\Lambda, \alpha, \nu) = (1, -0.5, 0.2)$ and its approximation at the first iteration. (d) The curvatures of the same GLAC and its approximation at the second iteration. (e) The curvature errors of the GLAC $(\Lambda, \alpha, \nu) = (1, -0.5, 0.2)$ and its approximation at the first and second iterations. (f) λ over the restricted solution region. (g) μ over the restricted solution region.

TABLE 2: Approximations of normalized κ-shift GLAC.

κ-GLAC	Cubic Bézier				Cubic trigonometric Bézier			
(α, Λ, ν)	(ξ_1, ω_1)	#Iterations	$\dot{\epsilon}$	Time (s)	(ξ_1, ω_1)	#Iterations	$\dot{\epsilon}$	Time (s)
$(-1, 0.5, 1)$	$(1, 1)$	0	0.596104	0.3906	$(1, 1)$	0	0.003419	0.8281
	$(0.985, 1.140)$	1	0.025942	0.8125				
$(0, 0.5, 0.5)$	$(1, 1)$	0	0.394663	0.4063	$(1, 1)$	0	0.007401	1.1094
	$(0.955, 1.113)$	1	0.008084	0.8594				
$(0.7, -0.6, 0.1)$	$(1, 1)$	0	0.634895	0.5000	$(1, 1)$	0	0.083813	0.8281
	$(1.242, 0.862)$	1	0.017444	0.8594	$(0.982, 0.970)$	1	0.008955	1.8594
$(1, 0.1, 0.6)$	$(1, 1)$	0	0.441414	0.3750	$(1, 1)$	0	0.000908	0.8281
	$(1.031, 1.062)$	1	0.017211	0.7656				
$(2, -0.1, 0.2)$	$(1, 1)$	0	0.34501	0.4063	$(1, 1)$	0	0.004157	0.8906
	$(1.056, 1.007)$	1	0.007432	0.7969				

TABLE 3: Approximations of κ-shift GLAC with various arc lengths.

κ-GLAC			Cubic Bézier			Cubic trigonometric Bézier			
(α, Λ, ν)	S	(ξ_1, ω_1)	# Iterations	$\dot{\epsilon}$	Time (s)	(ξ_1, ω_1)	# Iterations	$\dot{\epsilon}$	Time (s)
$(-1, 0.2, -0.1)$	3	(3, 3)	0	0.622259	0.3750	(3, 3)	0	0.007819	0.8750
		(2.337, 4.473)	1	0.016135	0.7344				
$(0, 0.5, 0.2)$	2	(2, 2)	0	0.467113	0.3750	(2, 2)	0	0.059382	0.8750
		(1.733, 2.525)	1	0.019160	0.7656	(1.948, 1.971)	1	0.005753	2.0625
$(0.3, 0.2, 0.1)$	2	(2, 2)	0	0.504361	0.3906	(2, 2)	0	0.021884	0.8438
		(1.953, 2.312)	1	0.032646	0.7656				
$(1, 0.1, -0.3)$	3	(3, 3)	0	0.2918	0.3750	(3, 3)	0	0.013707	1.0313
		(2.860, 3.498)	1	0.015333	0.7031				
$(2, 0.3, -0.4)$	5	(5, 5)	0	0.434436	0.3438	(5, 5)	0	0.063259	0.9219
		(2.554, 8.865)	1	0.022365	0.7031	(4.367, 4.663)	1	0.015615	1.7813

TABLE 4: Approximations of normalized ρ-shift GLAC.

