



**UNIVERSIDADE ESTADUAL DO CEARÁ
CENTRO DE CIÊNCIAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM MATEMÁTICA
MESTRADO PROFISSIONAL EM MATEMÁTICA EM REDE NACIONAL**

ARTUR TEIXEIRA PEREIRA

CURVAS DE BÉZIER

FORTALEZA – CEARÁ

2023

ARTUR TEIXEIRA PEREIRA

CURVAS DE BÉZIER

Dissertação apresentada ao Curso de Mestrado Profissional em Matemática do Programa de Pós-Graduação em Matemática do Centro de Ciências e Tecnologias da Universidade Estadual do Ceará, como requisito parcial à obtenção do título de mestre em Matemática. Área de Concentração: Matemática.

Orientador: Prof. Dr. Nicolas Alcântara de Andrade

FORTALEZA – CEARÁ

2023

Dados Internacionais de Catalogação na Publicação
Universidade Estadual do Ceará
Sistema de Bibliotecas

Pereira, Artur Teixeira.
Curvas de Bézier [recurso eletrônico] / Artur
Teixeira Pereira. - 2022.
70 f. : il.

Dissertação (MESTRADO PROFISSIONAL) -
Universidade Estadual do Ceará, Centro de Ciências
e Tecnologia, Curso de Mestrado Profissional Em
Matemática Rede Nacional - Profissional,
Fortaleza, 2022.

Orientação: Prof. Dr. Nicolas Alcântara de
Andrade.

1. curvas. 2. curvas de Bézier. 3. algoritmo
de casteljau.. I. Título.

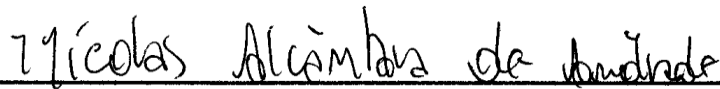
ARTUR TEIXEIRA PEREIRA

CURVAS DE BEZIER

Dissertação apresentada ao Curso de Mestrado Profissional em Matemática do Programa de Pós-Graduação em Matemática do Centro de Ciências e Tecnologias da Universidade Estadual do Ceará, como requisito parcial à obtenção do título de mestre em Matemática. Área de Concentração: Matemática.

Aprovada em: 22 de maio de 2023.

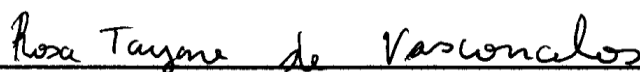
BANCA EXAMINADORA



Prof. Dr. Nicolas Alcântara de Andrade
Orientador - PROFMAT/UECE



Prof. Dr. Diego de Sousa Rodrigues
PROFMAT/UECE



Prof. Dra. Rosa Tayane de Vasconcelos

Instituto Federal de Educação, Ciência e Tecnologia do Ceará– IFCE

Dedicado a todos os que me ajudaram ao longo desta caminhada.

AGRADECIMENTOS

A Deus, por todas as bênçãos concedidas em minha vida.

A Maysa, minha esposa e companheira de todas as horas e meu filho Luiz Henrique, que tanto entenderam a necessidade da minha ausência em determinados momentos dessa minha jornada.

Aos meus pais e irmãos.

Aos meus amigos do mestrado: Danilo, Felipe, Mardney, Rafael Abreu, Rafael Mendonça, Sérgio, Tiago, Wellington Sampaio, que dividiram esse período de conquistas ao longo desse período. Aqui enalteço a importância de cada um nessa trajetória.

Aos professores do curso do Profmat: Leo Ivo e Thiago Caula, obrigado pelos ensinamentos.

Agradecimento especial ao Professor Nicolas Alcântara pelo privilégio de ter me aceito como orientando.

A todos que de forma direta ou indireta tiveram participação em mais um momento especial na minha vida.

