Annals of the University of Craiova, Mathematics and Computer Science Series
Volume 49(2), 2022, Pages 445–460, DOI: 10.52846/ami.v49i2.1703
ISSN: 1223-6934

Characterization of a cubic interpolation scheme dependent on two parameters and applications

Dana Simian, Oana-Adriana Ţicleanu, and Nicolae Constantinescu

Abstract. The aim of this paper is to provide a characterization diagram for a family of Bézier flexible interpolation curves as well as to present an application of our results in cryptography. In our interpolation scheme, two parameters, $t_1,\ t_2 \in (0,1)$ determine the position of the interpolation points on the Bézier curve. Consequently we obtain a family of Bézier interpolation curves depending on two parameters. Altering the values of the parameters we modify the intermediary control points and implicitly the shape of the interpolation curve. In order to control the shape of the interpolation curves from this family, we provide a partition of the domain $T = (0,1) \times (0,1)$ where the parameters lie according to the geometric characterization of these curves: with zero, one or two inflexion points; with loop; with cusp and degenerated in quadratic curves. The characterization diagram can be used as a tool for the choice of parameters, with possible applications in different fields. We present one of its application in cryptography, for finding certain subspaces over which particular elliptic sub-curves are defined. Computation, implementation and graphics are made using MATLAB.

2010 Mathematics Subject Classification. Primary 65D05; Secondary 14H52.
Key words and phrases. Bézier curves, Elliptic curves.

1. Introduction

Bézier techniques have a wide applicability in curves' and surfaces' generation in CAGD (Computer Aided Geometric Design). We refer to ([4]) and the references therein for a comprehensive presentation of topics on Bézier curves and surfaces. From computational reasons, quadratic and cubic Bézier curves are usually used. Bézier curves are parametric curves defined using combinations of control points and basis functions. The classic Bézier curves uses Bernstein basis ([4]), but trigonometric ([2]), harmonic ([11]) and rational bases ([2]) have a wide applicability in geometric modeling. The attempts of changing the shape of the curve without explicitly modifying the control points lead to introduction of Bézier curves with shape parameters. Different bases with one ore many parameters were defined, allowing the shape modification under fixed control points. Wang W. and Wang G. defined recursively, using an integral approach, a Bézier basis with one shape parameter ([14]). This basis has most properties of Bernstein basis and modifying the parameter are obtained Bézier curves with different shape with the same control polygon. An extension to the classical Bernstein basis functions of degree n using $n-1$ local shape control parameters is presented in ([10]). The new class of basis functions satisfies the conditions required both for continuity and geometric continuity up to second order.The joining of the extended Bézier curves is smoother than that of classical Bézier curves

Received October 8, 2022. November 26, 2022.

of degree n. Han introduced in ([6]) a class of cubic trigonometric polynomial curves with one parameter and in ([7]) a class of cubic trigonometric polynomial curves with two parameters. The presence of the parameters gives a easier control of the curve shape and allows to positioning it near to the control polygon. A rational quadratic trigonometric basis is defined in ([2]). The new curves provide best approximation of the traditional rational quadratic Bézier curve and represent exactly some quadratic trigonometric curves like ellipse and circle.

An important problem to be solved in the case of parametric curves dependent on parameters consists in construction of the shape diagram. The shape diagram represents the curve characterization on the space of parameters and provides a tool for identifying the presence of loops, cusps, or inflection points, therefore allowing the choice of suitable values of parameters in order to avoid these unwanted curves configurations. Such kind of characterizations have been done before, for different particular cases of parametric curves, using different methods. A geometric characterization of parametric cubic curves using algebraic method is given by Koprowski in ([9]); Wang, in ([15]), used algebraic methods for B-splines characterization; Su and Liu, in ([13]), presented a geometric approach for Bézier curves' characterization; Forrest, in ([5]), has taken into account the rational cubic curves. A shape analysis of cubic trigonometric Bézier curves with a shape parameter can be found in ([6]). In ([12]), Stone, Parc and Derose present the characterization diagram for Bézier curves in canonical form, that is the coordinates of three of the control points are fixed and the curve shape depends only on the position of the moving point about the plane. The coordinates of the this variable point take the role of the parameters. In ([6]) the influence of the shape parameters for cubic trigonometric Bézier curves with a shape parameter is revealed using a shape diagram.

The aim of this paper is to introduce a family of cubic Bézier curves dependent on two parameters and to make a geometric characterization of these curves. Using parameters $(t_1, t_2) \in (0,1)^2$ we construct cubic Bézier curves satisfying Lagrange type conditions in the two end points and in other two intermediate points. The choice of parameters t_1 and t_2 based on the characterization diagram proposed in this article can be also useful in other fields than computational geometry. In the end of our article, we illustrated the applicability of our main results in cryptography. The rest of the article is organized as follows. In Section 2 we introduce our flexible interpolation scheme. In Section 3 we illustrate the influence of the parameters on the Bézier curve's shape using several numerical examples. Section 4 contains the main theoretical results regarding the characterization of the proposed flexible cubic interpolation Bézier curves. Section 5 is dedicated to the characterization diagram. In Section 6 we illustrate the application of our flexible Bézier interpolation curves and of the characterization diagram in cryptography. Section 7 contains conclusions and further directions of study.

2. Our cubic flexible interpolation scheme

We start from the classic interpolation problem by cubic Bézier curves:

Problem 1. Given four points $P_i(px_i, py_i)$, $i = 0, \ldots, 3$, find a cubic Bézier curve passing through them.

