

RESEARCH

Open Access



Lupaş blending functions with shifted knots and q -Bézier curves

Kottakkaran Soopy Nisar^{1*}, Vinita Sharma² and Asif Khan²

*Correspondence:

n.soopy@psau.edu.sa

¹Department of Mathematics, College of Arts and Sciences, Prince Sattam bin Abdulaziz University, Wadi Aldawaser, 11991, Saudi Arabia

Full list of author information is available at the end of the article

Abstract

In this paper, we introduce blending functions of Lupaş q -Bernstein operators with shifted knots for constructing q -Bézier curves and surfaces. We study the nature of degree elevation and degree reduction for Lupaş q -Bézier Bernstein functions with shifted knots for $t \in [\frac{a}{[\mu]_q+b}, \frac{[\mu]_q+a}{[\mu]_q+b}]$. For the parameters $a = b = 0$, we get Lupaş q -Bézier curves defined on $[0, 1]$. We show that Lupaş q -Bernstein functions with shifted knots are tangent to fore-and-aft of its polygon at end points. We present a de Casteljau algorithm to compute Bernstein Bézier curves and surfaces with shifted knots. The new curves have some properties similar to q -Bézier curves. Similarly, we discuss the properties of the tensor product for Lupaş q -Bézier surfaces with shifted knots over the rectangular domain.

MSC: 65D17; 41A10; 41A25; 41A36

Keywords: q -integers; Degree elevation; De Casteljau-type algorithm; Lupaş q -Bernstein operators with shifted knots; Bézier curve; Tensor product; Shape preserving

1 Introduction

Approximation theory basically deals with approximation of functions by simpler functions. Broadly, it is divided into theoretical and constructive approximations. Recently, in the field of constructive approximation, Mursaleen et al. [27] introduced Lupaş q -Bernstein operators with shifted knots using q -calculus as follows.

Let $a, b \in \mathbb{N}_0$ (the set of all nonnegative integers), where $0 \leq a \leq b$. Then for $q \in (0, 1)$ and any $f \in C[0, 1]$, the Lupaş q -Bernstein operators with shifted knots are defined by

$$S_{\mu, q}^{(a, b)}(f; u) = \frac{1}{\binom{[\mu]_q}{[\mu]_q+b}_q} \sum_{s=0}^{\mu} \begin{bmatrix} \mu \\ s \end{bmatrix}_q \left(u - \frac{a}{[\mu]_q+b}\right)^s q^{\frac{(\mu-s)(\mu-s-1)}{2}} \left(\frac{[\mu]_q+a}{[\mu]_q+b} - u\right)^{\mu-s} f\left(\frac{[s]_q}{[\mu]_q}\right) \quad (1)$$

© The Author(s) 2020. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

or

$$S_{\mu,q}^{(a,b)}(f; u) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)^\mu} \sum_{j=0}^m \begin{bmatrix} \mu \\ s \end{bmatrix}_q \left(u - \frac{a}{[\mu]_q+b}\right)^s \left(\frac{[\mu]_q+a}{[\mu]_q+b} - u\right)^{\mu-s} f\left(\frac{[s]_q}{[\mu]_q}\right). \tag{2}$$

The other forms of these operators are as follows:

$$S_{\mu,q}^{(a,b)}(f; u) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)^\mu} \times \sum_{s=0}^{\mu} \begin{bmatrix} \mu \\ s \end{bmatrix}_q \left(\frac{[\mu]_q+a}{[\mu]_q+b} - u\right)^{\mu-s} q^{\frac{s(s-1)}{2}} \left(u - \frac{a}{[\mu]_q+b}\right)^s f\left(\frac{[s]_q}{[\mu]_q}\right) \tag{3}$$

or

$$S_{\mu,q}^{(a,b)}(f; u) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)^\mu} \times \sum_{s=0}^{\mu} \begin{bmatrix} \mu \\ s \end{bmatrix}_q \left(\frac{[\mu]_q+a}{[\mu]_q+b} - u\right)^{\mu-s} \left(u - \frac{a}{[\mu]_q+b}\right)^s f\left(\frac{[s]_q}{[\mu]_q}\right). \tag{4}$$

We can easily verify that all four forms are equivalent. Here $\frac{a}{[\mu]_q+b} \leq u \leq \frac{[\mu]_q+a}{[\mu]_q+b}$ for $0 \leq a \leq b$. In the case $a = b = 0$ the above operators reduce to the Lupaş q -Bernstein operators [23]. Further for $a = b = 0$ and $q = 1$, they reduce to the classical Bernstein operators [4].

Computer aided geometric design is a discipline that deals with study of computational aspects of geometric objects. Bases of Bernstein operators and its generalizations are used in computer aided geometric design to construct curves and surfaces. For more concepts and techniques used in CAGD, we refer to [2, 3, 9–12, 14, 15, 17, 39]. The most popular Bézier curves are constructed with the help of Bernstein bases [5].

Recently, Khalid et al. [20] studied Bézier curves and surfaces constructed with modified Bernstein bases of classical Bernstein operators with shifted knots.

We refer to [4, 8, 20, 21, 24, 25, 28–30, 33, 36–38] for details related to quantum calculus and approximation theory and to [1, 6, 7, 13, 16, 18, 19, 22, 23, 25, 31, 32, 34, 35] for computer aided geometric design.

Motivated by [20, 27], we study various properties of Lupaş q -Bernstein basis functions or blending functions with shifted knots [27]. Popular programs, like Adobe’s illustrator and flash, and font imaging systems such as postscript utilize Bernstein polynomials to form Bézier curves. The novelty of this paper is that we can generate blending functions on $[0, 1]$ and its subintervals and the parameters $q, a,$ and b provide flexibility in construction of blending functions and Bézier curves and surfaces. The algorithms and other derived results using blending functions with shifted knots will be very useful in implementation using computers for simulation purposes.