“Somos quem podemos ser
Sonhos que podemos ter (E teremos!)”
(Engenheiros do Hawaii)

RESUMO

As curvas de Bézier são curvas polinomiais descritas a partir da interpolação de pontos representativos, os pontos de controle. Recebe esse nome em homenagem ao francês Pierre Bézier que a desenvolveu na década de 1960 para o design de automóveis. Pouco tempo antes o também francês Paul de Casteljau já havia desenvolvido algo semelhante, mas não publicado em razão de sigilos empresariais. As curvas de Bézier são bastante utilizadas na computação, e, especialmente na computação gráfica. Trata-se de curvas paramétricas geralmente definidas no intervalo paramétrico $[0,1]$. Sua construção requer uma revisão de tópicos como curvas planas, curvas paramétricas diferenciáveis, interpolação linear e o método interpolador de Lagrange. Outro elemento fundamental na construção e aplicação das curvas de Bézier são os polinômios de Bernstein que são definidos a partir da fórmula binomial. Algumas propriedades dos polinômios de Bernstein garantem uma visão mais abrangente das curvas de Bézier. Ao fim podemos observar algumas aplicações em softwares computacionais como resultado de sua importância.

Palavras-chave: curvas; curvas de Bézier; algoritmo de Casteljau.

ABSTRACT

Bézier curves are polynomial curves described from the interpolation of representative points, the control points. It is named after the Frenchman Pierre Bézier who developed it in the 1960s for car design. Shortly before, Paul de Casteljaou, also a Frenchman, had already developed something similar, but it was not published due to business secrecy. Bézier curves are widely used in computing, and especially in computer graphics. These are parametric curves generally defined in the parametric interval $[0,1]$. Its construction requires a review of topics such as plane curves, differentiable parametric curves, linear interpolation and the Lagrange interpolator method. Another fundamental element in the construction and application of Bézier curves are the Bernstein polynomials that are defined from the binomial formula. Some properties of Bernstein polynomials provide a more comprehensive view of Bézier curves. At the end we can observe some applications in computational software as a result of its importance.

Keywords: curves; Bézier curves; Casteljaou algorithm.

LISTA DE FIGURAS

Figura 1 –	Trajectoria de um ponto P	15
Figura 2 –	Curva relativa ao exemplo 2.1.1.....	17
Figura 3 –	Circunferência de centro na origem e raio igual a 1.....	19
Figura 4 –	Curva em <i>loop</i>	20
Figura 5 –	Exemplo Elipse.....	20
Figura 6 –	Espiral de Arquimedes.....	21
Figura 7 –	Modelos de Fontes de Letras.....	22
Figura 8 –	Modelo de Interpolação para $i = 4$	23
Figura 9 –	Polinômio de Bernstein de grau 1.....	30
Figura 10 –	Polinômio de Bernstein de grau 2.....	31
Figura 11 –	Polinômio de Bernstein de grau 3.....	31
Figura 12 –	Aplicação do polinômio de Bernstein.....	34
Figura 13 –	Pierre Etienne Bézier.....	37
Figura 14 –	Croqui do corpo em CAD pelo sistema UNISURF.....	38
Figura 15 –	Modelo TrueType.....	39
Figura 16 –	Paul de Faget de Casteljaú.....	39
Figura 17 –	Polígono de Controle.....	41
Figura 18 –	Primeira etapa do algoritmo de Casteljaú para a curva de Bézier quadrática.....	42
Figura 19 –	Segunda etapa do algoritmo de Casteljaú para a curva de Bézier quadrática.....	43
Figura 20 –	Terceira etapa do algoritmo de Casteljaú para a curva de Bézier quadrática.....	43
Figura 21 –	Bézier Quadrática.....	44
Figura 22 –	Primeira etapa do algoritmo de Casteljaú para a curva de Bézier cúbica.....	44
Figura 23 –	Segunda etapa do algoritmo de Casteljaú para a curva de Bézier cúbica.....	45
Figura 24 –	Terceira etapa do algoritmo de Casteljaú para a curva de Bézier cúbica.....	45
Figura 25 –	Última etapa do algoritmo de Casteljaú para a curva de	46