No other details referring the shape of curve are given. There are many interpolation cubic polynomials satisfying the interpolation conditions. In Computer Aided Geometric Design it is of interest to analyze and compare these interpolation curves in order to avoid some shapes (curves with loops or cusps), or in order to find the most appropriate shape (in design of cars' body). A cubic Bézier curve in plane is a parametric curve totally defined by its four control points, $b_i = (bx_i, by_i)$, $i \in \{0, \ldots, 3\}$. The parametric equations of a cubic planar Bézier curve can easily be written in the matrix form:

$$f(t) = b \cdot B(t), \tag{1}$$

where $f(t) = (x(t), y(t))^\tau \in \mathcal{M}_{2,1}$ represents the coordinates vector of a current point on the Bézier curve; $b \in \mathcal{M}_{2,4}$ is the matrix of control points coordinates and $B(t) \in \mathcal{M}_{4,1}$ is the vector of Bernstein polynomials $B_i^3(t)$:

$$B_i^3(t) = \binom{3}{i}(1-t)^{3-i}t^i, \quad i = 0, \ldots, 3 \tag{2}$$

We denoted by M^τ the transpose of matrix M. A common way to solve the interpolation problem given in *Problem 1* is to consider that the interpolation points correspond to a uniform sequence of values for the curve parameter t, that is the interpolation conditions are:

$$f(0) = P_0;\ f(1/3) = P_1;\ f(2/3) = P_2;\ f(1) = P_3. \tag{3}$$

In our approach we renounce to the uniformity conditions and reformulate the interpolation conditions using two parameters $t_1, t_2 \in (0, 1)$, as follows:

$$f(0) = P_0;\ f(t_1) = P_1;\ f(t_2) = P_2;\ f(1) = P_3. \tag{4}$$

We mentioned for the first time, in [18], the possibility of obtaining flexible interpolation Bézier curves and surfaces by renouncing to the uniformity conditions for the interpolation points on the domain of parameters.

In order to find the Bézier interpolation curve our purpose is to find the control points using the conditions (4). These interpolation conditions led to the system $D \cdot b = P$, with D given in (4). By an easy algebraic calculus, using (1) and (4) we obtain the expression of the control points coordinates:

$$b^\tau = inv(D) \cdot P^\tau, \tag{5}$$

with $P = (px_i, py_i)^\tau \in \mathcal{M}_{2,4}$ and

$$D = \begin{pmatrix} 1 & 0 & 0 & 0 \\ (1-t_1)^3 & 3t_1(1-t_1)^2 & 3t_1^2(1-t_1) & t_1^3 \\ (1-t_2)^3 & 3t_2(1-t_2)^2 & 3t_2^2(1-t_2) & t_2^3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{6}$$

The system (5) has a unique solution because

$$\det(D) = 9t_1t_2(1-t_1)(1-t_2)(t_2-t_1) \neq 0, \forall t_1,\ t_2 \in (0,1). \tag{7}$$

We observe that three limit cases could appear:

$$\lim_{t_i \to 0} \det(D(t_1, t_2)) = 0; \quad \lim_{t_i \to 1} \det(D(t_1, t_2)) = 0; \quad \lim_{t_1 \to t_2} \det(D(t_1, t_2)) = 0. \tag{8}$$

where $i \in \{1, 2\}$.

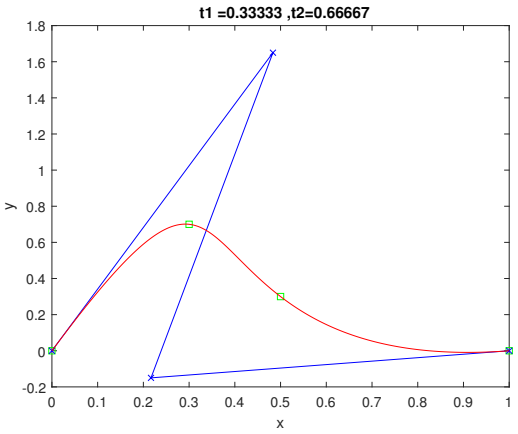

FIGURE 1. Bézier curve obtained for $t_1 = 1/3$ and $t_2 = 2/3$.

Using symbolic calculus in MATLAB we obtained

$$inv(D) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{-(t_1+t_2-2t_1t_2)}{3t_1t_2} & \frac{t_2}{3t_1(t_2-t_1)(1-t_1)} & \frac{-t_1}{3t_2(t_2-t_1)(1-t_2)} & \frac{t_1t_2}{3(1-t_1)(1-t_2)} \\ \frac{(1-t_1)(1-t_2)}{3t_1t_2} & \frac{-(1-t_2)}{3t_1(t_2-t_1)(1-t_1)} & \frac{1-t_1}{3t_2(t_2-t_1)(1-t_2)} & \frac{-(t_1+t_2-2t_1t_2)}{3(1-t_1)(1-t_2)} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (9)$$

3. Numerical examples

Let $P_0 = (0,0)$, $P_1 = (0.3, 0.7)$, $P_2 = (0.5, 0.3), P_3 = (1,0)$ be the interpolation points. Figure 1 shows the Bézier interpolation curve obtained in the case of uniform sequence of parameter values, that is for $t_1 = 1/3$ and $t_2 = 2/3$. The shape of the curve is concave-convex. In order to underline the effect of the parameters t_1 and t_2 on the shape of the curve we make a sensitivity analyses on t_1 and t_2. In this regards, first we keep $t_1 = 1/3$ and modify the value of t_2 and, second, we keep $t_2 = 2/3$ and modify the value of t_1. For $t_1 = 1/3$, we observe that for $t_2 = 0.001$ the curve tends to a straight line; for $t_2 = 0.1$ and $t_2 = 0.2$ the curve has a loop; for $t_2 = 0.33$, the curve is closed to a straight line; for $t_2 \in \{0.5, 0.6, 0.7\}$ the curve is again concave-convex; for $t_2 = 0.8$ and $t_2 = 0.9$ the curve presents a cusp; for $t_2 = 0.9$ the curve has a loop and for $t_2 = 0.99$ the curve is again closed to a straight line. Some of these cases are presented in figures Fig.2-Fig.4. The limit cases given in (8) are depicted in figures Fig.5 - Fig.7.

If we keep $t_2 = 2/3$ and modify t_1, we observe that for t_1 close to 0 and 2/3 the curve tends to a straight line. For $t_1 \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$ the curve is concave-convex.

