RESEARCH ARTICLE

A quadratic trigonometric spline for curve modeling

Shamaila Samreen^{1,2*}, Muhammad Sarfraz³, Malik Zawwar Hussain²

1 Department of Mathematics, University of Engineering and Technology, Lahore, Pakistan, **2** Department of Mathematics, University of the Punjab, Lahore, Pakistan, **3** Department of Information Science, College of Computing Sciences & Engineering, Kuwait University, Safat, Kuwait

* These authors contributed equally to this work.

* shamailasamreen16@gmail.com

Abstract

An imperative curve modeling technique has been established with a view to its applications in various disciplines of science, engineering and design. It is a new spline method using piecewise quadratic trigonometric functions. It possesses error bounds of order 3. The proposed curve model also owns the most favorable geometric properties. The proposed spline method accomplishes C^2 smoothness and produces a Quadratic Trigonometric Spline (QTS) with the view to its applications in curve design and control. It produces a C^2 quadratic trigonometric alternative to the traditional cubic polynomial spline (CPS) because of having four control points in its piecewise description. The comparison analysis of QTS and CPS verifies the QTS as better alternate to CPS. Also, the time analysis proves QTS computationally efficient than CPS.

OPEN ACCESS

Citation: Samreen S, Sarfraz M, Hussain MZ (2019) A quadratic trigonometric spline for curve modeling. PLoS ONE 14(1): e0208015. <https://doi.org/10.1371/journal.pone.0208015>

Editor: Marco Livesu, Consiglio Nazionale delle Ricerche, ITALY

Received: May 13, 2018

Accepted: November 9, 2018

Published: January 10, 2019

Copyright: © 2019 Samreen et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All relevant data are within the paper.

Funding: The authors received no specific funding for this work.

Competing interests: The authors have declared that no competing interests exist.

2. Introduction

Designing curves, especially robust curves, which are controllable, well behaved and easily worked out, contributes a special role in computer graphics and geometric modeling. A number of applications of these vigorous curves in modeling objects, CAD/CAM, font designing, object recognition, medical imaging and fingerprints recognition are the motivations in the direction of curve designing. Besides the applications, the significance of curve designing in computer visualization, robotics and even in broad casting is the inspiration to do research. In the existing literature, a wide-ranging effort has been put in the course of curve modeling. For reference, the readers are directed to [1–25].

It is favorable to adopt an effective, robust, well controlled and visually nice method as a key of many problems on a single platform. In the area of geometric modeling and computer graphics, a cubic polynomial spline (CPS) curve method is supposed to be an adequate approach in the current literature [1–25]. Deliberation of a stable and faster new spline, as a better alternate to CPS, accompanied by piecewise descriptions can be a better proposal for this paper. Thus, providing a quadratic trigonometric spline alternate of a CPS, which not only keeps decent attributes of CPS but is also computationally less expensive, could be a better choice for designers and engineers.

It is worthy to present a method for a class of trigonometric splines with a support of ideal geometric properties and efficiency. Boehm [1] constructed the curvature continuous curves and surfaces. Barsky [2] developed B-splines basis to Beta splines. He preserved the geometric smoothness property of the curve modeling while enabling the conditions of continuity on the splines at different knots to be assorted by certain shape parameters; thus, providing more flexibility. Cline [3] used the splines under tension for the curve fitting. Dierckx and Tytgat [4], presented a proficient procedure to compute the Bézier points of a generalized cubic β -spline curve and showed the connection with multiple knot insertion. They also determined the β -spline vertices for a composite G^2 Bézier curve. Farin [5] has the detail study on curves and surfaces for Computer-Aided Geometric Design (CAGD). Foley [6] constructed the weighted spline to acquire the tension locally. Foley [7] built B-spline like local basis functions for weighted nu-splines. The local basis for GC^2 splines was introduced by Lewis [8]. Nielson [9–10] described the classification of splines with tension and offered the alternates of splines under tension. The developed splines were the generalization of CPS, nu-splines and weighted splines. A general setting was specified by Pruess [11], for smooth interpolation of the splines depending on a parameter. Sarfraz [12–17] constructed various spline techniques for shape modeling using various spline which are a big contribution in literature. Sarfraz [18] built a spline for object designing. The developed spline presents the generalization of Ball's cubic spline. Hussain et al. [19] developed the shape preserving method using rational quadratic trigonometric function. Later Sarfraz et al. [20–23] and Samreen et al. [24] constructed different spline methods for shape designing using a quadratic trigonometric splines with well controlled shape influences of parameters.

This paper presents a quadratic trigonometric spline (QTS) as a better alternate to the CPS. The proposed QTS method is robust, geometrically ideal, and computationally faster. The paper has been organized in various sections. Section 3 offers the QTS and CPS whereas Section 4 calculates the error analysis of QTS using Peano-Kernel Theorem. Section 5 proposes the new C^2 QTS, it is presented here in interpolation form. An efficient and compact algorithm, to compute and plot the curve design, is also part of this section. Geometric properties of the proposed QTS are logically described and proved in Section 6. These properties include partition of unity, positivity, end-point interpolation, convex hull, affine invariance and variation diminishing. Section 7 provides a comparison analysis of the proposed QTS versus CPS. In this regard, area between two spline curves, visual differences, time elapsed of two splines and error analysis of both splines are taken into account. Section 8 is dedicated to specify the advantageous features of the proposed QTS method and analyzes the superiority of QTS over CPS. Finally, Section 9 concludes the paper.

3. The proposed spline

To describe the QTS, assume the data points $F_i \in \mathbb{R}^n$ at the knots $t_i, i = 0, 1, \dots, n-1$, where $t_0 < t_1 < \dots < t_n$. For $t \in [t_i, t_{i+1}]$, $i = 0, 1, \dots, n-1$, let $V_i, W_i \in \mathbb{R}^n$ and

$$P_i(t) = (1 - \sin\theta)^2 F_i + 2(1 - \sin\theta)\sin\theta V_i + 2(1 - \cos\theta)\cos\theta W_i + (1 - \cos\theta)^2 F_{i+1} \quad (1)$$

where, $\theta = \theta(t) = \left(\frac{t-t_i}{h_i}\right)\frac{\pi}{2}$, $h_i = t_{i+1} - t_i$, $0 \leq \theta \leq \frac{\pi}{2}$, and

$$V_i = F_i + \frac{h_i}{\pi} D_i, \quad W_i = F_{i+1} - \frac{h_i}{\pi} D_{i+1}. \quad (2)$$