	Bézier cúbica.....	
Figura 26 –	Bézier Cúbica.....	46
Figura 27 –	Fecho Convexo.....	49
Figura 28 –	Relação entre forma das curvas e posição dos pontos de controle.....	49
Figura 29 –	Subdivisão Recursiva.....	51
Figura 30 –	Emendando Curvas Bézier.....	52
Figura 31 –	Comportamento de $C(t)$	54
Figura 32 –	Imagem bitmap ampliada.....	57
Figura 33 –	Imagem Matricial x Imagem Vetorial	57
Figura 34 –	Ícone da Ferramenta Bézier no CorelDraw.....	58
Figura 35 –	Aplicação da Ferramenta Bézier I.....	59
Figura 36 –	Aplicação da Ferramenta Bézier II.....	59
Figura 37 –	Ferramenta Bézier de curvatura.....	61
Figura 38 –	Ferramenta Bézier padrão.....	61
Figura 39 –	Ferramenta Traço Sensível ao Conteúdo.....	62
Figura 40 –	Construção de animações para jogos I.....	63
Figura 41 –	Construção de animações para jogos II.....	64

SUMÁRIO

1	INTRODUÇÃO.....	12
2	CURVAS.....	14
2.1	Introdução.....	14
2.2	Curva Parametrizada Diferenciável.....	17
3	INTERPOLAÇÃO LINEAR.....	22
3.1	Conceitos Iniciais.....	22
3.2	Forma de Lagrange.....	23
3.3	Interpolação Linear.....	25
3.4	Polinômios De Bernstein.....	26
4	CURVAS DE BÉZIER.....	37
4.1	Introdução.....	37
4.2	Principais Aspectos Históricos das Curvas de Bézier e sua relação com a computação.....	37
4.3	Definição de Curvas de Bézier via Algoritmo de Casteljaou e Polinômio de Bernstein.....	40
4.4	Propriedades das Curvas de Bézier.....	48
4.5	Propriedade Da Subdivisão Recursiva e Bézier por Partes.....	50
4.6	Derivadas de Curva de Bézier.....	52
5	APLICAÇÕES DA FERRAMENTA BÉZIER.....	56
5.1	Introdução.....	56
5.2	Aplicações da Ferramenta Bézier em Alguns Softwares e em Games.....	58
5.2.1	Corel Draw.....	58
5.2.2	Adobe Photoshop.....	60
5.2.3	Aplicação da Ferramenta Bézier em Jogos e Animações.....	63
6	PRODUTO EDUCACIONAL.....	65
6.1	Plano de aula.....	67
7	CONCLUSÃO.....	69
	REFERENCIAS.....	70

1 INTRODUÇÃO

O aprofundamento e a ampliação dos objetivos dos Parâmetros Curriculares Nacionais (PCNs) pela Base Nacional Comum Curricular (BNCC) trouxeram mudanças que ressaltam a importância da Matemática na sociedade, reorganizando-a e criando novos temas para ampliação do processo ensino-aprendizagem.

A BNCC busca focar no que o aluno precisa desenvolver para que o conhecimento matemático possa servir como elemento transformador de sua realidade, tendo a questão da pesquisa estruturada em etapa como elemento primordial na formação do cidadão crítico.

Baseado nessas condições, tomando a Álgebra e a Geometria como unidade temática, podemos focar no que a BNCC chama de pensamento algébrico-geométrico. As ideias matemáticas vinculadas a esse pensamento orientam a utilização de modelos matemáticos na compreensão, representação e análise de situações cotidianas.

Nessa perspectiva alguns temas antes ensinados foram substituídos ou adaptados. Podemos citar como exemplo os números complexos. Em contrapartida, destinou-se um espaço mais significativo ao uso de aplicações computacionais. Com essa inserção pode-se ampliar a visão de estruturas matemáticas antes difíceis de se representar apenas em lousas e papéis.