Let us recall some basic definitions and notations of quantum calculus [16]. For any fixed real number $q > 0$, the q -integer $[s]_q$ for $s \in \mathbb{N}$ and q -factorial $[s]_q!$ are defined as

$$[s]_q := \begin{cases} \frac{(1-q^s)}{(1-q)}, & q \neq 1, \\ s, & q = 1, \end{cases}$$

and the q -factorial $[s]_q!$ by

$$[s]_q! := \begin{cases} [s]_q[s-1]_q \cdots [1]_q, & s \geq 1, \\ 1, & s = 0. \end{cases}$$

The q -analogue of binomial expansion is

$$(u + v)_q^\mu := (u + v)(u + qv)(u + q^2v) \cdots (u + q^{\mu-1}v).$$

From the above we have

$$(u)_q^\mu = u^\mu. \tag{5}$$

Also, from the q -analogue of binomial expansion we have

$$(v)_q^\mu := (v)(qv)(q^2v) \cdots (q^{\mu-1}v) = q^{\frac{\mu(\mu-1)}{2}} v^\mu. \tag{6}$$

From q -binomial expansion we can also derive:

$$\begin{aligned} (u - \alpha)_q^\mu &= (u - \alpha)_q^{\mu-1}(u - q^{\mu-1}\alpha), \\ (u - \alpha)_q^\mu &= (u - \alpha)_q^{\mu-1}(u - q\alpha)^{\mu-1}. \end{aligned}$$

In fact,

$$\begin{aligned} (u - \alpha)_q^{v+\mu} &\neq (u - \alpha)_q^v(u - \alpha)_q^\mu, \\ (u - \alpha)_q^{v+\mu} &= (u - \alpha)_q^v(u - q^v\alpha)_q^\mu, \end{aligned}$$

and

$$\begin{aligned} (\alpha - u)_q^\mu &\neq (-1)^\mu(u - \alpha)_q^\mu, \\ (\alpha - u)_q^\mu &= (-1)^\mu q^{\frac{\mu(\mu-1)}{2}}(u - q^{-\mu+1}\alpha)_q^\mu. \end{aligned}$$

When $(-v)$ is a negative integer, then

$$\begin{aligned} (u - \alpha)_q^{-v} &\neq \frac{1}{(u - \alpha)_q^v}, \\ (u - \alpha)_q^{-v} &= \frac{1}{(u - q^{-v}\alpha)_q^v}. \end{aligned}$$

The q -analogues of binomial coefficients are defined by

$$\begin{aligned} \begin{bmatrix} \mu \\ s \end{bmatrix}_q &:= \frac{[\mu]_q!}{[s]_q![\mu - s]_q!}, \\ \begin{bmatrix} \mu \\ s \end{bmatrix}_q &= \begin{bmatrix} \mu - 1 \\ s - 1 \end{bmatrix}_q + q^s \begin{bmatrix} \mu - 1 \\ s \end{bmatrix}_q, \end{aligned}$$

$$\begin{aligned} \begin{bmatrix} \mu \\ s \end{bmatrix}_q &= q^{\mu-s} \begin{bmatrix} \mu-1 \\ s-1 \end{bmatrix}_q + \begin{bmatrix} \mu-1 \\ s \end{bmatrix}_q, \\ \begin{bmatrix} \mu \\ s \end{bmatrix}_q [\mu-s]_q &= \begin{bmatrix} \mu \\ s+1 \end{bmatrix}_q [s+1]_q, \\ \begin{bmatrix} \mu \\ s \end{bmatrix}_q [\mu+1]_q &= \begin{bmatrix} \mu+1 \\ s+1 \end{bmatrix}_q [s+1]_q. \end{aligned}$$

A further extension of q -calculus is (p, q) -calculus. For details about (p, q) -calculus and its applications in approximation theory, we refer to [18, 25, 26].

2 Lupaş q -Bernstein functions with shifted knots

The Lupaş basis (blending) functions with shifted knots obtained from (2) are as follows:

$$B_{\mu,q}^{s,a,b}(t) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)_q^\mu} \begin{bmatrix} \mu \\ s \end{bmatrix}_q \left(t - \frac{a}{[\mu]_q+b}\right)^s \left(\frac{[\mu]_q+a}{[\mu]_q+b} - t\right)_q^{\mu-s}. \tag{7}$$

2.1 Characteristics of the Lupaş q -Bernstein functions with shifted knots

Theorem 2.1 *The Lupaş q -Bernstein functions with shifted knots have the following properties:*

1. *Nonnegativity:* $B_{\mu,q}^{s,a,b}(t) \geq 0, s = 0, 1, \dots, \mu, t \in \left[\frac{a}{[\mu]_q+b}, \frac{[\mu]_q+a}{[\mu]_q+b}\right]$.
2. *Partition of unity:* $\sum_{s=0}^\mu B_{\mu,q}^{s,a,b}(t) = 1$ for every $t \in \left[\frac{a}{[\mu]_q+b}, \frac{[\mu]_q+a}{[\mu]_q+b}\right]$.
3. *End-point interpolation property:*

$$B_{\mu,q}^{s,a,b}\left(\frac{a}{[\mu]_q+b}\right) = \begin{cases} 1 & \text{if } s = 0, \\ 0, & s \neq 0, \end{cases} \tag{8}$$

$$B_{\mu,q}^{s,a,b}\left(\frac{[\mu]_q+a}{[\mu]_q+b}\right) = \begin{cases} 1 & \text{if } s = \mu, \\ 0, & s \neq \mu. \end{cases} \tag{9}$$

Clearly, both sided end-point interpolation properties hold.

4. *Reducibility:* when $a = b = 0$ and $q = 1$, formula (7) reduces to the classical Bernstein bases on $[0, 1]$.

When $a = b = 0$, it reduces to Lupaş q -Bernstein bases (rational function).

When $q = 1$, it reduces to the shifted Bernstein function given by Khalid et al.

[20].

Proof Properties (1), (2), and (4) can be easily obtained from equation (7). Here we give a proof of property (3) only.