4. Characterization of the cubic flexible interpolation Bézier curves

4.1. Problem formulation. In *Problem 1* formulated in section 2, the coordinates of the interpolation points are input data. Therefore the control points $b_0 = P_0$

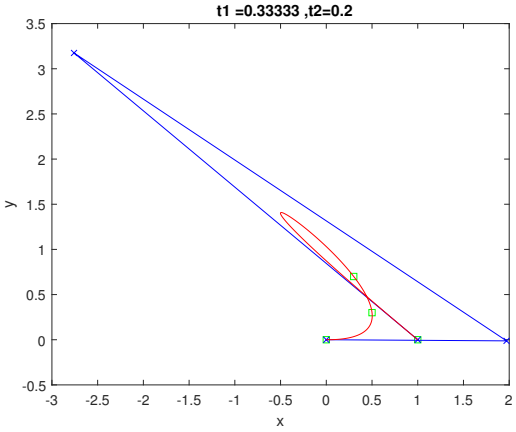

FIGURE 2. Bézier curve with loop, obtained for $t_1 = 1/3$ and $t_2 = 0.2$

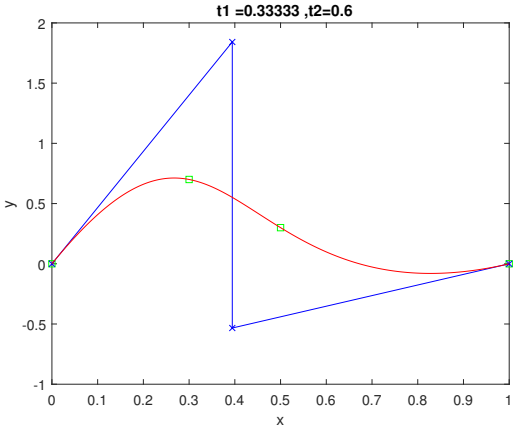

FIGURE 3. Bézier curve concave-convex, obtained for $t_1 = 1/3$ and $t_2 = 0.6$

and $b_3 = P_3$ are also fixed. When (t_1, t_2) move in $(0,1)^2$ the control points $b_1,\ b_2$ move in the real plane, causing the change of the curve shape. We want to provide a characterization diagram in coordinates (t_1, t_2) which allows us to chose the shape of the interpolation curve by choosing an appropriate pair of parameters.

The affine invariance of Bézier curves allows us to use an affine mapping $\phi : R^2 \to R^2$, such that $\phi(P_0) = (0,0)$ and $\phi(P_3) = (1,0)$ and to work with the interpolation points $\phi(P_i)$, instead of P_i, $i \in \{0, \ldots, 3\}$ in order to make the characterization diagram. Obviously ϕ is a composition of affine geometric transformations. We will refer to the points $\phi(P_i)$ as the canonical interpolation points. In the rest of the paper we will work with canonical interpolation points and we will denote them like the original ones, that is P_i. We make the assumption that the coordinates of the canonical interpolation points $P_1,\ P_2$, satisfy the inequalities $0 < px_1 < px_2 < 1$.

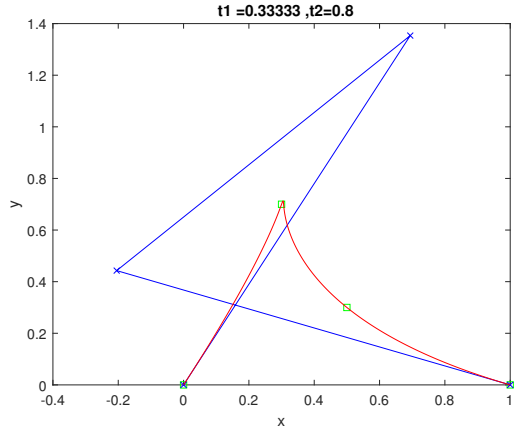

FIGURE 4. Bézier curve with cusp, obtained for $t_1 = 1/3$ and $t_2 = 0.8$

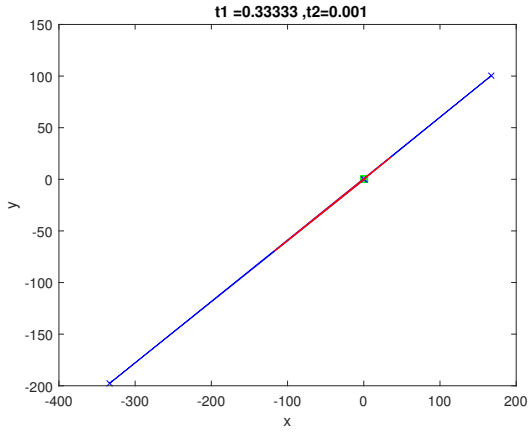

FIGURE 5. The limit case $t_2 \to 0$

This assumption can be made be cause reversing the control points of Bézier curve reverses only the parametrization without affecting the shape of the curve.

After an algebraic calculus, substituting the coordinates of control points given by (5) into equation (1), we obtain the equation of the family of Bézier curves:

$$f_{t_1,t_2}(t) = \begin{pmatrix} Cx_3 \cdot t^3 + Cx_2 \cdot t^2 + Cx_1 \cdot t \\ Cy_3 \cdot t^3 + Cy_2 \cdot t^2 + Cy_1 \cdot t \end{pmatrix}, \tag{10}$$

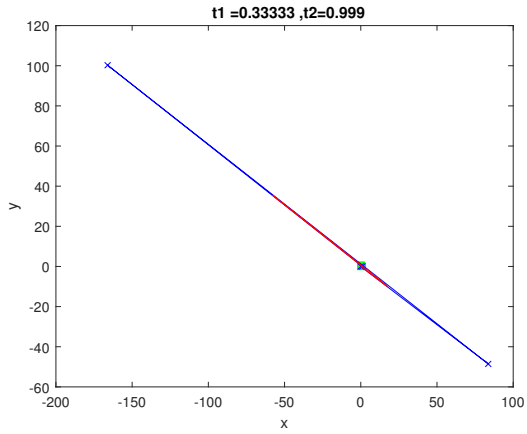

FIGURE 6. The limit case $t_2 \to 1$

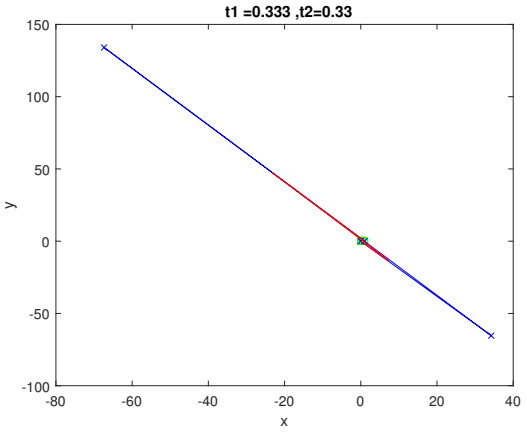

FIGURE 7. The limit case $t_1 \to t_2$

with

$$Cx_3 = \frac{1}{(1-t_1)(1-t_2)} + \frac{px_1}{t_1(t_2-t_1)(1-t_1)} - \frac{px_2}{t_2(t_2-t_1)(1-t_2)} \tag{11}$$