Associado a esse pensamento podemos incluir inúmeras condições favoráveis no processo de ensino. Para esse trabalho trataremos uma análise das curvas ou regiões utilizadas para interpolar suavemente uma outra curva passando por duas extremidades fixas. Esse processo faz com que seja possível transladar, ampliar ou diminuir figuras ou imagens geradas por esses pontos.

Inserimos nesse contexto as Curvas de Bézier, a fim de lidar com um estudo vetorial de imagens e podermos verificar suas vantagens de armazenamento e manipulação. As curvas de Bézier são curvas polinomiais denotadas pela interpolação linear de alguns pontos representativos. Esses pontos são denominados de pontos de controle.

Essas curvas são base para a utilização em diversos programas computacionais como por exemplo, Illustrator, Freehand, Fireworks, GIMP,

Photoshop, Processing, Inkscape, Krita e CorelDraw, e formatos de imagem vetorial como o SVG.

Para que possamos exibir, caracterizar e aplicar as Curvas de Bézier, inicialmente precisaremos fazer uma revisão sobre curvas, essa revisão será apresentada em nosso segundo capítulo. Nela apresentaremos uma visão geral sobre as curvas, suas definições e o conceito de curvas parametrizadas. As curvas parametrizadas proporcionam uma forma de representar funções, aqui algumas vezes referidas apenas como aplicações, com entradas unidimensionais e com resultados multidimensionais. Esse modelo chamamos de função paramétrica e a entrada chamamos de parâmetro.

No capítulo seguinte será exposto a ideia de interpolação e do polinômio de Bernstein, além de suas utilizações e características. No quarto capítulo estudaremos as Curvas de Bézier e o Algoritmo de Casteljau, exibiremos ao longo do capítulo o contexto histórico das curvas de Bézier, sua definição usando princípios de parametrização, via Algoritmo de Casteljau e polinômio de Bernstein, sua derivada, e aplicações. No quinto capítulo apresentamos algumas aplicações da ferramenta de Bézier em softwares de computadores. Por fim, no último capítulo propomos uma aplicação pedagógica para o tema de forma que se possa concretizar a relevância e aplicabilidade das curvas de Bézier.

Ao longo do trabalho será apresentado alguns exemplos que facilitarão no entendimento, apresentamos também uma série de figuras com gráficos de curvas que propiciarão uma visão geométrica da construção e aplicabilidades das curvas de Bézier.

2 CURVAS

2.1 Introdução

As curvas são objetos importantes em diversas áreas de estudos. Um exemplo muito simples é o gráfico de uma função contínua. Quando somos levados à dar um exemplo de uma curva, temos uma ideia, pelo menos intuitiva, de mencionar uma reta, por exemplo $y - x = 1$, ou uma circunferência, por exemplo $x^2 + y^2 = 1$. Os exemplos citados acima dão direção ao entendimento de que uma curva pode ter várias representações de escritas. Segundo Alencar, Santos e Neto (2020 p.3)

No contexto de geometria diferencial, em vez de considerarmos curvas definidas por equações, [...] uma curva C deve descrever a trajetória contínua do movimento de uma partícula sobre o plano. A vantagem dessa abordagem é que ela poderá ser facilmente formalizada e conterá várias informações sobre como o ponto percorre o conjunto C , e o sentido que o ponto “anda” sobre C permite definir sua velocidade, sua aceleração etc.

Definição 2.1.1 Curva, caminho ou trajetória, no \mathbb{R}^n , é uma aplicação contínua $\gamma: I \rightarrow \mathbb{R}^n$, definida num intervalo $I \subset \mathbb{R}$.

Definição 2.1.2 A imagem $\gamma(I) \subset \mathbb{R}^n$ é chamada de traço da curva γ .