Property 3: From equation (7) we have

$$B_{\mu,q}^{s,a,b}(t) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)_q^\mu} \begin{bmatrix} \mu \\ s \end{bmatrix}_q \left(t - \frac{a}{[\mu]_q+b}\right)^s \left(\frac{[\mu]_q+a}{[\mu]_q+b} - t\right)_q^{\mu-s}.$$

(i) When $s = 0$,

$$B_{\mu,q}^{s,a,b}(t) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)^\mu} \begin{bmatrix} \mu \\ 0 \end{bmatrix}_q \left(\frac{[\mu]_q+a}{[\mu]_q+b} - t\right)_q^\mu,$$

$$B_{\mu,q}^{s,a,b}\left(\frac{a}{[\mu]_q+b}\right) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)^\mu} \begin{bmatrix} \mu \\ 0 \end{bmatrix}_q \left(\frac{[\mu]_q}{[\mu]_q+b}\right)_q^\mu.$$

Hence

$$B_{\mu,q}^{s,a,b}\left(\frac{a}{[\mu]_q+b}\right) = 1. \tag{10}$$

(ii) When $s \neq 0$,

$$B_{\mu,q}^{s,a,b}\left(\frac{a}{[\mu]_q+b}\right) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)^\mu} \begin{bmatrix} \mu \\ s \end{bmatrix}_q \left(\frac{a}{[\mu]_q+b} - \frac{a}{[\mu]_q+b}\right)^s \left(\frac{[\mu]_q+a}{[\mu]_q+b} - t\right)_q^\mu.$$

Thus

$$B_{\mu,q}^{s,a,b}\left(\frac{a}{[\mu]_q+b}\right) = 0. \tag{11}$$

(iii) When $s = \mu$,

$$B_{\mu,q}^{s,a,b}(t) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)^\mu} \begin{bmatrix} \mu \\ \mu \end{bmatrix}_q \left(t - \frac{a}{[\mu]_q+b}\right)_q^\mu,$$

$$B_{\mu,q}^{s,a,b}\left(\frac{[\mu]_q+a}{[\mu]_q+b}\right) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)^\mu} \left(\frac{[\mu]_q}{[\mu]_q+b}\right)_q^\mu,$$

and from equation (5) we get

$$B_{\mu,q}^{s,a,b}\left(\frac{[\mu]_q+a}{[\mu]_q+b}\right) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)^\mu} \left(\frac{[\mu]_q}{[\mu]_q+b}\right)_q^\mu = 1.$$

Similarly,

(iv) when $s \neq \mu$, then

$$B_{\mu,q}^{s,a,b}\left(\frac{[\mu]_q+a}{[\mu]_q+b}\right) = 0.$$

□

3 Degree elevation and reduction for Lupaş q -Bernstein functions with shifted knots

This algorithm has been used to change the bases of Bézier curves. We can elevate the degree of curve to obtain more local control in designing the curve. With the help of this

algorithm, we can construct a new control polygon by taking a convex combination of the old control points that retains the previous points. For this application, identities (12) and (13) and Theorem 3.1 are useful.

3.1 Identities

$$\begin{aligned} & \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) B_{\mu,q}^{s,a,b}(t) \\ &= \frac{[\mu + 1 - s]_q}{[\mu + 1]_q} \left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right] B_{\mu+1,q}^{s,a,b}(t) \end{aligned} \tag{12}$$

and

$$\begin{aligned} & q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) B_{\mu,q}^{s,a,b}(t) \\ &= q^{\mu-s} \left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right] \left(\frac{[s + 1]_q}{[\mu + 1]_q} \right) B_{\mu+1,q}^{s+1,a,b}(t). \end{aligned} \tag{13}$$

Proof Consider

$$B_{\mu,q}^{s,a,b}(t) = \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q + b} \right)_q^\mu} \begin{bmatrix} \mu \\ s \end{bmatrix}_q q^{\frac{s(s-1)}{2}} \left(t - \frac{a}{[\mu]_q + b} \right)^s \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)^{\mu-s}.$$

Similarly, from (3) we can also obtain its other forms:

$$\begin{aligned} & \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)_q B_{\mu,q}^{s,a,b}(t) \\ &= \frac{1}{\left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + \left(t - \frac{a}{[\mu]_q + b} \right) \right]_q^\mu} \\ & \quad \times \left\{ \begin{bmatrix} \mu \\ s \end{bmatrix}_q q^{\frac{s(s-1)}{2}} \left(t - \frac{a}{[\mu]_q + b} \right)^s \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)^{\mu+1-s} \right\}, \\ & \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)_q B_{\mu,q}^{s,a,b}(t) \\ &= \frac{\begin{bmatrix} \mu \\ s \end{bmatrix}_q}{\begin{bmatrix} \mu+1 \\ s \end{bmatrix}_q} \left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right] \\ & \quad \times \left\{ \frac{1}{\left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + \left(t - \frac{a}{[\mu]_q + b} \right) \right]_q^{\mu+1}} \begin{bmatrix} \mu+1 \\ s \end{bmatrix}_q \right. \\ & \quad \left. \times \left(t - \frac{a}{[\mu]_q + b} \right)^s \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)^{\mu-s+1} \right\} \\ &= \left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right] \frac{\begin{bmatrix} \mu \\ s \end{bmatrix}_q}{\begin{bmatrix} \mu+1 \\ s \end{bmatrix}_q} B_{\mu+1,q}^{s,a,b}(t), \end{aligned}$$

$$\begin{aligned} & \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)_q B_{\mu,q}^{s,a,b}(t) \\ &= \left(\frac{[\mu + 1 - j]_q}{[\mu + 1]_q} \right) \left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right] B_{\mu+1,q}^{s,a,b}(t). \end{aligned}$$

Similarly, from

$$q^\mu \left(t - \frac{a}{[\mu]_q + b} \right)_q B_{\mu,q}^{s,a,b}(t) = \left(\frac{[\mu]_q}{[\mu]_q + b} \right)_q \left(\frac{[s + 1]_q}{[\mu + 1]_q} \right) B_{\mu+1,q}^{s+1,a,b}(t)$$