Also,

$$P(t_i) = F_i, \quad P'(t_i) = D_i, \quad P'(t_{i+1}) = D_{i+1}, \quad P(t_{i+1}) = F_{i+1}. \quad (3)$$

Then, eventually piecewise QTS behaves like a Hermite CPS. The Hermite CPS, defined for $t \in [t_i, t_{i+1}]$, $t_i < t_{i+1}$, $\forall i = 0, 1, \dots, n-1$, with the data points F_i and V_i , $W_i \in \mathbb{R}^n$, is described by:

$$P_i(t) = (1 - \theta)^3 F_i + (1 - \theta)^2 \theta V_i + (1 - \theta) \theta^2 W_i + \theta^3 F_{i+1}, \quad (4)$$

where, $\theta = \theta(t) = \left(\frac{t-t_i}{h_i}\right)$, $h_i = t_{i+1} - t_i$, $0 \leq \theta \leq 1$, and

$$V_i = F_i + \frac{h_i}{3} D_i, \quad W_i = F_{i+1} - \frac{h_i}{3} D_{i+1}. \quad (5)$$

Also, it can be obviously seen, the CPS fulfills the interpolation properties as in (3).

4. Error analysis of QTS

The error of QTS function (1) is calculated in this segment while the function $F(t) \in [t_0, t_n]$ is interpolated. Since the quadratic trigonometric function interpolated locally without losing generality, the error is calculated in the subinterval $[t_i, t_{i+1}]$. Let $P(t)$ be the quadratic trigonometric function of $F(t)$ interpolated in $[t_i, t_{i+1}]$ as defined in (1) then by applying the Peano-Kernel Theorem, $E[f] = F(t) - P(t) = \frac{1}{2} \int_{t_i}^{t_{i+1}} F^3(\tau) E_t[(t - \tau)_+^2] d\tau$ where $E_t[(t - \tau)_+^2]$ is the kernel of integral defined for the quadratic trigonometric function given by

$$E_t[(t - \tau)_+^2] = \begin{cases} u_1(\tau, t), & t_i < \tau < t, \\ v_1(\tau, t), & t < \tau < t_{i+1}, \end{cases} \quad (6)$$

with, $u_1(\tau, t) = (t - \tau)^2 - \{2\cos\theta(1 - \cos\theta)[(t_{i+1} - \tau)^2 - \frac{2h_i}{\pi}(t_{i+1} - \tau)] + (1 - \cos\theta)^2(t_{i+1} - \tau)^2\}$ and $v_1(\tau, t) = -\{2\cos\theta(1 - \cos\theta)[(t_{i+1} - \tau)^2 - \frac{2h_i}{\pi}(t_{i+1} - \tau)] + (1 - \cos\theta)^2(t_{i+1} - \tau)^2\}$.

The proof of the error estimation mainly consisting of two steps; discussing the properties of function $u_1(\tau, t)$ and $v_1(\tau, t)$ and in the second step, computing the absolute values $\int_{t_i}^t |u_1(\tau, t)| d\tau$ and $\int_t^{t_{i+1}} |v_1(\tau, t)| d\tau$.

Step 1: $u_1(\tau, t)$ and $v_1(\tau, t)$, $\tau \in [t_i, t]$, are the quadratic trigonometric functions of τ . Also for all $\theta \in [0, \frac{\pi}{2}]$, $u(t_i, t) = 0$ and $v(t_{i+1}, t) = 0$.

Now to calculate the roots of $u_1(\tau, t)$ and $v_1(\tau, t)$, put $\tau = t$ in Eq (6)

as $u_1(t, t) = (t - t)^2 - \{2\cos\theta(1 - \cos\theta)[(t_{i+1} - t)^2 - \frac{2h_i}{\pi}(t_{i+1} - t)] + (1 - \cos\theta)^2(t_{i+1} - t)^2\}$ and $v_1(t, t) = -\{2\cos\theta(1 - \cos\theta)[(t_{i+1} - t)^2 - \frac{2h_i}{\pi}(t_{i+1} - t)] + (1 - \cos\theta)^2(t_{i+1} - t)^2\}$, where, $t_{i+1} - t = t_{i+1} - t_i + t_i - t = h_i(1 - \frac{2\theta}{\pi}) = h_i(1 - \sigma)$, with $\sigma = \frac{2\theta}{\pi}$, implies that $u_1(t, t) = -\{2\cos\theta(1 - \cos\theta)[h_i^2(1 - \sigma)^2 - \frac{2h_i^2}{\pi}(1 - \sigma)] + (1 - \cos\theta)^2 h_i^2(1 - \sigma)^2\} = v_1(t, t)$.

Let, $(1 - \sigma) - \frac{2}{\pi} = 0$, be an equation defined in θ , its root in $(0, \frac{\pi}{2})$ is

$$\sigma^* = 1 - \frac{2}{\pi}, \quad (7)$$

It implies, $\theta^* = \frac{\pi}{2} - 1$. It can easily be observed that, for $\sigma \leq \sigma^*$, $u_1(t, t) \leq 0$ and for $\sigma \geq \sigma^*$, $u_1(t, t) \geq 0$. Consider $(t_{i+1} - \tau) = (t_{i+1} - t + t - \tau) = h_i(1 - \sigma) + (t - \tau)$. Now rewrite $u_1(\tau, t)$ to observed sign of $u_1(\tau, t)$ in $[t_i, t]$, as $u_1(\tau, t) = (t - \tau)^2(1 - A_2 - A_3) + (t - \tau)2h_i\{\frac{A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\} + (1 - \sigma)h_i^2\{\frac{2A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\}$.

Then the two roots of $u_1(\tau, t)$ are $\tau_1 = \frac{h_i(B+D)}{A} + t$ and $\tau_2 = \frac{h_i(B-D)}{A} + t$, where $A = 1 - A_2 - A_3$, $B = \frac{A_2}{\pi} - (1 - \sigma)(A_2 + A_3)$ and $D = \sqrt{(\frac{A_2}{\pi} - (1 - \sigma)(A_2 + A_3))^2 - (1 - A_2 - A_3)(1 - \sigma)(\frac{2A_2}{\pi} - (1 - \sigma)(A_2 + A_3))}$ with the real values.

