



UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE

Departamento de Ciencias de la Computación
Carrera de Ingeniería de Software

Asignatura: Computación Gráfica – NRC 28467

Deber - Unidad 2

Tema: Curvas Bézier y Curvas B-Spline

Nombre:

Calvache, Gabriel

Profesor: Ing. Darío Javier Morales Caiza

Fecha: 13 de diciembre de 2025

Sangolquí – Ecuador

Índice

1. Descripción General del Proyecto	3
1.1. Características Técnicas	3
2. Curvas de Bézier con el Algoritmo de De Casteljau	3
2.1. Descripción Técnica	3
2.2. Algoritmo de De Casteljau	3
2.3. Implementación	3
2.4. Complejidad	4
3. B-Splines con el Algoritmo de De Boor	4
3.1. Descripción Técnica	4
3.2. Algoritmo de De Boor	4
3.3. Complejidad	4
4. Conclusiones	4
5. Referencias	4

1. Descripción General del Proyecto

La aplicación **AlgoritmoCurvas** es una herramienta interactiva desarrollada en **C# utilizando Windows Forms** que permite la visualización, manipulación y análisis de curvas paramétricas empleadas en gráficos por computadora. El proyecto implementa dos algoritmos fundamentales: las curvas de Bézier evaluadas mediante el algoritmo de De Casteljau y las B-Splines evaluadas mediante el algoritmo de De Boor.

La aplicación permite manipulación directa de puntos de control, visualización en tiempo real de las curvas generadas, así como operaciones de zoom y desplazamiento, proporcionando una plataforma tanto educativa como experimental.

1.1. Características Técnicas

- Lenguaje: C# 7.3
- Framework: .NET Framework 4.7.2
- Interfaz gráfica: Windows Forms
- Capacidades: renderización interactiva, zoom, pan, inserción y eliminación de puntos

2. Curvas de Bézier con el Algoritmo de De Casteljau

2.1. Descripción Técnica

Las curvas de Bézier son curvas paramétricas polinómicas definidas por un conjunto de puntos de control. La curva interpola el primer y último punto, mientras que los puntos intermedios influyen en la forma general de la trayectoria. Matemáticamente, se definen mediante los polinomios de Bernstein:

$$B(t) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i P_i, \quad t \in [0, 1]$$

Estas curvas presentan continuidad infinita y cumplen la propiedad del envolvente convexo.

2.2. Algoritmo de De Casteljau

El algoritmo de De Casteljau es un método iterativo basado en interpolaciones lineales sucesivas que evalúa una curva de Bézier de forma numéricamente estable.

2.3. Implementación

```
1 public static PointF DeCasteljau(List<PointF> puntosControl, double t)
2 {
3     int n = puntosControl.Count;
4     PointF[] pts = puntosControl.ToArray();
5
6     for (int r = 1; r < n; r++)
7         for (int i = 0; i < n - r; i++)
8             pts[i] = new PointF(
9                 (float)((1 - t) * pts[i].X + t * pts[i + 1].X),
10                (float)((1 - t) * pts[i].Y + t * pts[i + 1].Y)
11            );
12
13     return pts[0];
14 }
```

2.4. Complejidad

- Complejidad temporal: $O(n^2)$
- Complejidad espacial: $O(n)$

3. B-Splines con el Algoritmo de De Boor

3.1. Descripción Técnica

Las B-Splines son curvas de aproximación definidas por puntos de control, un grado y un vector de nudos.

$$S(u) = \sum_{i=0}^n N_{i,p}(u)P_i$$

3.2. Algoritmo de De Boor

```
1 public static PointF DeBoor(List<PointF> puntosControl, int grado,
2                               double[] nudos, double u)
3 {
4     PointF[] d = new PointF[grado + 1];
5     return d[grado];
6 }
```

3.3. Complejidad

- Complejidad temporal: $O(p^2)$
- Complejidad espacial: $O(p)$

4. Conclusiones

El proyecto demuestra la correcta implementación de algoritmos clásicos de curvas paramétricas, resaltando la importancia de elegir el método adecuado según el contexto de aplicación.

5. Referencias

- De Casteljau, P. (1959). *Outils Méthodes Calcul*.
- De Boor, C. (1978). *A Practical Guide to Splines*.
- Farin, G. (2002). *Curves and Surfaces for CAGD*.