we have

$$\begin{aligned} & q^\mu \left(t - \frac{a}{[\mu]_q + b} \right)_q B_{\mu,q}^{s,a,b}(t) \\ &= \frac{q^\mu}{\left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + \left(t - \frac{a}{[\mu]_q + b} \right) \right]_q^\mu} \\ & \quad \times \left\{ \begin{bmatrix} \mu \\ s \end{bmatrix}_q q^{\frac{s(s-1)}{2}} \left(t - \frac{a}{[\mu]_q + b} \right)^{s+1} \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)^{\mu-s} \right\} \\ &= \frac{\begin{bmatrix} \mu \\ s \end{bmatrix}_q}{\begin{bmatrix} \mu+1 \\ s+1 \end{bmatrix}_q} \left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right] \\ & \quad \times \left\{ \frac{q^\mu}{\left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + \left(t - \frac{a}{[\mu]_q + b} \right) \right]_q^{\mu+1}} \begin{bmatrix} \mu + 1 \\ s + 1 \end{bmatrix}_q \right. \\ & \quad \times \left. q^{\frac{s(s-1)}{2}} \left(t - \frac{a}{[\mu]_q + b} \right)^{s+1} \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)^{\mu-s} \right\} \\ &= \frac{\begin{bmatrix} \mu \\ s \end{bmatrix}_q}{\begin{bmatrix} \mu+1 \\ s+1 \end{bmatrix}_q} q^{\mu-s} \left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right] B_{\mu+1,q}^{s+1,a,b}(t) \\ &= \left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right] \frac{[s + 1]_q}{[\mu + 1]_q} B_{\mu+1,q}^{s+1,a,b}(t). \quad \square \end{aligned}$$

Theorem 3.1 *Each Lupaş q-Bernstein function with shifted knots of degree μ is a linear combination of two Lupaş q-Bernstein functions with shifted knots of degree $\mu + 1$:*

$$B_{\mu,q}^{s,a,b}(t) = \left(\frac{[\mu + 1 - j]_q}{[\mu + 1]_q} \right) B_{\mu+1,q}^{s,a,b}(t) + q^{\mu-s} \left(\frac{[s + 1]_q}{[\mu + 1]_q} \right) B_{\mu+1,q}^{s+1,a,b}(t), \tag{14}$$

where $\frac{a}{[\mu]_q + b} \leq t \leq \frac{[\mu]_q + a}{[\mu]_q + b}$ for nonnegative integers a, b satisfying $0 \leq a \leq b$.

Proof We obtain this result by adding identities (12) and (13). □

Theorem 3.2 Each Lupaş q -Bernstein function with shifted knots of degree μ is a linear combination of two Lupaş q -Bernstein functions with shifted knots of degree $\mu - 1$:

$$B_{\mu,q}^{s,a,b}(t) = \frac{1}{\left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right]} \times \left[q^{s-1} \left(t - \frac{a}{[\mu]_q + b} \right) B_{\mu-1,q}^{s-1,a,b}(t) + q^s \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) B_{\mu-1,q}^{s,a,b}(t) \right], \tag{15}$$

where $\frac{a}{[\mu]_q + b} \leq t \leq \frac{[\mu]_q + a}{[\mu]_q + b}$ for nonnegative integers a, b satisfying $0 \leq a \leq b$.

Proof Using a Pascal-type relation, we have

$$\begin{aligned} B_{\mu,q}^{s,a,b}(t) &= \begin{bmatrix} \mu \\ s \end{bmatrix}_q \frac{1}{\left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right]} \\ &\quad \times q^{\frac{s(s-1)}{2}} \left(t - \frac{a}{[\mu]_q + b} \right)^s \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)^{\mu-s} \\ &= \left(\begin{bmatrix} \mu - 1 \\ s - 1 \end{bmatrix}_q + q^s \begin{bmatrix} \mu - 1 \\ s \end{bmatrix}_q \right) \frac{1}{\left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right]} \\ &\quad \times \left(t - \frac{a}{[\mu]_q + b} \right)^s \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)^{\mu-s}_q \\ &= \frac{1}{\left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right]} \\ &\quad \times \left[q^{s-1} \left(t - \frac{a}{[\mu]_q + b} \right) B_{\mu-1,q}^{s-1,a,b}(t) + q^s \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) B_{\mu-1,q}^{s,a,b}(t) \right]. \quad \square \end{aligned}$$

4 Lupaş q -Bernstein Bézier curves with shifted knots

The Lupaş q -Bernstein Bézier curves with shifted knots of degree μ can be represented in the form of linear combination of control points and Lupaş q -Bernstein functions with shifted knots:

$$P(t) = \sum_{s=0}^{\mu} P_s B_{\mu,q}^{s,a,b}(t), \tag{16}$$

where P_s are the control points. After joining these points, we get a polygon called a control polygon. Now after defining the properties of Lupaş q -Bernstein functions with shifted knots, we examine the properties of the above curves.

Theorem 4.1 Property of derivative at the end points:

$$P' \left(\frac{a}{[\mu]_q + b} \right) = ([\mu]_q + b)(P_1 - P_0), \tag{17}$$

$$P' \left(\frac{[\mu]_q + a}{[\mu]_q + b} \right) = ([\mu]_q + b)(P_\mu - P_{\mu-1}), \tag{18}$$

that is, Lupaş q -Bernstein curves with shifted knots are tangential at the end points of its control polygon.

Proof

$$\begin{aligned} \mathbf{P}(t) &= \sum_{s=0}^{\mu} \mathbf{P}_s B_{\mu,q}^{s,a,b}(t) \\ &= \sum_{s=0}^{\mu} \mathbf{P}_s \begin{bmatrix} \mu \\ s \end{bmatrix}_q \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)_q^{\mu}} \left(t - \frac{a}{[\mu]_q+b}\right)_q^s \left(\frac{[\mu]_q+a}{[\mu]_q+b} - t\right)^{\mu-s}. \end{aligned}$$

Let

$$\mathbf{P}(t) = \mathbf{V}(t).$$

Taking the derivatives of both sides with respect to t , we have

$$\mathbf{P}'(t) = \mathbf{V}'(t).$$

Let

$$A_s^{\mu}(t) = \begin{bmatrix} \mu \\ s \end{bmatrix}_q \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)_q^{\mu}} \left(t - \frac{a}{[\mu]_q+b}\right)_q^s \left(\frac{[\mu]_q+a}{[\mu]_q+b} - t\right)^{\mu-s}.$$