Similarly, it can be noted that for $\sigma \leq \sigma^*$, $v_1(t, t) \leq 0$ and for $\sigma \geq \sigma^*$, $v_1(t, t) \geq 0$. To show this sign change behavior rewrite $v_1(\tau, t)$ as $v_1(\tau, t) = (t - \tau)^2(-A_2 - A_3) + (t - \tau)2h_i\{\frac{A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\} + (1 - \sigma)h_i^2\{\frac{2A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\}$. Then the two roots of $v_1(\tau, t)$ are $\tau_1^* = t_{i+1}$ and $\tau_2^* = t_{i+1} - \frac{2h_iA_2}{\pi(A_2 + A_3)}$, using $(t_{i+1} - \tau) = (t_{i+1} - t + t - \tau) = h_i(1 - \sigma) + (t - \tau)$.

Step 2: While $\sigma > \sigma^*$, $u_1(\tau, t) > 0$ and when $\sigma < \sigma^*$, $u_1(\tau, t) < 0$. It can also be noted that $t_i < \tau_2 < \tau_1 < t$. Thus, for $\sigma < \sigma^*$, $u_1(\tau, t) < 0 \forall \tau \in [t_i, t]$,

$$\begin{aligned} \int_{t_i}^t |u_1(\tau, t)| d\tau &= \int_{t_i}^t (-u_1(\tau, t)) d\tau \\ &= -\frac{1}{3}(1 - A_2 - A_3)h_i^3\sigma^3 - \left(\frac{A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right)h_i^3\sigma^2 - (1 - \sigma)h_i^3\sigma\left\{\frac{2A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right\}. \end{aligned} \quad (8)$$

Also for $\sigma > \sigma^*$, $u_1(\tau, t)$ changes sign on both sides of τ_1 and τ_2 , so

$$\begin{aligned} \int_{t_i}^t |u_1(\tau, t)| d\tau &= \int_{t_i}^{\tau_2} u_1(\tau, t) d\tau + \int_{\tau_2}^{\tau_1} (-u_1(\tau, t)) d\tau + \int_{\tau_1}^t u_1(\tau, t) d\tau \\ &= \frac{1}{3}(1 - A_2 - A_3)\left[-h_i^3\sigma^3 + 2h_i^3\left(\frac{B - D}{A}\right)^3 - 2h_i^3\left(\frac{B + D}{A}\right)^3\right] + h_i\left\{\frac{A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right\}\left[h_i^2\sigma^2 - 2h_i^2\left(\frac{B - D}{A}\right)^2 + 2h_i^2\left(\frac{B + D}{A}\right)^2\right] \\ &\quad + (1 - \sigma)h_i^2\left\{\frac{2A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right\}\left[h_i\sigma + 2h_i\left(\frac{B - D}{A}\right) - 2h_i\left(\frac{B + D}{A}\right)\right]. \end{aligned} \quad (9)$$

Similarly, it can be displayed easily that when $\sigma \leq \sigma^*$, $v_1(\tau, t)$ moves from negative side to positive side of τ_1^* , and while $\sigma \geq \sigma^*$, $v_1(\tau, t) > 0$ in (t, t_{i+1}) , where σ^* is given by Eq (7). Thus while taking $\sigma \leq \sigma^*$

$$\begin{aligned} \int_t^{t_{i+1}} |v_1(\tau, t)| d\tau &= \int_t^{\tau_2^*} (-v_1(\tau, t)) d\tau + \int_{\tau_2^*}^{t_{i+1}} v_1(\tau, t) d\tau \\ &= \left(-\frac{1}{3}(A_2 + A_3)\right)\left[2\left(h_i(\sigma - 1) + \frac{2h_iA_2}{\pi(A_2 + A_3)}\right)^3 - h_i^3(\sigma - 1)^3\right] + h_i\left\{\frac{A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right\}\left[2\left(h_i(\sigma - 1) + \frac{2h_iA_2}{\pi(A_2 + A_3)}\right)^2 - h_i^2(\sigma - 1)^2\right] \\ &\quad + (1 - \sigma)h_i^2\left\{\frac{2A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right\}\left[h_i(\sigma - 1) + \frac{4h_iA_2}{\pi(A_2 + A_3)}\right], \end{aligned} \quad (10)$$

but while $\sigma \geq \sigma^*$,

$$\begin{aligned} \int_t^{t_{i+1}} |v_1(\tau, t)| d\tau &= \int_t^{t_{i+1}} v_1(\tau, t) d\tau \\ &= -\frac{1}{3}(A_2 + A_3)h_i^3(1 - \sigma)^3 - h_i\left\{\frac{A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right\}h_i^2(1 - \sigma)^2 + (1 - \sigma)h_i^2\left\{\frac{2A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right\}h_i(1 - \sigma). \end{aligned} \quad (11)$$

So, combining (8) and (10), the following result holds for $\sigma \leq \sigma^*$, $|F(t) - P(t)| \leq \frac{\|F^{(3)}\|}{2} \int_{t_i}^{t_{i+1}} |R_i[(t - \tau)_+^2]| d\tau = \|F^{(3)}\| h_i^3 \gamma^*(\sigma)$, where $\gamma^*(\sigma)$ is defined by

$$\begin{aligned} \gamma^*(\sigma) &= \int_{t_i}^t (-u_1(\tau, t)) d\tau + \int_t^{\tau_2^*} (-v_1(\tau, t)) d\tau + \int_{\tau_2^*}^{t_{i+1}} v_1(\tau, t) d\tau \\ &= -\frac{1}{3}(1 - A_2 - A_3)h_i^3\sigma^3 - \left(\frac{A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right)h_i^3\sigma^2 - (1 - \sigma)h_i^3\sigma \left\{\frac{2A_2}{\pi} \right. \\ &\quad \left. - (1 - \sigma)(A_2 + A_3)\right\} + \left(-\frac{1}{3}(A_2 + A_3)\right) \left[2(h_i(\sigma - 1) + \frac{2h_iA_2}{\pi(A_2 + A_3)})^3 \right. \\ &\quad \left. - h_i^3(\sigma - 1)^3\right] + h_i \left\{\frac{A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right\} \left[2(h_i(\sigma - 1) + \frac{2h_iA_2}{\pi(A_2 + A_3)})^2 \right. \\ &\quad \left. - h_i^2(\sigma - 1)^2\right] + (1 - \sigma)h_i^2 \left\{\frac{2A_2}{\pi} - (1 - \sigma)(A_2 + A_3)\right\} [(h_i(\sigma - 1) \\ &\quad + \frac{4h_iA_2}{\pi(A_2 + A_3)})], \end{aligned} \quad (12)$$