$$Cx_2 = \frac{-px_1(1+t_2)}{t_1(t_2-t_1)(1-t_1)} + \frac{px_2(1+t_1)}{t_2(t_2-t_1)(1-t_2)} - \frac{t_1+t_2}{(1-t_1)(1-t_2)} \tag{12}$$

$$Cx_1 = \frac{t_1 \cdot t_2}{(1-t_1)(1-t_2)} + \frac{px_1 \cdot t_2}{t_1(t_2-t_1)(1-t_1)} - \frac{px_2 \cdot t_1}{t_2(t_2-t_1)(1-t_2)} \tag{13}$$

$$Cy_3 = \frac{py_1}{t_1(t_2-t_1)(1-t_1)} - \frac{py_2}{t_2(t_2-t_1)(1-t_2)} \tag{14}$$

$$Cy_2 = \frac{-py_1(1+t_2)}{t_1(t_2-t_1)(1-t_1)} + \frac{py_2(1+t_1)}{t_2(t_2-t_1)(1-t_2)} \tag{15}$$

$$Cy_1 = \frac{py_1 \cdot t_2}{t_1(t_2-t_1)(1-t_1)} - \frac{py_2 \cdot t_1}{t_2(t_2-t_1)(1-t_2)} \tag{16}$$

The control points are given by:

$$b_0 = (0,0) \tag{17}$$

$$bx_1 = \frac{t_1 \cdot t_2}{3(t_1-1)(t_2-1)} + \frac{px_1 \cdot t_2}{3t_1(t_1-t_2)(t_1-1)} - \frac{px_2 \cdot t_1}{3t_2(t_1-t_2)(t_2-1)} \tag{18}$$

$$by_1 = \frac{py_1 \cdot t_2}{3t_1(t_1-t_2)(t_1-1)} - \frac{py_2 \cdot t_1}{3t_2(t_1-t_2)(t_2-1)} \tag{19}$$

$$bx_2 = \frac{px_1(t_2-1)}{3t_1(t_1-t_2)(t_1-1)} - \frac{t_1+t_2-2\cdot t_1\cdot t_2}{3(t_1-1)(t_2-1)} - \frac{px_2(t_1-1)}{3t_2(t_1-t_2)(t_2-1)} \tag{20}$$

$$by_2 = \frac{py_1(t_2-1)}{3t_1(t_1-t_2)(t_1-1)} - \frac{py_2(t_1-1)}{3t_2(t_1-t_2)(t_2-1)} \tag{21}$$

$$b_3 = (1,0) \tag{22}$$

If the interpolation points are collinear, the canonical interpolation points are situated on the Ox axis and using the expression of the control points we obtain that all the control points are situated on Ox axis and therefore the Bézier interpolation curve degenerate also in a segment from Ox axis. This prove the linear precision of our interpolation scheme.

4.2. Main theoretical results.

Theorem 4.1. *Let $\mathcal{P}$ be a set of four interpolation points. Let be $\phi : R^2 \to R^2$ a change of coordinates such that $P_0 = (0,0)$ and $P_3 = (1,0)$. We also assume that, after this change of coordinates the other 2 interpolation points $P_j(px_j, py_j)$, $j \in \{1,2\}$ satisfy the condition: $0 < px_1 < px_2 < 1$. Let f be the interpolation cubic Bézier curve given in (10). Then*

Case 1. *If the system $Cx_3 = 0$; $Cy_3 = 0$, with Cx_3, Cy_3 defined in (11) and (14), has a solution $(\tilde{t}_1, \tilde{t}_2)$, in $(0,1)^2$, $\tilde{t}_1 \neq \tilde{t}_2$, then the cubic Bézier curve f, obtained for these values of parameters (t_1, t_2), degenerates into a parabola.*

Case 2. *The case of the nondegenerated curve.*

In this case we have the following subcases:

Case 2.1 *If*

$$A_1 = d + py_1 \cdot t_2 - py_2 \cdot t_1 = 0 \tag{23}$$

the Bézier curve has exactly one inflexion point.

Case 2.2. *If $A_1 \neq 0$ and $\Delta_1 = 0$, with*

$$\Delta_1 = 3(py_1 \cdot t_2^2 - py_2 \cdot t_1^2 + d)^2 - 4(py_1 \cdot t_2^3 - py_2 \cdot t_1^3 + d) \cdot (d - py_2 \cdot t_1 + py_1 \cdot t_2) \tag{24}$$

the Bézier curve has a cusp point. We notice that d is defined in (31).

Case 2.3. *If $A_1 \neq 0$ and $\Delta_1 < 0$ the Bézier curve has a loop point.*

Case 2.4. *If $A_1 \neq 0$ and $\Delta_1 > 0$ the Bézier curve has zero, one or two inflection points. The number of inflexion points is equal to the number of the roots of equation (25), situated in the interval [0,1].*

Proof. We use the following lemmas formulated taking into account the results presented in ([9], [13], [15]) and also used in ([12]).

Lemma 4.2. *([9], [13], [15])A parametric cubic curve can not have at the same time loop, cusps or inflexion point. Moreover, a nondegenerate parametric cubic curve can not have more than one cusp, one loop or two inflexion points.*

Lemma 4.3. *([13],[15]) Let $f(t) = (x(t), y(t))$ be a parametric cubic curve and*

$$F(t) = det \begin{pmatrix} f'(t) \\ f''(t) \end{pmatrix} = x'(t)y''(t) - x''(t)y'(t) \tag{25}$$

Then $F(t)$ is a quadratic form $F(t) = At^2 + Bt + C$, which characterizes the presence of loop, cusp or inflexion points. Let denote

$$\Delta = B^2 - 4AC, \tag{26}$$

then

(1) *If $A = 0$, then there is exact one inflexion point.*
(2) *If $A \neq 0$ and $\Delta > 0$ there are exact two inflexion points.*
(3) *If $A \neq 0$ and $\Delta < 0$ there is a loop.*
(4) *If $A \neq 0$ and $\Delta = 0$ there is a cusp.*

We make the observation that $F(t)$ is proportional with the signed curvature of the curve at the point $(x(t), y(t))$. At the inflection points the first and second derivative vectors are linearly dependent therefore at these points $F(t) = 0$.