Then

$$\begin{aligned} \mathbf{V}(t) &= \sum_{s=0}^{\mu} \mathbf{P}_s A_s^{\mu}(t), \\ (A_s^{\mu}(t))' &= \begin{bmatrix} \mu \\ s \end{bmatrix}_q \frac{1}{\left(\frac{[\mu]_q}{[\mu]_q+b}\right)_q^{\mu}} \left\{ s \left(t - \frac{a}{[\mu]_q+b}\right)_q^{s-1} \left(\frac{[\mu]_q+a}{[\mu]_q+b} - t\right)^{\mu-s} \right. \\ &\quad \left. - (\mu - s) \left(t - \frac{a}{[\mu]_q+b}\right)_q^s \left(\frac{[\mu]_q+a}{[\mu]_q+b} - t\right)^{\mu-s-1} \right\}. \end{aligned}$$

After some calculation, we get

$$\mathbf{V}'\left(\frac{a}{[\mu]_q+b}\right) = \mathbf{P}'\left(\frac{a}{[\mu]_q+b}\right) = ([\mu]_q+b)(\mathbf{P}_1 - \mathbf{P}_0).$$

Similarly, we have

$$\mathbf{V}'\left(\frac{[\mu]_q+a}{[\mu]_q+b}\right) = \mathbf{P}'\left(\frac{[\mu]_q+a}{[\mu]_q+b}\right) = ([\mu]_q+b)(\mathbf{P}_{\mu} - \mathbf{P}_{\mu-1}),$$

which completes the proof. □

4.1 Degree elevation for Lupaş q -Bernstein Bézier curves with shifted knots

Lupaş q - Bernstein Bézier curves with shifted knots have a degree elevation formula, which is the same as that for the classical Bézier curves. With the help of this technique, we can attain more control over the shape of a given curve:

$$\mathbf{P}(t) = \sum_{s=0}^{\mu} \mathbf{P}_s \mathcal{B}_{\mu,q}^{s,a,b}(t) \tag{19}$$

after using degree elevation

$$\mathbf{P}(t) = \sum_{s=0}^{\mu+1} \mathbf{P}_s^* \mathcal{B}_{\mu+1,q}^{s,a,b}(t), \tag{20}$$

where

$$\mathbf{P}_s^* = \left(\frac{[\mu + 1 - s]_q}{[\mu + 1]_q} \right) \mathbf{P}_s + q^{\mu+1-s} \left(\frac{[s]_q}{[\mu + 1]_q} \right) \mathbf{P}_{s-1}. \tag{21}$$

This statement can be obtained from Theorem 3.1. If we put $a = b = 0$ and $q = 1$, then formula (21) changes to the Bézier curves degree elevation formula. Denoting by $P = (P_0, P_1, \dots, P_{\mu})^T$ the vector of control points of the initial Bézier curve of degree μ and by $\mathbf{P}^{(1)} = (P_0^*, P_1^*, \dots, P_{\mu+1}^*)$ the vector of control points of the degree elevated Bézier curve of degree $\mu + 1$, we can define the degree elevation procedure as

$$\mathbf{P}^{(1)} = T_{\mu+1} \mathbf{P},$$

where $T_{\mu+1}$ is given by

$$M_{hs} = \begin{cases} \frac{[\mu+1-s]_q}{[\mu+1]_q} & \text{when } h = s, \\ q^{\mu+1-s} \frac{[s]_q}{[\mu+1]_q} & \text{when } h = s + 1, \\ 0, & \text{when } h \neq s, s + 1. \end{cases}$$

After degree elevation, the vector of new control points of Bézier curves of degree $\mu + l$ is $\mathbf{P}^{(l)} = T_{\mu+l} T_{\mu+2} \dots T_{\mu+1} \mathbf{P}$ for $l \in \mathbb{N}$.

As $l \rightarrow \infty$, the control polygon $\mathbf{P}^{(l)}$ converges to the Bézier curve.

In next section, we study a de Casteljau-type algorithm. The de Casteljau algorithm is an elementary technique of shape designs. This algorithm can be used to split a single curve into two curves at an arbitrary parameter value.

4.2 De Casteljau algorithm

Bézier curves with shifted knots of degree μ can be represented in the form of a linear combination of two Bézier curves with shifted knots of degree $\mu - 1$, and we can obtain two algorithms to assess Bézier curves with shifted knots.

Algorithm 1

$$\begin{cases} \mathbf{P}_i^0(t) \equiv \mathbf{P}_i^0 \equiv \mathbf{P}_i, & i = 0, 1, 2, \dots, \mu, \\ \mathbf{P}_i^r(t) = \frac{1}{\left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^r \left(t - \frac{a}{[\mu]_q + b} \right) \right]} \\ \quad \times \left[q^{r-1} \left(t - \frac{a}{[\mu]_q + b} \right) \mathbf{P}_{i+1}^{r-1}(t) + q^r \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) \mathbf{P}_i^{r-1}(t) \right], \\ r = 1, \dots, \mu, \quad i = 0, 1, \dots, \mu - r, \quad \frac{a}{[\mu]_q + b} \leq t \leq \frac{[\mu]_q + a}{[\mu]_q + b}, \quad 0 \leq a \leq b. \end{cases} \tag{22}$$

Then

$$\mathbf{P}(t) = \sum_{i=0}^{\mu-1} \mathbf{P}_i^1(t) = \dots = \sum \mathbf{P}_i^r(t) b_q^{i, \mu-r}(t) = \dots = \mathbf{P}_0^\mu(t). \tag{23}$$

It is clear that the results can be obtained from Theorem 3.2. Let $P^0 = (P_0, P_1, \dots, P_\mu)^T$ and $P^r = (P_0^r, P_1^r, \dots, P_{\mu-r}^r)^T$. Then the algorithm of de Casteljau type can be expressed as follows.