and the result obtained for $\sigma \geq \sigma^*$, by combining (9) and (11), $|F(t) - P(t)| \leq \frac{\|F^{(3)}\|}{2} \int_{t_i}^{t_{i+1}} |R_i[(t - \tau)_+^2]| d\tau = \|F^{(3)}\| h_i^3 \gamma_*(\sigma)$, where,

$$\begin{aligned} \gamma_*(\sigma) &= \int_{t_i}^{\tau_2} u_1(\tau, t) d\tau + \int_{\tau_2}^{\tau_1} (-u_1(\tau, t)) d\tau + \int_{\tau_1}^t u_1(\tau, t) d\tau + \int_t^{t_{i+1}} v_1(\tau, t) d\tau \\ &= \frac{1}{3}(1 - A_2 - A_3) \left[-h_i^3\sigma^3 + 2h_i^3\left(\frac{B-D}{A}\right)^3 - 2h_i^3\left(\frac{B+D}{A}\right)^3\right] + h_i \left\{\frac{A_2}{\pi} - (1 - \sigma)(A_2 \right. \\ &\quad \left. + A_3)\right\} \left[h_i^2\sigma^2 - 2h_i^2\left(\frac{B-D}{A}\right)^2 + 2h_i^2\left(\frac{B+D}{A}\right)^2\right] + (1 - \sigma)h_i^2 \left\{\frac{2A_2}{\pi} - (1 - \sigma)(A_2 \right. \\ &\quad \left. + A_3)\right\} \left[h_i\sigma + 2h_i\left(\frac{B-D}{A}\right) - 2h_i\left(\frac{B+D}{A}\right)\right] - \frac{1}{3}(A_2 + A_3)h_i^3(1 - \sigma)^3 - h_i \left\{\frac{A_2}{\pi} \right. \\ &\quad \left. - (1 - \sigma)(A_2 + A_3)\right\} h_i^2(1 - \sigma)^2 + (1 - \sigma)h_i^2 \left\{\frac{2A_2}{\pi} \right. \\ &\quad \left. - (1 - \sigma)(A_2 + A_3)\right\} h_i(1 - \sigma). \end{aligned} \quad (13)$$

The above analysis yields the following results.

Theorem 1. Let $P(t)$ be the quadratic trigonometric function, as defined in (1). The error estimation of $P(t)$ holds the followings:

$$|F(t) - P(t)| \leq \|F^{(3)}\| h_i^3 d^*, \text{ where, } d^* = \max_{0 \leq \theta \leq 1} \gamma(\sigma), \text{ and } \gamma(\theta) = \begin{cases} \gamma^*(\sigma) & 0 \leq \sigma \leq \sigma^*, \\ \gamma_*(\sigma) & \sigma_* \leq \sigma \leq 1, \end{cases}$$

where $\gamma^*(\sigma)$ and $\gamma_*(\sigma)$ are respectively, taken from (12) and (13).

Remark 1. The coefficient of error term, d^* can be defined as:

$$d^* = \max\{\max_{0 \leq \theta \leq 1/2} \gamma^*(\sigma), \max_{1/2 \leq \theta \leq 1} \gamma_*(\sigma)\}. \quad (14)$$

5. Formation of C^2 QTS

As seen in Section 2, the QTS is C^1 by its construction. A C^2 QTS can be formed by applying C^2 continuity at the joints of curve segments as follows:

$$P_i''(t_i) = P_{i-1}''(t_i). \quad (15)$$

From the second order derivative of (1), we simply achieve the followings:

$$P_i''(t_i) = \frac{\pi^2}{2h_i^2} F_i - \frac{\pi^2}{h_i^2} V_i + \frac{\pi^2}{2h_i^2} W_i, \quad (16)$$

and

$$P_{i-1}''(t_i) = \frac{\pi^2}{2h_i^2} F_i + \frac{\pi^2}{2h_i^2} V_{i-1} - \frac{\pi^2}{h_i^2} W_{i-1}. \quad (17)$$

Let

$$\Delta_i = \frac{F_{i+1} - F_i}{h_i}, \quad (18)$$

then, the Eqs (16) and (17), respectively, crop to:

$$P_i''(t_i) = \frac{\pi}{2h_i} (\Delta_i \pi - 2D_i - D_{i+1}), \quad (19)$$

and

$$P_{i-1}''(t_i) = \frac{\pi}{2h_{i-1}} (-\Delta_{i-1} \pi + D_{i-1} + 2D_i). \quad (20)$$

Also

$$P_{i-1}'(t_i) = D_i = P_i'(t_i), \quad (21)$$

Thus, using (19), (20) and (21) in (14), a tri-diagonal system of consistency equations is acquired by the followings:

$$\frac{1}{2h_{i-1}} D_{i-1} + \left(\frac{1}{h_{i-1}} + \frac{1}{h_i} \right) D_i + \frac{1}{2h_i} D_{i+1} = \frac{\pi}{2h_i} \Delta_i + \frac{\pi}{2h_{i-1}} \Delta_{i-1}. \quad (22)$$

The above system can be expressed by the following matrix:

$$\begin{bmatrix} \frac{1}{h_1} + \frac{1}{h_0} & \frac{1}{2h_1} & 0 \\ \frac{1}{2h_1} & \frac{1}{h_2} + \frac{1}{h_1} & \frac{1}{2h_1} \\ \ddots & \ddots & \ddots \\ 0 & \frac{1}{2h_{n-2}} & \frac{1}{h_{n-1}} + \frac{1}{h_{n-2}} \end{bmatrix} \begin{bmatrix} D_1 \\ D_2 \\ \vdots \\ D_{n-1} \end{bmatrix} = \begin{bmatrix} \frac{\pi}{2h_1} \Delta_1 + \frac{\pi}{2h_0} \Delta_0 \\ \frac{\pi}{2h_2} \Delta_2 + \frac{\pi}{2h_1} \Delta_1 \\ \vdots \\ \frac{\pi}{2h_{n-1}} \Delta_{n-1} + \frac{\pi}{2h_{n-2}} \Delta_{n-2} \end{bmatrix} \quad (23)$$

It is supposed that end conditions D_0 and D_n are given. One can choose the arbitrary end conditions by choice. Overall, for open curves, the end conditions D_0 and D_n are determined

using the given data points as follows:

$$D_0 = 2(F_1 - F_0) - \frac{F_2 - F_0}{2}, \quad (24)$$

$$D_n = 2(F_n - F_{n-1}) - \frac{F_n - F_{n-2}}{2}. \quad (25)$$

For closed curves, the following periodic end conditions $D_0 = D_n$ are assumed:

$$F_n = F_0, F_{n-1} = F_{-1}, F_{n+1} = F_1. \quad (26)$$

Thus, for the appropriate end conditions, the above system (22) is a tri-diagonal linear system and is diagonally dominant. It also retains unique solutions for D_i 's and hence a unique solution for QTS. To solve the tri-diagonal system for D_i 's, it is efficient to accomplish the LU-decomposition method. Thus, the above conversation can be concise as follows:

Theorem 2. The C^2 QTS exists and has a unique solution.