Lemma (4.3) is formulated for the case of untrimmed cubic curves, without restricted the domain of parameter t. In the case of cubic Bézier curves we restrict the domain of t to $[0, 1]$, therefore, if $A \neq 0$ and $\Delta > 0$ the curve can have zero, one or two inflection points depending on the number of roots of $F(t)$ which fall on the interval $[0, 1]$.

The computation made in the case of our Bézier interpolation curve gives:

$$A = 3a(d + py_1 \cdot t_2 - py_2 \cdot t_1) \tag{27}$$
$$B = -3a(py_1 \cdot t_2^2 - py_2 \cdot t_1^2 + d) \tag{28}$$
$$C = a(py_1 \cdot t_2^3 - py_2 \cdot t_1^3 + d) \tag{29}$$
$$a = \frac{2}{t_1 t_2 (t_2 - t_1)(1 - t_1)(1 - t_2)} \tag{30}$$
$$d = px_1 \cdot py_2 - px_2 \cdot py_1 \tag{31}$$

We remark that $A_1 = \dfrac{A}{3a}$ and $\Delta_1 = \dfrac{\Delta}{3a^2}$ with A and Δ defined in Lemma (4.3). Obviously $sign(\Delta_1) = sign(\Delta)$ and, for $0 < t_1 \neq t_2 < 1$, we have $A1 = 0$ iff $A = 0$. □

The linear equation $A_1(t_1, t_2) = 0$, represents a line that we call, like in ([12]), "the one inflection point line". We will name "the cusp curve", the curve given by the equation $\Delta_1(t_1, t_2) = 0$.

We will discuss in detail the cases of degenerate cubic Bézier curve.

Case 1. Analyzing the relations (8) we can define 3 limit cases in which the cubic Bézier curve degenerates to a straight line: $t_i = 0, t_i = 1$, $i \in \{0, 1\}$ and $t_1 = t_2$. The first 2 cases were excluded from the domain. The third one, was excluded by the condition $px_1 < px_2$. Therefore the domain used in our characterization diagram will be

$$T_1 = (0, 1)^2 \setminus \{(t_1, t_2) | t_1 = t_2\} \tag{32}$$

Case 2. The case in which the cubic Bézier curve degenerates into a parabola, presented in Theorem (4.1), appears when the parameters are on the form:

$$(t_1, t_2) = \left(\frac{-d \cdot py_2 \pm \sqrt{s}}{py_2(py_1 - py_2)}, \frac{-d \cdot py_1 \pm \sqrt{s}}{py_1(py_1 - py_2)} \right), \tag{33}$$

with d given in (31) and $s = d \cdot py_1 \cdot py_2(py_1 - py_2 + d)$. We will denote the points $(t_1, t_2) \in (0,1)^2$ satisfying (33) with (tp_1, tp_2) and will name parabolic points.
Case 2.1: If $s = 0$ there is one parabolic point:

$$(tp_1, tp_2) = \left(\frac{-d}{py_1 - py_2}, \frac{-d}{py_1 - py_2} \right) \tag{34}$$

Taking into account the expression of s, the following subcases could occur:
Case 2.1.1: $d = 0$.
In this case the parabolic point $(td_1, td_2) = (0,0) \notin T_1$. Actually in this case the "cusp curve" degenerate into two lines and cubic Bézier degenerate into a line.

Case 2.1.2: $py_1 - py_2 - d = 0$, or equivalent $\dfrac{py_1}{py_2} = \dfrac{1 - px_1}{1 - px_2}$.
We obtain the parabolic point $(td_1, td_2) = (1,1) \notin T_1$. As in the previous case the "cusp curve" degenerate into two lines and cubic Bézier degenerate into a line.

Case 2.1.3: $py_2 = 0$ or $py_1 = 0$. In this case the Bézier curve degenerate into a line segment.

We can observe directly from (34) and (32) that $(tp_1, tp_2) \notin T_1$.

Case 2.2: If $s < 0$ we do not have parabolic points and for any values of the parameters the cubic curve does not degenerate into a parabola.

Case 2.3: If $s > 0$ it is possible to have zero, one or two parabolic points depending on whether the pairs (tp_1, tp_2) fall on the domain $(0,1)^2$.

5. Characterization diagram

Separating the region T_1 given in (32) by the inflexion line and the cusp curve and labeling the subregions using the sign of Δ we obtain a "characterization diagram" from which we can extract different pairs of parameters so we get the desired shape of the interpolation curve. We underline again that our "characterization diagram" is conceptual different from other existent characterization diagrams. For each set of interpolation points we obtain a different "characterization diagram". It is very difficult if not almost impossible to make a theoretic discussion for all sets of interpolation points.
The problem of representing the "characterization diagram" for a given set of interpolation points was solved by implementing the theoretic results presented before in this section, using Matlab Symbolic Toolbox and the following algorithm:

In figure (3), we illustrate the characterization diagram obtained for the particular choice of canonical interpolation points used in section (3): $P_1 = (0.3, 0.7)$, $P_2 = (0.5, 0.3)$. The diagram makes clear now the results presented in section (3). A vertical line raised on $t_1 = 1/3$ intersects all the regions of the diagram, showing us that the shape of the curve changes significant for different values of t_2.
Fig. 6 shows the characterization diagram obtained for the canonical interpolation points: $P_1 = (0.1, -0.5)$, $P_2 = (0.6, 0.3)$. We observe that in this case the "one inflexion point line" does not intersect the domain $[0,1]^2$. In the diagram appear only two "cusp curves" and the degenerate case $t_1 - t_2 = 0$.