Algorithm 2

$$\mathbf{P}^r(t) = M_r(t) \cdots M_2(t) M_1(t) \mathbf{P}^0, \tag{24}$$

where $M_r(t)$ is a $(\mu - r + 1) \times (\mu - r + 2)$ matrix:

$$M_r(t) = \frac{1}{\left[\left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right) + q^\mu \left(t - \frac{a}{[\mu]_q + b} \right) \right]} [D]$$

with

$$D = \begin{bmatrix} q^r \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)_q & q^{r-1} \left(t - \frac{a}{[\mu]_q + b} \right)_q & \dots & 0 & 0 \\ 0 & q^r \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)_q & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & \dots & q^{r-1} \left(t - \frac{a}{[\mu]_q + b} \right)_q & 0 \\ 0 & 0 & \dots & q^r \left(\frac{[\mu]_q + a}{[\mu]_q + b} - t \right)_q & q^{r-1} \left(t - \frac{a}{[\mu]_q + b} \right)_q \end{bmatrix}_{(\mu+2) \times (\mu+1)}.$$

5 Tensor product of Lupaş q -Bernstein Bézier surfaces with shifted knots on

$$\left[\frac{a}{[\mu]_q + b}, \frac{[\mu]_q + a}{[\mu]_q + b} \right] \times \left[\frac{a}{[\mu]_q + b}, \frac{[\mu]_q + a}{[\mu]_q + b} \right]$$

We define a two-parameter family $\mathbf{P}(u, v)$ of tensor product surfaces of degree $v \times \mu$ as follows:

$$\begin{aligned} \mathbf{P}(u, v) &= \sum_{i=0}^v \sum_{j=0}^\mu \mathbf{P}_{ij} B_{v,q}^{i,a,b}(u) B_{\mu,q}^{j,a,b}(v), \\ (u, v) &\in \left[\frac{a}{[\mu]_q + b}, \frac{[\mu]_q + a}{[\mu]_q + b} \right] \times \left[\frac{a}{[\mu]_q + b}, \frac{[\mu]_q + a}{[\mu]_q + b} \right], \end{aligned} \tag{25}$$

where $\mathbf{P}_{ij} \in \mathbb{R}^3$ ($i = 0, 1, \dots, v, s = 0, 1, \dots, \mu$), and $B_{v,q}^{i,a,b}(u)$ and $B_{\mu,q}^{j,a,b}(v)$ are Lupaş and Bernstein functions, respectively. Here \mathbf{P}_{ij} denotes the control points. By joining adjacent points of same rows/columns we can get a control net of the tensor product Bézier surface.

5.1 Properties

- Affine invariance property: Since

$$\sum_{i=0}^v \sum_{j=0}^{\mu} B_{v,q}^{i,a,b}(u) B_{\mu,q}^{j,a,b}(v) = 1, \tag{26}$$

$\mathbf{P}(u, v)$ denotes is combination of its control points.

- Convex hull property: Convex combination of \mathbf{P}_{ij} is denoted by $\mathbf{P}(u, v)$ and lies in the convex hull of its control net.
- Isoparametric property for curves: The isoparametric curves $v = v^*$ and $u = u^*$ of a tensor product Bézier surface are respectively the Lupaş Bézier curves with shifted knots of degrees v and μ , namely,

$$\mathbf{P}(u, v^*) = \sum_{i=0}^v \left(\sum_{j=0}^{\mu} \mathbf{P}_{ij} B_{\mu,q}^{s,a,b}(v^*) \right) B_{v,q}^{i,a,b}(u), \quad u \in \left[\frac{a}{[\mu]_q + b}, \frac{[\mu]_q + a}{[\mu]_q + b} \right];$$

$$\mathbf{P}(u^*, v) = \sum_{j=0}^{\mu} \left(\sum_{i=0}^v \mathbf{P}_{i,s} B_{\mu,q}^{s,a,b}(u^*) \right) B_{v,q}^{i,a,b}(v), \quad v \in \left[\frac{a}{[\mu]_q + b}, \frac{[\mu]_q + a}{[\mu]_q + b} \right].$$

The boundaries of the curves of $\mathbf{P}(u, v)$ are evaluated by $\mathbf{P}(u, \frac{a}{[\mu]_q + b})$, $\mathbf{P}(u, \frac{[\mu]_q + a}{[\mu]_q + b})$, $\mathbf{P}(\frac{a}{[\mu]_q + b}, v)$, and $\mathbf{P}(\frac{[\mu]_q + a}{[\mu]_q + b}, v)$.

- Interpolation property at corner points: The corner control net coincides with the four corners of the surface:

$$\mathbf{P}\left(\frac{a}{[\mu]_q + b}, \frac{a}{[\mu]_q + b}\right) = \mathbf{P}_{0,0}, \quad \mathbf{P}\left(\frac{a}{[\mu]_q + b}, \frac{[\mu]_q + a}{[\mu]_q + b}\right) = \mathbf{P}_{0,\mu},$$

$$\mathbf{P}\left(\frac{[v]_q + a}{[v]_q + b}, \frac{a}{[\mu]_q + b}\right) = \mathbf{P}_{v,0}, \quad \mathbf{P}\left(\frac{[v]_q + a}{[v]_q + b}, \frac{[\mu]_q + a}{[\mu]_q + b}\right) = \mathbf{P}_{v,\mu}.$$

- Reducibility: When $a = b = 0$ and $q = 1$, formula (25) reduces to the classical tensor product Bézier patch.

5.2 Degree elevation and de Casteljau algorithm

A tensor product Lupaş q -Bernstein surface with shifted knots of degree $v \times \mu$ is $\mathbf{P}(u, v)$. As an example, for getting the same surface as a surface of degree $(v + 1) \times (\mu + 1)$, we need to find new control points \mathbf{P}_{ij}^* such that

$$\mathbf{P}(u, v) = \sum_{i=0}^v \sum_{j=0}^{\mu} \mathbf{P}_{ij} B_{v,q}^{i,a,b}(u) B_{\mu,q}^{j,a,b}(v) = \sum_{i=0}^{v+1} \sum_{j=0}^{\mu+1} \mathbf{P}_{ij}^* B_{v+1,q}^{i,a,b}(u) B_{\mu+1,q}^{s,a,b}(v). \tag{27}$$

Let $a_i = 1 - \frac{[v+1-i]_q}{[v+1]_q}$, $b_j = 1 - \frac{[\mu+1-j]_q}{[\mu+1]_q}$. Then

$$\mathbf{P}_{ij}^* = a_i b_j \mathbf{P}_{i-1,j-1} + a_i (1 - b_j) \mathbf{P}_{i-1,j} + (1 - a_i) (1 - b_j) \mathbf{P}_{i,j}, \tag{28}$$

which can be written in matrix form as

$$\begin{bmatrix} 1 - \frac{[v+1-i]_q}{[v+1]_q} & \frac{[v+1-i]_q}{[v+1]_q} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{i-1,j-1} & \mathbf{P}_{i-1,j} \\ \mathbf{P}_{i,j-1} & \mathbf{P}_{i,j} \end{bmatrix} \begin{bmatrix} 1 - \frac{[\mu+1-s]_q}{[\mu+1]_q} \\ \frac{[\mu+1-s]_q}{[\mu+1]_q} \end{bmatrix}.$$