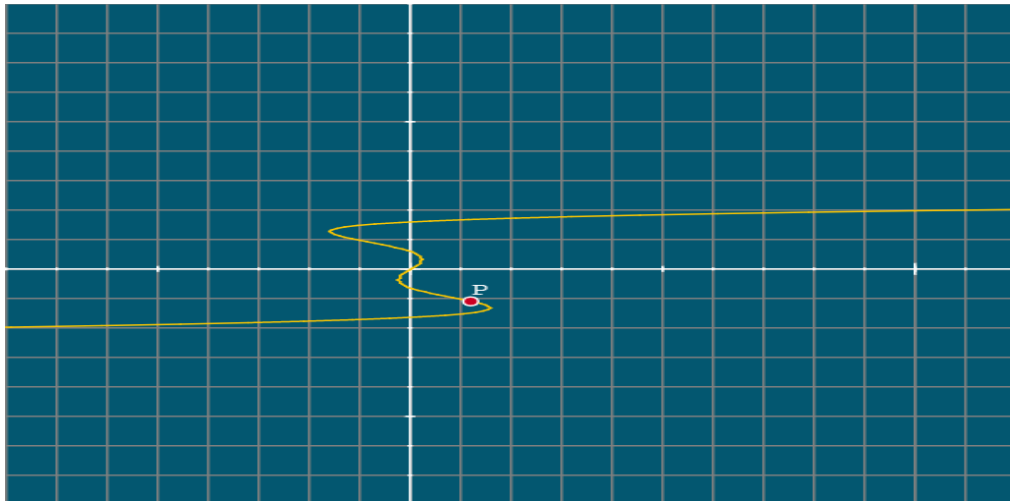
Observação 2.1.1 A aplicação γ em \mathbb{R}^2 definida por

$$\gamma(t) = (x(t), y(t)),$$

é contínua, se cada função coordenada $x, y: I \rightarrow \mathbb{R}$ é uma função contínua.

Na figura abaixo podemos interpretar o ponto P como se estivesse se movendo sobre a curva. Para isso, podemos “arrastar” este ponto sobre a curva. Observe que a curva traçada em destaque é o gráfico da trajetória seguida pelo ponto P.

Figura 1 – Trajetória de um ponto P



Fonte: elaborado pelo autor.

Imaginemos, agora, que uma partícula se move ao longo da curva \mathcal{C} da imagem anterior. Como podemos ver, a curva \mathcal{C} não é um gráfico de uma função $y = f(x)$ pois, retas verticais interceptam essa curva em mais de um ponto. Desta forma, não é possível descrever a curva \mathcal{C} usando uma equação $y = f(x)$. Mas como as coordenadas x e y da partícula são funções do tempo, a posição da partícula pode ser determinada em cada momento t por duas funções coordenadas do tipo $x = f(t)$ e $y = g(t)$. Cada valor de t designa um ponto (x, y) que pode ser marcado em um plano coordenado. Quando t muda, o ponto $(x, y) = (f(t), g(t))$ também muda e traça a curva \mathcal{C} .

Como resultado, em cada instante de tempo t , podemos determinar a posição da partícula P usando um par de equações da forma $x = f(t)$ e $y = g(t)$. Esta é uma maneira muito fácil de descrever a trajetória de uma partícula, pois para cada momento de tempo t considerado, obtemos as coordenadas exatas x e y que descrevem a posição do ponto P . O vetor $\vec{v}(t) = OP$, com extremidades inicial em $O = (0,0)$ e final no ponto P , indica a posição da partícula em cada instante de tempo t , e por isso é conhecido como a posição vetorial do ponto P .

As equações $x = f(t)$ e $y = g(t)$, que determinam a posição do ponto P à medida que ele se move ao longo da curva \mathcal{C} em cada instante de tempo t , são conhecidas como equações paramétricas. A variável t é conhecida como parâmetro. De um modo geral, se uma partícula se desloca sobre uma curva \mathcal{C} do plano de um

ponto A (x_0, y_0) até um ponto B (x_1, y_1) , é possível determinar em cada instante t por duas funções do tempo $x(t)$ e $y(t)$ ditas funções paramétricas. A função que a cada t associa o vetor posição $\vec{v}(t)$ é denominada uma parametrização da curva \mathcal{C} .