Algorithm Characterization Diagram

Step I: Symbolic part of algorithm

- Compute the matrices D and inv(D): see (6) and (9).
- Compute the coordinates of the control points: see (5).
- Compute Cx_3, Cy_3 and find the expression of parabolic points, (tp_1, tp_2): see (11), (14) and (33).
- Compute A, B, C and Δ: see (27) - (29) and (26).
- Find the "symbolic" equation of the "one inflexion point line": $A(t_1, t_2) = 0$.
- Find the "symbolic" equation of the "cusp curve": $\Delta(t_1, t_2) = 0$.

Step II: Specific part of algorithm for each set of interpolation points

- Input: coordinates of interpolation points
- Obtain the the canonical form of the interpolation points. $Pcanonical = \phi(P) = a * P + b$, with $a = (1/(px3 - px0), 1/(py3 - py0))^\tau$ and $b = (-px0/(px3 - px0), -py0/p(y3 - py0))^\tau$.
- Represent in the same graphic the "one inflexion point line", "cusp curve" and limit case $t_1 = t_2$.
- Extract points from each region of T_1 using the function getinput.
- Compute the sign of Δ for these regions and label them.

Step III: Graphic representations

- Extract pairs of parameters (points) from the desired regions of characterization diagram and make the representation of Bézier curve, using (1) and (5).

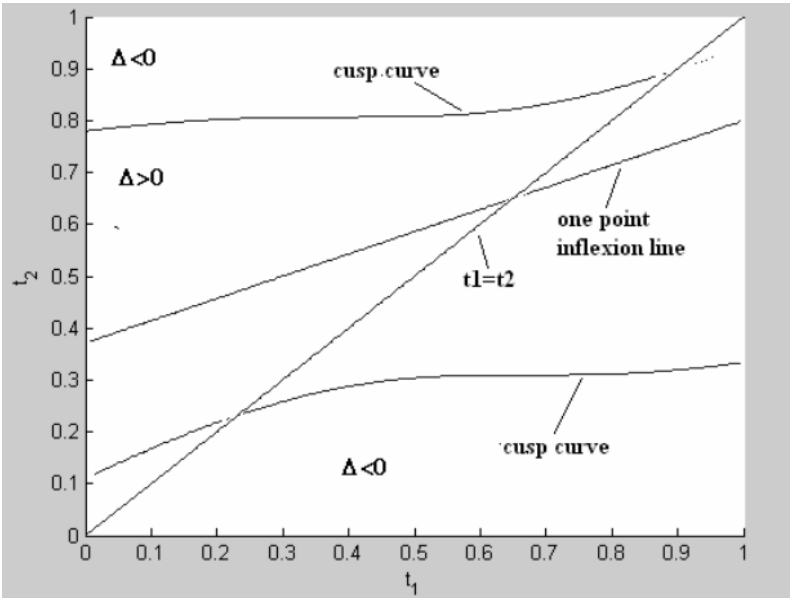

FIGURE 8. Characterization diagram for $P_1 = (0.3, 0.7)$, $P_2 = (0.5, 0.3)$.

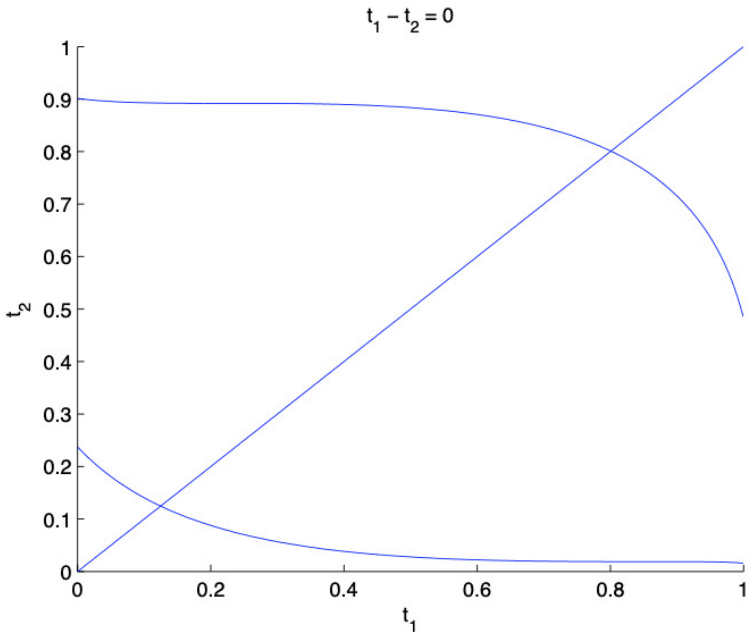

FIGURE 9. Characterization diagram for $P_1 = (0.1, -0.5)$, $P_2 = (0.6, 0.3)$.

6. Application in cryptographic authentication

Within the parameter constructions that form the necessary values for a cryptographic system, methods are used to extract spaces with applicability in the field of functions for which it is computationally feasible (calculation times are determined by algorithms of polynomial complexity) to calculate a value y, knowing the value of x and the calculation function f, but it is difficult (it is not computationally feasible) to calculate x, when the values of y and the function f are known. These functions are called hardly invertible functions. They are used in cryptographic processes, to generate the parameters that enter into the composition of certain encryption processes, to secure data or to authenticate entities that want to enter into the data communication processes.

In this respect, the first methods used were those based on RSA-type systems, methods developed starting from the algorithms created by Rivest, Shamir and Adleman ([17, 1]). These methods have as their main disadvantage, for the systems they fit with certain values from the MIDP (Mobile Information Device Profile), the fact that the computing power required to generate the intermediate parameters is greater than the computing resources and memory of these devices.

For these systems, computational models based on spaces defined by certain elliptic curves, with particular properties, have been proposed and are used. Several such models are used, depending on their applicability directions, and the systems where they want to be implemented. In practice, according to the studied mathematical models as well as the implementations in functional systems, the authors proposed solutions for several models of systems based on non-supersingular elliptic curves.

The studies of the mathematical models involved, carried out on various cases, have highlighted the advantages brought by the use of certain subspaces from the general space over which an elliptic curve is defined. This construction model leads to increasing the degree of attack resistance of the implemented cryptographic systems, by using ([3]) space quanta. For such mathematical models, the primary study factor is to define the way of choosing these quantiles and the way to do the operations with elements from this subspace, in view of the possibility of implementing the proposed model in cryptographic algorithms. For this we will construct a morphism from the space defined over the particular Bézier curves, according to the study from section (4.2) and the highlighted results. The morphism will be built from these spaces to a space over which an elliptic curve is defined, in this way we will define the quanta of spaces, which are actually subspaces of the general space, and these will become the multitudes of values from which elements with cryptographic properties will be selected.