5.1. Algorithm design

The above discussion is summarized in some steps here. It is done by designing a suitable algorithm for the curve modeling applications. In this way, a user may have an open choice to play with a curve at its own satisfaction level. It can be achieved by an appropriate algorithm design as follows:

Step 1: Input control points F_i 's.

Step 2: Compute the tangents D_i 's, from the data points in Step 1, using system of Eqs (23–26).

Step 3: Compute the QTS curve (1) of Section 3.

This is a simple algorithm to implement QTS. The algorithm for CPS is also similar and hence not been mentioned here. Both of CPS and QTS have been implemented using Matlab [25] software.

5.2. Data of objects

The data of various objects drawn in the Figs is given in Table 1.

For the chosen data points $F_i \in \mathbb{R}^2, i = 1, \dots, n$, C^2 QTSs are demonstrated in Fig 1A–1F.

6. Geometric properties of QTS

The QTS owns several ideal geometric properties discussed by the propositions as follow:

Proposition 1 (Convex Hull Property (CHP)): The QTS curve lies completely inside the convex hull determined by its control polygon.

Proof: By re-writing Eq (1) as:

$$P_i(t) = B_0(\theta)F_i + B_1(\theta)V_i + B_2(\theta)W_i + B_3(\theta)F_{i+1}, \quad (27)$$

where $B_j(\theta) \geq 0, j = 0, \dots, 3$, are Bernstein Bézier like functions and $\sum_{j=0}^3 B_j(\theta) = 1$. Therefore, the QTS curve lies within the convex hull as shown in Fig 2.

Proposition 2 (Affine Invariance Property (AIP)): Let $P(t) = \sum_{j=0}^3 d_j B_j(t)$ be the C^2 QTS curve and $d_j = \{F_i, V_i, W_i, F_{i+1}\} \in \mathbb{R}^n$ be the control points. Then C^2 QTS curve is invariant under affine transformation.

Table 1. Table of data of objects.

#	Different Objects.	Data of the objects	
(1)	Circle	x	0 1 2 1
		y	1 0 1 2
(2)	Lamp	x	0 0.5 .2 .2 .5 -.2 -.2 -.2 1.7 3.5 1.7 1 1.3 1.3 1 1.5 1.5
		y	0 .5 1 1.5 2 2.5 3 3 5 5 3 3 2.5 2 1.5 1 .5 0
(3)	Car	x	.5 1.4 1.7 2.4 2.9 4.6 5 5.5 5.9 7 7.3 6.4 5.5 2.9 2.3 0.5
		y	.5 .5 0.9 0.9 .5 .5 .9 .9 .5 .5 2 2 2.5 2.5 1.8 1.5
(4)	Flower	x	0 1 2 1 2 1 0 1
		y	0 1 0 1 2 1 2 1
(5)	Guitar	x	1.5 3 4 5.5 6 5 5.5 5 4 4 3 3 3 2 1.5 2 1
		y	1 .5 .5 1 3 4 5 6.2 7 9 12 12 9 7 6.2 5 4 3
(6)	Vase	x	2.5 1 2.3 2.3 0 1 4 5 2.7 2.7 4
		y	4 3.5 3.5 2.5 1.5 0 0 1.5 2.5 3.5 3.5
(7)	'A' alphabet	x	0 1 1.2 1.8 2 3 1.5
		y	0 0 1 1 0 0 5.5

<https://doi.org/10.1371/journal.pone.0208015.t001>

Proof: Let, $(x_1, y_1) = (a_1x + a_2y + a_3, b_1x + b_2y + b_3)$, be an affine transformation and $d_j(l_j, m_j)$, $j = 0, \dots, 3$ be the control points of the C^2 QTS curve $P(t) = \sum_{j=0}^3 d_j B_j(t)$ for $t \in [t_i, t_{i+1}]$. Then, $P(t) = (x(t), y(t)) = (\sum_{j=0}^3 l_j B_j(t), \sum_{j=0}^3 m_j B_j(t))$.

Now, $\mathcal{L}(P(t)) = \left(\begin{array}{c} a_1 \sum_{j=0}^3 l_j B_j(t) + a_2 \sum_{j=0}^3 m_j B_j(t) \\ + a_3, b_1 \sum_{j=0}^3 l_j B_j(t) + b_2 \sum_{j=0}^3 m_j B_j(t) + b_3 \end{array} \right)$. As we have $\sum_{j=0}^3 B_j(t) = 1$ for $t \in [t_i, t_{i+1}]$. Thus, $\mathcal{L}(P(t)) = \left(\begin{array}{c} a_1 \sum_{j=0}^3 l_j B_j(t) + a_2 \sum_{j=0}^3 m_j B_j(t) + a_3 \sum_{j=0}^3 B_j(t), \\ b_1 \sum_{j=0}^3 l_j B_j(t) + b_2 \sum_{j=0}^3 m_j B_j(t) + b_3 \sum_{j=0}^3 B_j(t) \end{array} \right) = \sum_{j=0}^3 (a_1 l_j + a_2 m_j + a_3) B_j(t), \sum_{j=0}^3 (b_1 l_j + b_2 m_j + b_3) B_j(t) = \sum_{j=0}^3 (a_1 l_j + a_2 m_j + a_3, b_1 l_j + b_2 m_j + b_3) B_j(t) = \sum_{j=0}^3 \mathcal{L}(d_j) B_j(t)$.

The AIP proved in Proposition 2, is illustrated in Fig 3.