Para sintetizarmos essas condições tomamos como exemplo, as equações paramétricas de uma reta. Dada uma reta s que passa pelos pontos M e N e seja P um ponto pertencente a s de tal forma que o vetor \overrightarrow{MP} é múltiplo de \overrightarrow{MN} , logo $\overrightarrow{MP} = t\overrightarrow{MN}$. Escrevemos a equação paramétrica de s da seguinte forma:

$$s: P = M + t(N - M), t \in \mathbb{R}$$

Suponha por fim que $M = (a, b)$, $N = (c, d)$ e $P = (x, y)$ são pontos de um sistema de coordenadas então $P \in s$ se e somente se,

$$\begin{aligned} (x, y) &= (a, b) + t(c - a, d - b), t \in \mathbb{R} \\ &= (a + t(c - a), b + t(d - b)), t \in \mathbb{R} \\ \begin{cases} x = a + t(c - a) \\ y = b + t(d - b) \end{cases}, t \in \mathbb{R} \end{aligned}$$

Exemplo 2.1.1 Vamos identificar e ilustrar a curva cujas equações paramétricas são dadas por $x = t^2 - 2t$ e $y = t + 1$. Como solução verifiquemos que para cada valor que damos a t encontramos um par (x, y) que nos fornece um ponto da curva. Na tabela abaixo indicamos alguns valores de t e os respectivos valores de x e de y .

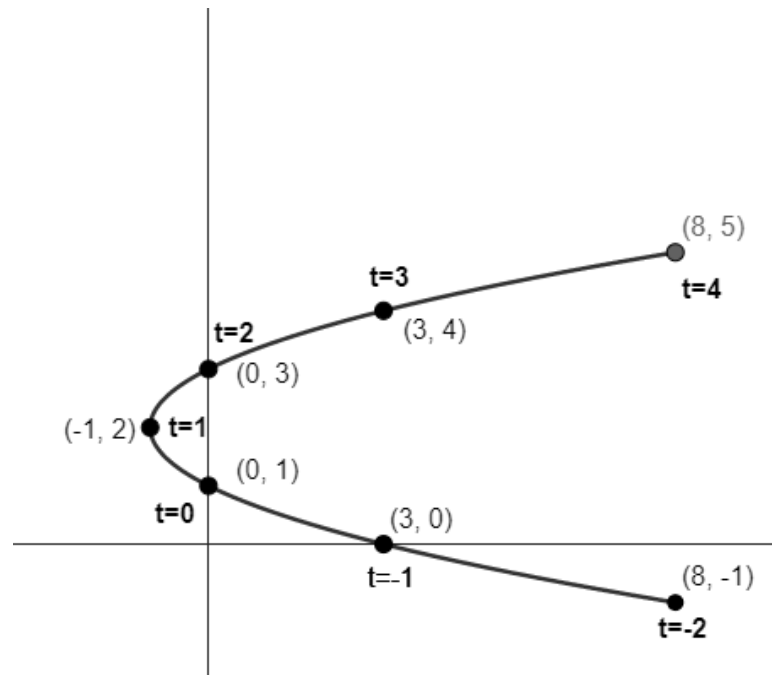
Tabela 1 – Relação entre o parâmetro t e os valores de x e y .

t	x	y
-2	8	-1
-1	3	0
0	0	1
1	-1	2
2	0	3
3	3	4
4	8	5

Fonte: elaborado pelo autor.

Na figura abaixo reunimos todos os pontos (x, y) determinados a partir da escolha numérica do parâmetro t

Figura 2 – Curva relativa ao exemplo 2.1.1.



Fonte: elaborado pelo autor, no Geogebra.

2.2 Curva parametrizada diferenciável

Para Carmo (2005), uma função de uma variável real é diferenciável ou suave se, em todos os pontos, possui derivadas de todas as ordens.

Definição 2.2.1 Uma curva diferenciável parametrizada em \mathbb{R}^n é uma aplicação diferenciável $\gamma: I \rightarrow \mathbb{R}^n$ definida num intervalo $I \subset \mathbb{R}$.