6.1. Subspace construction. We start by defining the general elliptic curve, from which the entire construction starts.

Definition 6.1 ([3])**.** Let be $\tau > 3$ a prime integer. We define it as an elliptic degree curve of τ order, according with equation

$$\sigma^2 = \rho^3 + \eta_1\rho + \eta_2 \tag{35}$$

which takes values from the space of $\mathtt{Z}_\tau$, as being a set of solutions of the form $(\rho, \sigma) \in \mathtt{Z}_\tau \times \mathtt{Z}_\tau$, having satisfied the congruence

$$\sigma^2 \equiv \rho^3 + \eta_1\rho + \eta_2 \ (\ mod\ \tau) \tag{36}$$

with the values η_1 and η_2 from the space of $\mathtt{Z}_\tau$, being constants that satisfy the condition

$$4\eta_1^3 + 27\eta_2^2 \not\equiv 0(\ mod\ \tau) \tag{37}$$

To these, is added a special point $\mathcal{O}$, called point at infinity.

For the studied case, we will establish subspaces over which we will define sets of elliptic curves, in this way creating specific quantities over which sets of elliptic curves are defined, from which points with cryptographic properties will be selected. For this we will define the form of such a quantile and the associated elliptic curve, starting from the general definition, presented in Definition 6.1.

Definition 6.2. Let ξ be a quantile in a $\mathtt{Z}_\tau$ space, which we denote by $\mathtt{Z}_\tau^\xi$. Let $\tau_\xi > 3$ be a prime integer value. An elliptic curve of order τ_ξ, determined by the equation $\sigma^2 = \rho^3 + \eta_1\rho + \eta_2$, with values in $\mathtt{Z}_\tau^\xi$, is given by a set of solutions of the form (ρ, σ) from $\mathtt{Z}_\tau^\xi \times \mathtt{Z}_\tau^\xi$, satisfying the following congruence $\sigma^2 \equiv \rho^3 + \eta_1\rho + \eta_2\ (\ mod\ \tau_\sigma)$. The space $\mathtt{Z}_\tau^\xi$ will be the quantile over which the values are defined. η_1 and η_2 will be chosen from this subspace, with respect to the property

$$\begin{cases} \eta_1 \not\equiv \tau(\ mod\ \tau_\xi) \\ \eta_2 \not\equiv \tau(\ mod\ \tau_\xi) \end{cases} \tag{38}$$

$4\eta_1^3 + 27\eta_2^2 \not\equiv 0(\ mod\ \tau_\xi)$, as well as the existence of a point $\mathcal{O}_\xi$, called point at infinity of the quantile ξ.

In this regard, a restricted morphism (39) will be defined over the space from Case 2 of the Theorem 4.1, where the point of inflection will have as its correspondent the point $\mathcal{O}$.

$$\upsilon = \begin{cases} \phi\left(\overline{\mathtt{R}^2_{(t_1,t_2)}}\right) \to \phi\left(\overline{\mathtt{Z}^2_\tau}\right) & \\ (px, py) \to (\rho_\tau, \sigma_\tau), & \textit{when the condition (23) is not fulfilled} \\ (px, py) \to \mathcal{O}, & \textit{when the condition } A_1 = d + py_1 t_2 - p_2 t_1 = 0, \\ & \textit{from Theorem 4.1, is fulfilled.} \end{cases} \tag{39}$$

In this way, for each pair of parameters (t_1, t_2) a quantile ξ can be computed. The Bézier curves restricted to these subspaces, corresponding to the chosen parameters, will define the quantiles needed in the computation of the subspaces over which the series of elliptic curves are defined.

6.2. Algorithm description. In accordance with the above study, we will conclude those obtained by describing the algorithm that generates subspaces over which are defined series of elliptic curves from which points with cryptographic properties can be extracted. The algorithm describes the method by which are chosen those subspaces that fulfill the conditions to be used in cryptographic processes.

Elliptic Curves of order τ_ξ generation

Step 1: Domain values reset: Are initialised the sets of parameters, according with conditions from Theorem 4.1

Step 2: (t_1, t_2) are chosen with corresponding quantile ξ

Step 3: will be computed the corresponding values of $\phi\left(\overline{\mathtt{Z}^2_\tau}\right)$

Step 4: using trace of Frobenius ([19]) the number of points on the corresponding elliptic curve is computed

Step 5: using the (33) and (34) formulas, the degrees of correlation of the corresponding points on the elliptic curve are computed, using conditions from (38). If we are in the situations in which has reduced number of points, according with ([20, 8]), the curve is declared without cryptographic properties and we return to step 1

Step 6: will be verified conditions from (37) formulas. If these are fulfilled, the corresponding quantile is extended, according with formula (36).

Step 7: return corresponding parameters for elliptic curve defined over $\left(\mathtt{Z}^\xi_\tau\right)^{(t_1,t_2)}$

This algorithm will be called for a defined number of times, according with the number of quantiles that will form the series of spaces over which the elliptic curves are defined. At each of its calls, the parameters (t_1, t_2) will be recorded along with the corresponding sets of values or $\emptyset$, if the computation of a quantile with cryptographic properties failed.

7. Conclusion and further directions of the study

The interpolation scheme introduced in this article provides a family of interpolation cubic Bézier curves dependent on two parameters. Different choices of parameters give different shapes of interpolation curves. The shape is characterized in term of presence

or absence of specific type of points: inflexion points, loop and cusp. Theoretical results valid for any set of interpolation points are formulated. The characterization diagram refers to the parameters' domain $T_1 \subset (0,1)^2$ and is obtained labeling the domain regions situated between the "cusp curve" and "one inflection point line". For any set of interpolation points results a particular characterization diagram. In order to obtain this diagram and to allow the parameters' choice and graphic representation of Bézier interpolation curve, we implemented in Matlab the algorithm presented in Section 5. If $\Delta > 0$ we know only that the curve could have zero, one, or two inflection points. The real situation is given by the third part of the algorithm, that is by the graphic of interpolation curve. The further direction of study consists in adding new components to the characterization diagram in order to separate the sub regions with zero, one and two inflection points, which will lead to an increase in the size of the quantiles τ_ξ generated by the previously created morphism, so to a greater complexity of the calculation according to the associated cryptographic analysis, to increase the degree of resistance to the attack of the generated subspaces.