ρ-GLAC		Cubic Bézier			Cubic trigonometric Bézier			
(α, Λ, ν)	(ξ_1, ω_1)	#Iterations	$\dot{\epsilon}$	Time (s)	(ξ_1, ω_1)	#Iterations	$\dot{\epsilon}$	Time (s)
$(-1, -0.4, -0.1)$	(1, 1)	0	0.474890	0.5312	(1, 1)	0	0.006546	0.8438
	(1.147, 0.940)	1	0.008330	0.9531				
$(0, 0.6, 0.5)$	(1, 1)	0	0.034375	0.3594	(1, 1)	0	0.007147	0.7344
$(0.9, 0.7, 0.2)$	(1, 1)	0	0.033466	0.5469	(1, 1)	0	0.005927	1.0312
$(1, 0.8, 0.9)$	(1, 1)	0	0.015756	0.4531	(1, 1)	0	0.003125	0.7813
$(2, 0.5, -0.5)$	(1, 1)	0	0.436882	0.5156	(1, 1)	0	0.078244	0.8594
	(0.879, 1.195)	1	0.014967	0.9063		1	0.007277	1.7344

TABLE 5: Approximations of ρ-shift GLAC with various arc lengths.

ρ-GLAC			Cubic Bézier			Cubic trigonometric Bézier			
(α, Λ, ν)	S	(ξ_1, ω_1)	#Iterations	$\dot{\epsilon}$	Time (s)	(ξ_1, ω_1)	#Iterations	$\dot{\epsilon}$	Time (s)
$(-1, 0.1, 0.9)$	4	(4, 4)	0	0.258526	0.5938	(4, 4)	0	0.004819	0.9844
		(3.957, 4.619)	1	0.014342	0.9531				
$(0, 0.1, 0.3)$	3	(3, 3)	0	0.393629	0.3594	(3, 3)	0	0.011349	0.8906
		(3.035, 3.424)	1	0.035227	0.7500				
$(0.5, 0.5, 0.5)$	5	(5, 5)	0	0.454235	0.4063	(5, 5)	0	0.090672	0.4870
		(3.710, 7.212)	1	0.012465	0.7969	(4.557, 4.748)	1	0.009789	1.6250
$(1, 0.7, 0.9)$	2	(2, 2)	0	0.036563	0.3438	(2, 2)	0	0.008822	1.0469
$(2, 0.1, 0.5)$	2	(2, 2)	0	0.209512	0.3594	(2, 2)	0	0.002147	0.7969
		(2.002, 2.123)	1	0.004711	0.7031				

10. Conclusion and Future Work

A new set of G^2 formulas to approximate GLAC with cubic T-Bézier for the two shape parameters λ and μ is derived. These new formulas employ the basis functions of cubic T-Bézier as proposed by Han et al. [11] and the intermediate control points are expressed in the form of GLACs' shape parameters. A solution region is established to ensure the efficiency of the approximation is at optimum. Based on the numerical results, the proposed algorithm may obtain an approximation of GLAC within 0.05 tolerances either on the first or second iteration. Future work includes approximating GLAC with generalized Trigonometric Bézier curves controlled at curvature extrema.

Data Availability

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

Disclosure

The manuscript was already published as an abstract [25].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Acknowledgments

References

[1] R. U. Gobithaasan and K. T. Miura, "Aesthetic spiral for design," *Sains Malaysiana*, vol. 40, no. 11, pp. 1301–1305, 2011.

[2] B. Cross and R. J. Cripps, "G^3 quintic polynomial approximation for generalised Cornu spiral segments," *Journal of Computational and Applied Mathematics*, vol. 236, no. 13, pp. 3111–3122, 2012.

[3] K. T. Miura, "A general equation of aesthetic curves and its self-affinity," *Computer-Aided Design and Applications*, vol. 3, no. 1-4, pp. 457–464, 2006.

[4] N. Yoshida and T. Saito, "Shape information of curves and its visualization using two-tone pseudo coloring," in *Proceedings of the Computer-Aided Design and Applications*, pp. 1–18, Minneapolis, MN, USA, October 2005.

[5] L. Lu and X. Xiang, "Quintic polynomial approximation of log-aesthetic curves by curvature deviation," *Journal of Computational and Applied Mathematics*, vol. 296, pp. 389–396, 2016.

[6] I. J. Schoenberg, "On trigonometric spline interpolation," *Indiana University Mathematics Journal*, vol. 13, no. 5, pp. 795–825, 1964.