Similarly, the de Casteljau algorithms can be extended to evaluate points on a Bézier surface. Given the control points $\mathbf{P}_{ij} \in \mathbb{R}^3, i = 0, 1, \dots, \nu, s = 0, 1, \dots, \mu,$

$$\left\{ \begin{array}{l} \mathbf{P}_{ij}^{0,0}(u, \nu) \equiv \mathbf{P}_{ij}^{0,0} \equiv \mathbf{P}_{ij} \quad i = 0, 1, \dots, \nu; j = 0, 1, \dots, \mu, \\ \mathbf{P}_{ij}^{r,r}(u, \nu) = [E] \begin{bmatrix} \mathbf{P}_{ij}^{r-1,r-1} & \mathbf{P}_{i+1,j}^{r-1,r-1} \\ \mathbf{P}_{i+1,j}^{r-1,r-1} & \mathbf{P}_{i+1,j+1}^{r-1,r-1} \end{bmatrix} [F], \\ \text{where} \\ E = \left[\frac{1}{\left(\frac{[v]_q+a}{[v]_q+b} - u \right) + q^\nu \left(u - \frac{a}{[v]_q+b} \right)} \right] \left(\frac{[v]_q+a}{[v]_q+b} - u \right) \frac{1}{\left(\frac{[v]_q+a}{[v]_q+b} - u \right) + q^\nu \left(u - \frac{a}{[v]_q+b} \right)} \left(u - \frac{a}{[v]_q+b} \right) \right] \\ \text{and} \\ F = \left[\frac{1}{\left(\frac{[\mu]_q+a}{[\mu]_q+b} - \nu \right) + q^\mu \left(\nu - \frac{a}{[\mu]_q+b} \right)} \right] \left(\frac{[\mu]_q+a}{[\mu]_q+b} - \nu \right) \frac{1}{\left(\frac{[\mu]_q+a}{[\mu]_q+b} - \nu \right) + q^\mu \left(\nu - \frac{a}{[\mu]_q+b} \right)} \left(\nu - \frac{a}{[\mu]_q+b} \right) \right], \\ r = 1, \dots, s, \quad s = \min(\nu, \mu), \\ i = 0, 1, 2, \dots, \nu - r; \quad s = 0, 1, 2, \dots, \mu - r. \end{array} \right. \tag{29}$$

When $\nu = \mu,$ to get a point on the surface, we can directly use the above algorithms. When $\nu \neq \mu,$ to get a point on the surface after s applications of formula (29), we perform formula (24) for the intermediate point $\mathbf{P}_{ij}^{s,s}.$

Note that we get Lupaş q -Bézier curves and surfaces for $(u, \nu) \in \left[\frac{a}{[\mu]_q+b}, \frac{[\mu]_q+a}{[\mu]_q+b} \right] \times \left[\frac{a}{[\mu]_q+b}, \frac{[\mu]_q+a}{[\mu]_q+b} \right]$ when we set the parameters $a = b = 0.$

Further, we have classical Bézier curves and surfaces for $(u, \nu) \in \left[\frac{a}{[\mu]_q+b}, \frac{[\mu]_q+a}{[\mu]_q+b} \right] \times \left[\frac{a}{[\mu]_q+b}, \frac{[\mu]_q+a}{[\mu]_q+b} \right]$ when we set the parameters $a = b = 0$ and $q = 1.$

Acknowledgements

None.

Funding

Not applicable.

Availability of data and materials

None.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

All authors jointly worked on the results, and they all read and approved the final manuscript.

Author details

¹Department of Mathematics, College of Arts and Sciences, Prince Sattam bin Abdulaziz University, Wadi Aldawaser, 11991, Saudi Arabia. ²Department of Mathematics, Aligarh Muslim University, Aligarh, 202002, India.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 16 March 2020 Accepted: 22 June 2020 Published online: 06 July 2020

References

1. Acar, T., Mohiuddine, S.A., Mursaleen, M.: Approximation by (p, q) -Baskakov–Durrmeyer–Stancu operators. *Complex Anal. Oper. Theory* **12**(6), 1453–1468 (2018)
2. Ali, F.A.M., Karim, S.A.A., Saaban, A., Hasan, M.K., Ghaffar, A., Nisar, K.S., Baleanu, D.: Construction of cubic timmer triangular patches and its application in scattered data interpolation. *Mathematics* **8**(2), 159 (2020)
3. Ashraf, P., Nawaz, B., Baleanu, D., Nisar, K.S., Ghaffar, A., Khan, M.A.A., Akram, S.: Analysis of geometric properties of ternary four-point rational interpolating subdivision scheme. *Mathematics* **8**(3), 338 (2020)