Proposition 3 (Variation Diminishing Property (VDP)): Consider the QTS curve for $t \in [t_i, t_{i+1}]$, $P(t) = \sum_{j=0}^3 d_j B_j(t)$ having control points $d_j = \{F_i, V_i, W_i, F_{i+1}\} \in \mathbb{R}^n$. Then any $N-1$

Fig 1. The data points of different objects (a) Circle, (b) Lamp, (c) Car, (d) Flower, (e) Guitar, (f) Vase, are interpolated by QTS with periodic end conditions.

<https://doi.org/10.1371/journal.pone.0208015.g001>

Fig 2. CHP using Flower data and its QTS.

<https://doi.org/10.1371/journal.pone.0208015.g002>

dimensional plane will intersect the QTS curve no more times than it will intersect the control polygon.

The VDP is displayed, in Fig 4(A), (B) and (C), for different object data of Butterfly, Fish, and Car respectively. The interpolated QTS curves behave positively in all the model curves.

7. Comparison analysis

In this section, a brief comparison analysis of QTS and CPS is discussed. In this regard area of each spline, area between two spline curves QTS and CPS, visual differences of QTS and CPS, time elapsed of the two splines and error analysis of both splines are taken into account.

7.1. Area covered by CPS

The area of the CPS, defined in the interval $[t_i, t_{i+1}]$, is derived as follows:

$\int_{t_i}^{t_{i+1}} P_i(t) dt = \int_{t_i}^{t_{i+1}} [(1 - \theta)^3 F_i + (1 - \theta)^2 \theta V_i + (1 - \theta) \theta^2 W_i + (1 - \theta)^3 F_{i+1}] dt$. Here,

$\theta = \theta(t) = \left(\frac{t - t_i}{h_i}\right)$, $h_i = t_{i+1} - t_i$, $0 \leq \theta \leq 1$, and so, $d\theta = \frac{1}{h_i} dt$. Thus,

$$\begin{aligned} \int_{t_i}^{t_{i+1}} P_i(t) dt &= h_i \int_0^1 [(1 - \theta)^3 F_i + (1 - \theta)^2 \theta V_i + (1 - \theta) \theta^2 W_i + \theta^3 F_{i+1}] d\theta \\ &= \frac{h_i}{4} (F_i + V_i + W_i + F_{i+1}), \end{aligned} \quad (28)$$

7.2. Area covered by QTS

The QTS, defined by (1) in the interval $[t_i, t_{i+1}]$, has the area to be derived as follows:

$\int_{t_i}^{t_{i+1}} P_i(t) dt = \int_{t_i}^{t_{i+1}} [(1 - \sin\theta)^2 F_i + 2(1 - \sin\theta) \sin\theta V_i + 2(1 - \cos\theta) \cos\theta W_i + (1 - \cos\theta)^2 F_{i+1}] dt$,

Fig 3. AIP using Data of Letter "A" and its corresponding QTS with (a) translation, (b) scaling, (c) rotation with angle π and (d) shearing.

<https://doi.org/10.1371/journal.pone.0208015.g003>

Fig 4. VDP for various object data and their corresponding QTSs: (a) Butterfly, (b) Fish, (c) Car.

<https://doi.org/10.1371/journal.pone.0208015.g004>

since, $\theta = \theta(t) = \left(\frac{t-t_i}{h_i}\right)\frac{\pi}{2}$, $h_i = t_{i+1} - t_i$, $0 \leq \theta \leq \frac{\pi}{2}$, so, $d\theta = \frac{\pi}{2h_i} dt$. Thus,

$$\begin{aligned} \int_{t_i}^{t_{i+1}} P_i(t) dt &= \frac{2h_i}{\pi} \int_0^{\frac{\pi}{2}} [(1 - \sin\theta)^2 F_i + 2(1 - \sin\theta)\sin\theta V_i + 2(1 - \cos\theta)\cos\theta W_i + (1 \\ &\quad - \cos\theta)^2 F_{i+1}] d\theta \\ &= \frac{2h_i}{\pi} \left(\left(\frac{3\pi}{4} - 2\right) F_i + \left(2 - \frac{\pi}{2}\right) V_i + \left(2 - \frac{\pi}{2}\right) W_i + \left(\frac{3\pi}{4} - 2\right) F_{i+1} \right) \\ &= h_i \left(\left(\frac{3}{2} - \frac{4}{\pi}\right) F_i + \left(\frac{4}{\pi} - 1\right) V_i + \left(\frac{4}{\pi} - 1\right) W_i + \left(\frac{3}{2} - \frac{4}{\pi}\right) F_{i+1} \right) \\ &= h_i (.2267 F_i + .2732 V_i + .2732 W_i + .2267 F_{i+1}). \end{aligned} \quad (29)$$

7.3. Area analysis of QTS and CPS

It can be noticed from (28) and (29) that mathematically, the areas of the CPS and QTS are two different entities. This difference can also be noticed visually. For example, the QTS and the CPS are interpolated using the functions $f = \sin(t)$, $\tan(t)$ and e^t . The area plots and the area between these two spline curves are demonstrated for area analysis of both splines.

In Figs 5(A), 6(A) and 7(A), CPS interpolates the functions $f = \sin(t)$, $\tan(t)$ and e^t , respectively, while in Figs 5(B), 6(B) and 7(B), these functions are interpolated by QTS. Figs 5(C), 6(C) and 7(C) demonstrate the area plots of the functions $f = \sin(t)$, $\tan(t)$ and e^t , through the CPS, while Figs 5(D), 6(D) and 7(D) demonstrate the area plots of these functions by QTS. Finally, in the Figs 5(E), 6(E) and 7(E), area differences between two the splines CPS and QTS are demonstrated.

7.4. Visual difference between the CPS and QTS

The QTS interpolates almost like the CPS. However, because of different areas covered by them, their visual outputs are slightly variant from each other. To view the differences between

Fig 5. Area Analysis of QTS and CPS using $f = \sin(t)$: (a) Plot of CPS, (b) Plot of QTS, (c) Area plot of CPS, (d) Area plot of QTS, and (e) Area between CPS and QTS.

<https://doi.org/10.1371/journal.pone.0208015.g005>

Fig 6. Area Analysis of QTS and CPS using $f = \tan(t)$: (a) Plot of CPS, (b) Plot of QTS, (c) Area plot of CPS, (d) Area plot of QTS, and (e) Area between CPS and QTS.

<https://doi.org/10.1371/journal.pone.0208015.g006>

the two splines CPS and QTS, the data of distinct existing objects is interpolated to produce design curves.