[7] X. Han, "Piecewise quadratic trigonometric polynomial curves," *Mathematics of Computation*, vol. 72, no. 243, pp. 1369–1378, 2003.

[8] X. Han, "Cubic trigonometric polynomial curves with a shape parameter," *Computer Aided Geometric Design*, vol. 21, no. 6, pp. 535–548, 2004.

[9] X. Han, "A class of general quartic spline curves with shape parameters," *Computer Aided Geometric Design*, vol. 28, no. 3, pp. 151–163, 2011.

[10] G. Walz, "Trigonometric Bézier and Stancu polynomials over intervals and triangles," *Computer Aided Geometric Design*, vol. 14, no. 4, pp. 393–397, 1997.

[11] X. A. Han, Y. C. Ma, and X. L. Huang, "The cubic trigonometric bézier curve with two shape parameters," *Applied Mathematics Letters*, vol. 22, no. 2, pp. 226–231, 2009.

[12] N. Choubey and A. Ojha, "Constrained curve drawing using trigonometric splines having shape parameters," *Computer-Aided Design*, vol. 39, no. 12, pp. 1058–1064, 2007.

[13] P. E. Koch, T. Lyche, M. Neamtu, and L. L. Schumaker, "Control curves and knot insertion for trigonometric splines," *Advances in Computational Mathematics*, vol. 3, no. 4, pp. 405–424, 1995.

[14] J. M. Peña, "Shape preserving representations for trigonometric polynomial curves," *Computer Aided Geometric Design*, vol. 14, no. 1, pp. 5–11, 1997.

[15] L. L. Schumaker and C. Traas, "Fitting scattered data on spherelike surfaces using tensor products of trigonometric and polynomial splines," *Numerische Mathematik*, vol. 60, no. 1, pp. 133–144, 1991.

[16] A. Kajla, "On the bézier variant of the srivastava-gupta operators," *Constructive Mathematical Analysis*, vol. 1, no. 2, pp. 99–107, 2018.

[17] M. Z. Hussain, S. Abbas, and M. Irshad, "Quadratic trigonometric B-spline for image interpolation using GA," *PLoS One*, vol. 12, no. 6, Article ID 179721, 2017.

[18] D. S. Meek, "Coaxing a planar curve to comply," *Journal of Computational and Applied Mathematics*, vol. 40, no. 1–2, pp. 599–618, 2002.

[19] R. Gobithasan and J. M. Ali, "Towards G^2 curve design with timmer parametric cubic," in *Proceedings-International Conference on Computer Graphics, Imaging and Visuualization*, pp. 109–114, CGIV, Penang, Malaysia, July 2004.

[20] H. Pottmann, "Locally controllable conic splines with curvature continuity," *Association for Computing Machinery Transport. Graphics*, vol. 10, pp. 366–377, 1991.

[21] L. Lu, "A note on quintic polynomial approximation of generalized Cornu spiral segments," *Journal of Computational and Applied Mathematics*, vol. 253, pp. 123–130, 2013.

[22] D.' J. Albayari, *The Approximation of High Quality Curves with Standard and Trigonometric Bezier Curves Using Curvature Error Measures*, PhD Thesis, University of Malaysia Terengganu, Malaysia, 2018.

[23] D.' J. Albayari, R. U. Gobithaasan, and K. T. Miura, "The approximation of generalized Log-aesthetic curves using Quintic Bezier curves," *Journal of Physics: Conference Series*, vol. 890, 2017.

[24] F. Yamaguchi, *Curves and Surfaces in Computer Aided Geometric Design*, Springer-Verlag Berlin Heidelberg, Berlin, Germany, 1988.

[25] R. U. Gobithaasan, D.' J. Albayari, and K. T. Miura, "The approximation of generalized log-aesthetic curves with quintic bezier/trigo bezier curves," in *Proceedings of the Geometric Computation and Applications Workshop 2018*, Trinity College, Dublin, Ireland, November 2018.