4. Bernstein, S.N.: Constructive proof of Weierstrass approximation theorem. *Comm. Kharkov Math. Soc.* (1912)
5. Bézier, P.E.: *Numerical Control: Mathematics and Applications*. Wiley, London (1972)
6. Disibuyuk, C., Oruc, H.: Tensor product q -Bernstein polynomials. *BIT Numer. Math.* **48**, 689–700 (2008)
7. Farouki, T.R., Rajan, V.T.: Algorithms for polynomials in Bernstein form. *Comput. Aided Geom. Des.* **5**(1), 1–26 (1988)
8. Gadjiev, A.D., Ghorbanalizadeh, A.M.: Approximation properties of a new type Bernstein–Stancu polynomials of one and two variables. *Appl. Math. Comput.* **216**(3), 890–901 (2010)
9. Ghaffar, A., Bari, M., Ullah, Z., Iqbal, M., Nisar, K.S., Baleanu, D.: A new class of $2q$ -point nonstationary subdivision schemes and their applications. *Mathematics* **7**(7), 639 (2019)
10. Ghaffar, A., Iqbal, M., Bari, M., Hussain, S.M., Manzoor, R., Nisar, K.S., Baleanu, D.: Construction and application of nine-tic B-spline tensor product SS. *Mathematics* **7**(8), 675 (2019)
11. Ghaffar, A., Ullah, Z., Bari, M., Nisar, K.S., Al-Qurashi, M.M., Baleanu, D.: A new class of $2m$ -point binary non-stationary subdivision schemes. *Adv. Differ. Equ.* **2019**, 325 (2019)
12. Ghaffar, A., Ullah, Z., Bari, M., Nisar, K.S., Baleanu, D.: Family of odd point non-stationary subdivision schemes and their applications. *Adv. Differ. Equ.* **2019**, 171 (2019)
13. Han, L.-W., Chu, Y., Qiu, Z.: Generalized Bézier curves and surfaces based on Lupaş q -analogue of Bernstein operator. *J. Comput. Appl. Math.* **261**, 352–363 (2014)
14. Harim, N.A., Karim, S.A.A., Othman, M., Saaban, A., Ghaffar, A., Nisar, K.S., Baleanu, D.: Positivity preserving interpolation by using rational quartic spline. *AIMS Math.* **5**(4), 3762–3782 (2020)
15. Hussain, S.M., Rehman, A.U., Baleanu, D., Nisar, K.S., Ghaffar, A., Karim, S.A.A.: Generalized 5-point approximating subdivision scheme of varying arity. *Mathematics* **8**(4), 474 (2020)
16. Kac, V., Cheung, P.: *Quantum Calculus*. Universitext Series, vol. IX. Springer, Berlin (2002)
17. Karim, S.A.A., Saaban, A., Skala, V., Ghaffar, A., Nisar, K.S., Baleanu, D.: Construction of new cubic Bézier-like triangular patches with application in scattered data interpolation. *Adv. Differ. Equ.* **2020**, 151 (2020)
18. Khan, K., Lobiyal, D.K.: Bézier curves based on Lupaş (p, q) -analogue of Bernstein functions in CAGD. *J. Comput. Appl. Math.* **317**, 458–477 (2017)
19. Khan, K., Lobiyal, D.K., Kilicman, A.: A de Casteljau algorithm for Bernstein type polynomials based on (p, q) -integers. *Appl. Appl. Math.* **13**(2), 997–1017 (2018)
20. Khan, K., Lobiyal, D.K., Kilicman, A.: Bézier curves and surfaces based on modified Bernstein polynomials. *Azerb. J. Math.* **9**(1), 3–21 (2019)
21. Korovkin, P.P.: *Linear Operators and Approximation Theory*. Hindustan Publishing Corporation, Delhi (1960)
22. Lorentz, G.G.: *Bernstein Polynomials*. University of Toronto Press, Toronto (1953)
23. Lupaş, A.: A q -analogue of the Bernstein operator. *Semin. Numer. Stat. Calc., Univ. Cluj-Napoca* **9**, 85–92 (1987)
24. Mishra, V.N., Pandey, S.: On (p, q) -Baskakov–Durrmeyer–Stancu operators. *Adv. Appl. Clifford Algebras* (2016). <https://doi.org/10.1007/s00006-016-0738-y>
25. Mursaleen, M., Ansari, K.J., Khan, A.: On (p, q) -analogue of Bernstein operators. *Appl. Math. Comput.* **266**, 874–882 (2015) [Erratum: 278, 70–71 (2016)]
26. Mursaleen, M., Ansari, K.J., Khan, A.: Some approximation results by (p, q) -analogue of Bernstein–Stancu operators. *Appl. Math. Comput.* **264**, 392–402 (2015) Corrigendum: *Appl. Math. Comput.* **269**, 744–746 (2015)
27. Mursaleen, M., Ansari, K.J., Khan, A.: Approximation properties and error estimation of q -Bernstein shifted operators. *Numer. Algorithms* **84**, 207–227 (2020)
28. Mursaleen, M., Khan, A.: Generalized q -Bernstein–Schurer operators and some approximation theorems. *J. Funct. Spaces Appl.* **2013**, Article ID 719834 (2013). <https://doi.org/10.1155/2013/719834>
29. Mursaleen, M., Khan, F., Khan, A.: Approximation by (p, q) -Lorentz polynomials on a compact disk. *Complex Anal. Oper. Theory* **10**(8), 1725–1740 (2016)
30. Mursaleen, M., Nasiruzzaman, M., Khan, A., Ansari, K.J.: Some approximation results on Bleimann–Butzer–Hahn operators defined by (p, q) -integers. *Filomat* **30**(3), 639–648 (2016)
31. Oruc, H., Phillips, G.M.: q -Bernstein polynomials and Bézier curves. *J. Comput. Appl. Math.* **151**, 1–12 (2003)
32. Ostrovska, S.: On the Lupaş q -analogue of the Bernstein operator. *Rocky Mt. J. Math.* **36**(5), 1615–1629 (2006)
33. Phillips, G.M.: Bernstein polynomials based on the q -integers. *Ann. Numer. Math.* **4**, 511–518 (1997)
34. Rababah, A., Manna, S.: Iterative process for G_2 -multi degree reduction of Bézier curves. *Appl. Math. Comput.* **217**, 8126–8133 (2011)
35. Sederberg, T.W.: *Computer aided geometric design course notes*. Department of Computer Science, Brigham Young University (2014)
36. Stancu, D.D.: Approximation of functions by a new class of linear polynomial operators. *Rev. Roum. Math. Pures Appl.* **13**, 1173–1194 (1968)
37. Wafi, A., Rao, N.: Bivariate–Schurer–Stancu operators based on (p, q) -integers. *Filomat* **32**(4), 1251–1258 (2018)
38. Wafi, A., Rao, N.: (p, q) -Bivariate–Bernstein–Chlowdosj operators. *Filomat* **32**(2), 369–378 (2018)
39. Zulkifli, N.A.B., Karim, S.A.A., Shafie, A.B., Sarfraz, M., Ghaffar, A., Nisar, K.S.: Image interpolation using a rational bi-cubic Ball. *Mathematics* **7**(11), 1045 (2019)