In Fig 8(A), data of the letter “A” is interpolated by CPS, while in Fig 8(B), the same object is interpolated using QTS. One can easily judge that the QTS is showing visually better smoothness than the CPS in almost every piece of curve. The curve in QTS is piece wisely more bulged outward as compared to its CPS counterpart.

Like in Fig 8, one can notice similar to the behavior in Fig 9 too. In Fig 9(A), data of “Lamp” is interpolated by CPS, while in Fig 9(B), it is interpolated using QTS. It is very clearly judgable that the QTS shows visually better smoothness than the CPS in almost every piece of curve. The curve in QTS is piece wisely more bulged outward as compared to its CPS counterpart.

Fig 7. Area Analysis of QTS and CPS using $f = e^t$: (a) Plot of CPS, (b) Plot of QTS, (c) Area plot of CPS, (d) Area plot of QTS, and (e) Area between CPS and QTS.

<https://doi.org/10.1371/journal.pone.0208015.g007>

Fig 8. Data of the letter “A” is interpolated: (a) by CPS; (b) by QTS.

<https://doi.org/10.1371/journal.pone.0208015.g008>

In Fig 10(A) and Fig 10(B), data of the “Car” is interpolated by CPS and QTS respectively. The QTS is showing visually smoother and bulging attitude in every piece of curve as compared to the CPS.

Adding another example of a Vase in Fig 11, does not surprise to previous figures demonstrations. Fig 11(A) and Fig 11(B) are interpolated curves by CPS and QTS respectively. The QTS, as usual, is showing visually smoother and bulging attitude in every piece of curve as compared to the CPS.

7.5. Time elapsed by the two splines

The comparison analysis that the QTS is superior to CPS is also justified by comparing the time of execution of both the splines. Data of nine different objects, see Table 2, have been chosen for an extensive study.

In Table 2, different computed timings have been calculated for a variety of curves for data of nine objects mentioned in Column 2. The data in Column 3 represents the time elapsed by CPS, Column 4 represents the time elapsed by QTS, and Column 5 represents the difference of the times elapsed between CPS and QTS. It is very obvious to observe that the time elapsed by QTS is much smaller in quantity than that of the CPS. This difference is very significant as can

Fig 9. Data of a lamp is interpolated: (a) by CPS; (b) by QTS.

<https://doi.org/10.1371/journal.pone.0208015.g009>

Fig 10. Data of a car is interpolated: (a) by CPS; (b) by QTS.

<https://doi.org/10.1371/journal.pone.0208015.g010>

be seen in Column 5 of Table 2. Hence, with a variety of experimentations, in Table 2, it is not difficult to decide that the proposed QTS is much faster curve interpolation scheme than the traditional CPS. Hence, QTS is a better substitute to CPS.

7.6. Error analysis for CPS and QTS

The error calculated by both CPS and QTS is defensible for comparing these splines. It is presented in Table 3 for six distinct trigonometric, logarithmic and polynomial functions $\sin(t)$, $\cos(t)$, $\tan(t)$, $\sec(t)$, $\log(t)$ and $\sqrt{t+6} + (t+2)^2$ shown in column 3. The domain of the

Fig 11. Data of a vase is interpolated: (a) by CPS; (b) by QTS.

<https://doi.org/10.1371/journal.pone.0208015.g011>

Table 2. Table of time elapsed of both the splines.

#	Data Used	Time elapsed by CPS	Time elapsed by QTS	Difference between the times elapsed
		(in seconds)		
1	Fish	0.966504	0.328712	0.637792
2	Butterfly	0.494404	0.235974	0.258430
3	Car	0.877330	0.269906	0.607424
4	Lamp	0.850372	0.291029	0.559343
5	Guitar	1.024333	0.401127	0.623206
6	Letter 'A'	0.561426	0.426110	0.135316
7	Flower	0.313687	0.230219	0.083468
8	Vase	0.435195	0.232384	0.202811
9	Ellipses	0.483795	0.245647	0.238148

<https://doi.org/10.1371/journal.pone.0208015.t002>

Table 3. Table of error analysis, for CPS and QTS, for a variety of functions.

#	Value of t (taken)	Function used	Error calculated by CPS	Error calculated by QTS	Difference of Errors calculated by both splines
1	$-\pi \leq t \leq \pi$	$\sin(t)$	1.4439e-013	1.1924e-013	2.52E-14
2	$-\pi \leq t \leq \pi$	$\cos(t)$	1.2434e-014	1.2434e-014	0.000000
3	$-\pi \leq t \leq \pi$	$\tan(t)$	1.4550e-013	1.4530e-013	2.00E-16
4	$-\pi \leq t \leq \pi$	$\sec(t)$	1.2434e-014	1.2323e-014	1.11E-16
5	$2 \leq t \leq 10$	$\log(t)$	1.7764e-015	2.2204e-015	-4.44E-16
6	$1 \leq t \leq 10$	$\sqrt{t+6} + (t+2)^2$	4.2633e-012	4.2917e-012	-2.84E-14

<https://doi.org/10.1371/journal.pone.0208015.t003>

functions are mentioned in column 2. Errors of these functions with CPS and QTS are shown in columns 4 and 5 respectively. Column 6 of Table 3 shows the difference of errors calculated by CPS and QTS. The difference, in column 6, between errors of both the splines is calculated to observe the accuracy of the splines.

8. Advantages of the proposed QTS

In this paper, a substantial method has been developed to construct a C^2 QTS and a brief comparison analysis is discussed. The advantages of proposed scheme are comprehended as follows:

- The proposed QTS scheme has decent characteristics of trigonometric splines.
- The method keeps the suitable geometric properties of splines.
- It carries out the C^2 smoothness.
- The proposed QTS produces an alternative to traditional CPS because of having four control points in its piecewise description.
- The comparison analysis of QTS and CPS, in Section 6, verifies the QTS as smother, more flexible, and more accurate alternate to CPS.
- The time analysis proves QTS computationally faster than CPS.

We can analyze the features of both QTS as well as CPS to see an overview of the two spline methods. It will highlight the differences between QTS and CPS. This is demonstrated in Table 4.

Table 4. Difference Table for QTS and CPS.