Our scheme is very useful to construct flexible interpolation problems and offer an easy to use tool for the choice of parameters, which is highlighted by its applicability in the determination of certain subspaces over which particular elliptic sub-curves are defined, by modeling the quantiles of subspaces that contain points with cryptographic properties, by illustrating within the described mathematical model and its implementation in the Section 6.

Acknowledgement: The first author, Dana Simian, was supported from the project financed by Lucian Blaga University of Sibiu through the research grant LBUS-IRG-2022-08.

References

[1] K. Adesh, M. Yahya, V. Kumar, and V. Khan, A secure user authentication protocol using elliptic curve cryptography. *Journal of discrete mathematical sciences & Cryptography* **22** (2019), no. 4, 521–530. DOI: 10.1080/09720529.2019.1637155

[2] U. Bashir, M. Abbas, and J.M. Ali, The g2 and c2 rational quadratic trigonometric bézier curve with two shape parameters with applications, *Applied Mathematics and Computation* **219** (2013), no. 20, 10183–10197. DOI: 10.1016/j.amc.2013.03.110

[3] N. Constantinescu, Non supersingular elliptic curves – From theory to application. Algorithm attacks discussions, *Mathematica* **50** (2008), no. 2, 177–186.

[4] G. Farin, *Curves and surfaces for computer aided geometric design. A practical guide*, Academic Press, Boston, 1996.

[5] A.R. Forrest, The twisted cubic curve: A computer-aided geometric design approach, *Computer Aided Design* **12** (1980), no. 4, 165–172. DOI: 10.1016/0010-4485(80)90149-9

[6] X.-A. Han, X. Huang, and Y. Ma, Shape analysis of cubic trigonometric bézier curves with a shape parameter, *Applied Mathematics and Computation* **217** (2010), no. 6, 2527–2533. DOI: 10.1016/j.amc.2010.07.065

[7] X.-A. Han, Y. Ma, and X. Huang, The cubic trigonometric bézier curve with two shape parameters, *Applied Mathematics Letters* **22** (2009), no. 2, 226–231. DOI: 10.1016/j.aml.2008.03.015

[8] A. Karrar, R. Taher, and R. Ajeena, The elliptic scalar multiplication graph and its application in elliptic curve cryptography. *Journal of discrete mathematical sciences & Cryptography* **24** (2021), no. 6, 1793–1807. DOI: 10.1080/09720529.2021.1932896

[9] P. Koprowski, Algebraic approach to geometric characterization of parametric cubics, *J. for Geometry and Graphics* **11** (2007), no. 2, 173–178.

[10] X. Qin, G. Hu, N. Zhang, X. Shen, and Y. Yang, A novel extension to the polynomial basis functions describing bezier curves and surfaces of degree n with multiple shape parameters, *Applied Mathematics and Computation* **223** (2013), 1–16. DOI: 10.1016/j.amc.2013.07.073
[11] J. Sachez-Reyes, Harmonic rational bézier curves, p- Bézier curves and trigonometric polynomials, *Computed Aided Geometric Design* **15** (1998), no. 9, 909–923. DOI: 10.1016/S0167-8396(98)00031-4
[12] M. C. Stone, X. Parc, and T.D. Derose, A geometric characterization of parametric cubic curves, *ACM Transactions on Graphics* **8** (1989), no. 3, 147–163. DOI: 10.1145/77055.77056
[13] B. Su and D. Liu, An affine invariant and its application in computational geometry, *Scientia Sinica, Series A* **26** (1983), no. 3, 259–272.
[14] C. Y. Wang, Shape classification of the parametric cubic curve and parametric b-spline cubic curve, *Comput. Aided Des.* **13** (1981), no. 4, 199–206. DOI: 10.1016/0010-4485(81)90141-X
[15] W.-T. Wang and G.-Z. Wang, Bézier curves with shape parameter, *Journal of Zhejiang University: Science A* **6** (2005), no. 6, 497–501. DOI: 10.1007/BF02841755
[16] B. Mazur and K. Rubin, Organizing the arithmetic of elliptic curves, *Advances in Mathematics* **198** (2005), 504–546. DOI: 10.1016/j.aim.2005.05.024
[17] A. Sachdeva, A Study of Encryption Algorithms AES, DES and RSA for Security, *Network Web and Security* **13** (2013), no. 15, 1–9.
[18] D. Simian and C. Simian, On an approach for cubic Bézier interpolation, *Modelling and Development of Intelligent Systems*, Proceedings of the 2nd Int. Conf., Sibiu, Romania, (2011), 152–159.
[19] G. Stephanides and N. Constantinescu, The GN-authenticated key agreement, *Applied Mathematics and Computation*, Elsevier **170** (2005), 531–544. DOI: 10.1016/j.amc.2004.12.013
[20] C, Xavier and L. Reynald, Fast computation of elliptic curve isogenies in characteristic two. *Journal of the London mathematical society – second series* **104** (2021), no. 4, 1901–1929. DOI: 10.48550/arXiv.2003.06367

(Dana Simian) Research Center in Informatics and Information Technology, Faculty of Sciences, Lucian Blaga University of Sibiu, 5-7 Ion Raţiu Street, Sibiu, 550012, Romania
ORCID ID 0000-0002-5210-1810
E-mail address: dana.simian@ulbsibiu.ro

(Oana-Adriana Ţicleanu) Research Center in Informatics and Information Technology, Faculty of Sciences, Lucian Blaga University of Sibiu, 5-7 Ion Raţiu Street, Sibiu, 550012, Romania
ORCID ID 0000-0002-1304-7479
E-mail address: oana.ticleanu@ulbsibiu.ro

(Nicolae Constantinescu) Research Center in Informatics and Information Technology, Faculty of Sciences, Lucian Blaga University of Sibiu, 5-7 Ion Raţiu Street, Sibiu, 550012, Romania
ORCID ID 0000-0001-5136-4849
E-mail address: nicolae.constantinescu@ulbsibiu.ro