#	CPS	QTS
1.	CPS is algebraic piecewise polynomial.	QTS is trigonometric piecewise polynomial.
2.	CPS is Cubic.	QTS is Quadratic.
3.	CPS has four blending functions in an interval.	QTS has four blending functions in an interval.
4.	CPS owns the suitable geometric properties of splines.	QTS has the same geometric properties of splines as that of CPS.
5.	CPS has C^2 continuity.	QTS has C^2 continuity.
6.	The curve represented by CPS has reasonable bulge to represent flexibility.	The curve represented by QTS has more bulge than CPS.
7.	The area covered by the CPS is quite reasonable.	The area covered by the QTS is more than CPS.
8.	CPS is visually pleasing looking.	QTS is visually pleasing looking too and has more flexibility being able to have more bulge in the curve.
9.	CPS has reasonable computation error.	QTS observes less error of computation as compared to CPS and hence has more accuracy of the spline approximation.
10.	CPS elapses more time for execution.	The time elapsed by QTS is much smaller than CPS.
11.	CPS is economical in computation.	QTS is more economical in computation as compared to CPS.
12.	CPS is a well-known and recognized spline method used for Curve design.	QTS is a better alternative to CPS and hence expected to be more popular than CPS.

<https://doi.org/10.1371/journal.pone.0208015.t004>

9. Conclusion

A C^2 spline technique QTS is proposed and built with the eagerness of the object modeling using quadratic trigonometric functions. The curve model built through the proposed method owns the best suitable geometric properties such as partition of unity, CHP, AIP and VDP. The proposed scheme is more advantageous over the traditional CPS method. It is smoother, more flexible, faster and more accurate alternate to CPS. Furthermore, the built curve method is modest overall and is ideal for curve modeling. The authors, as future work, are also looking to expand the idea of the proposed QTS curve models for the designing of surface models.

Supporting information

S1 Table. Table of data of objects.
(PDF)

Author Contributions

Conceptualization: Shamailla Samreen, Muhammad Sarfraz.

Investigation: Muhammad Sarfraz.

Methodology: Shamailla Samreen.

Software: Shamailla Samreen.

Supervision: Muhammad Sarfraz, Malik Zawwar Hussain.

Visualization: Shamailla Samreen.

Writing – original draft: Shamailla Samreen.

Writing – review & editing: Muhammad Sarfraz.

References

1. Boehm W., Curvature continuous curves and surfaces, *Computer Aided Geometric Design*, 2(2) (1985) 313–323.
2. Barsky B., The Beta-spline: A local representation based on shape parameters and fundamental geometric measure, Ph.D. Thesis, University of Utah, 1981.
3. Cline A., Curve fitting in one and two dimensions using splines under tension, *Communications of the ACM*, 17 (1974) 218–223.
4. Dierckx P., Tytgat B., Generating the Bézier points of B-spline curve, *Computer Aided Geometric Design*, 6 (1989) 279–291.
5. Farin G. E., *Curves, surfaces for CAGD*, New York, Academic Press (1996).
6. Foley T. A., Local control of interval tension using weighted splines, *Computer Aided Geometric Design*, 3 (1987) 281–294.
7. Foley T. A., Interpolation with interval and point tension controls using cubic weighted u -splines, *ACM Transactions on Mathematical Software*, 13 (1987) 68–96.
8. Lewis J., B-spline bases for splines under tension, Nu splines, and fractional order splines, Presented at the SIAM-SIGNUM-Meeting, San Francisco, USA, (1975).
9. Nielson G M., Some piecewise polynomial alternatives to S-splines under tension, *Computer aided geometric design*, (1974) 209–235.
10. Nielson G. M., Rectangular u -splines, *IEEE Computer Graphics Application*, 6 (1986) 35–40.
11. Pruess S., Alternatives to the exponential spline in tension, *Mathematics of Computation*, 33 (1979) 1273–1281.
12. Sarfraz M., A C^2 rational cubic spline alternative to the NURBS, *Computers and Graphics* 16(1) (1992) 69–78.
13. Sarfraz M., curves, surfaces for CAD using C^2 rational cubic splines. *International Journal of Engineering with Computers*, 11(2) (1995) 94–102.
14. Sarfraz M., Freeform rational bicubic spline surfaces with tension control, *FACTA UNIVERSITATIS (NIS). Series of Mathematics and Informatics*, 9 (1994) 83–93.
15. Sarfraz M., Cubic spline curves with shape control, *International Journal of Computers and Graphics*, 18(5) (1994) 707–713.
16. Sarfraz M., Weighted Nu splines: an alternative to NURBS, advances in geometric modeling, Sarfraz M. (Ed), NewYork: Wiley; ISBN: 0-470-85937-7, (2003) 81–95.
17. Sarfraz M., Weighted Nu splines with local support basis, *Computers and Graphics*, 28 (2004) 539–549.
18. Sarfraz M., Designing objects with a spline, *International Journal of Computer Mathematics*, 87 (4) (2010) 797–817.
19. Hussain M. Z., Hussain M. and Waseem A., Shape-preserving trigonometric functions, *Computational and Applied Mathematics*, 33 (2) (2014) 411–431.
20. Sarfraz M., Samreen S., Hussain M. Z., (2016), Modeling of 2D objects with weighted-Quadratic Trigonometric Spline, 13th International Conference on Computer Graphics, Imaging and Visualization, Benni Mellal, Morocco, March 29-April 01, 2016, pp. 29–34, ISSN: 978-1-5090-0811-7, IEEE.
21. Sarfraz M., Samreen S., Hussain M. Z., (2017), A Quadratic Trigonometric Nu Spline with Shape Control, *International Journal of Image and Graphics*, Vol. 17(3), pp. 01–17, World Scientific.
22. Sarfraz M., Samreen S., Hussain M. Z., (2017), A Quadratic Trigonometric Weighted Spline with Local Support Basis Functions, *Alexandria Engineering Journal*, Elsevier, <https://doi.org/10.1016/j.aej.2017.02.016>.
23. Sarfraz M., Samreen S., Hussain M. Z., (2018), Designing by a Quadratic Trigonometric Spline with point and Interval Shape Control, The 2018 International Conference on Scientific Computing (CSC'18), Las Vegas, USA, Jul 30- Aug 2, 2018, IEEE (Accepted).
24. Samreen S., Sarfraz M., Jabeen N., Hussain M. Z., (2017), Computer Aided Design using a Rational Quadratic Trigonometric Spline with Interval Shape Control, International Conference on Computational Science and Computational Intelligence, Las Vegas, USA, Dec 14–16, 2017, pp. 246–251, ISSN: 978-1-5386-2652-8, IEEE.
25. Matlab, (2018), MathWorks, USA, <https://www.mathworks.com/>.