Supondo $\gamma(t) = (f_1(t), f_2(t), \dots, f_n(t))$, com $I \subset \mathbb{R}$, cada uma das funções $f_i(t): I \rightarrow \mathbb{R}$ são chamadas funções coordenadas de $\gamma(t)$.

Observação 2.2.1 γ ser diferenciável equivale à $f_1(t), f_2(t), \dots, f_n(t)$ serem diferenciáveis.

Observação 2.2.2 Curva parametrizada é uma aplicação diferenciável, já traço é um subconjunto de \mathbb{R}^n .

Observação 2.2.3 (Caso particular de curva) Os gráficos de funções são casos particulares de curvas planas. Seja $f: I \rightarrow \mathbb{R}$ uma função contínua. O conjunto

$$G = \{(x, y) \in I \times \mathbb{R}; y = f(x)\} \subset \mathbb{R}^2$$

é o gráfico de f . O conjunto G pode ser parametrizado pela curva contínua $\gamma: I \rightarrow \mathbb{R}^2$ dada por $\gamma(t) = (t, f(t))$.

Definição 2.2.2 Vetor tangente ou vetor velocidade de $\gamma(t)$ em t é dado por $\gamma'(t) = (f'_1(t), f'_2(t), \dots, f'_n(t))$, onde cada $f'_i(t)$ representa a derivada de f_i no ponto t , para $i = 1, \dots, n$.

Definição 2.2.3 Uma aplicação $\gamma: I \rightarrow \mathbb{R}^n$ é dita curva de classe C^k se cada uma das funções coordenadas na expressão $\gamma(t) = (f_1(t), f_2(t), \dots, f_n(t))$ tem derivadas contínuas até a ordem k . Se γ é apenas contínua, diz-se que γ é de classe C^0 .

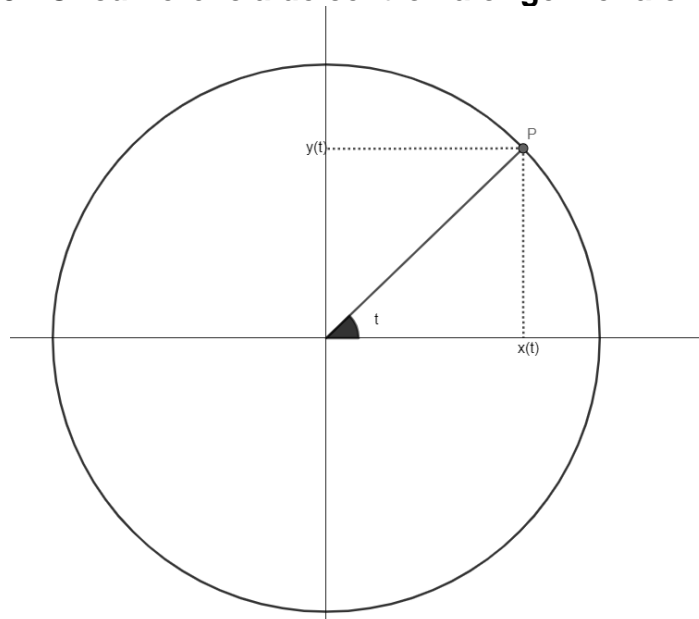
A seguir apresentaremos exemplos de aplicações de curva parametrizada diferenciável e seu respectivo traço.

Exemplo 2.2.1 A curva constante de aplicação $\gamma: \mathbb{R} \rightarrow \mathbb{R}^2$ dada por $\gamma(t) = (a, b)$ é uma curva parametrizada diferenciável cujo traço é o ponto (a, b) .

Exemplo 2.2.2 A reta, como vimos anteriormente, com aplicação $\gamma: \mathbb{R} \rightarrow \mathbb{R}^2$ dada por $\gamma(t) = (x_0 + at, y_0 + bt)$ onde $a^2 + b^2 \neq 0$, é uma curva parametrizada diferenciável cujo traço é uma linha reta passando pelo ponto (x_0, y_0) e paralela ao vetor de coordenadas (a, b) .

Exemplo 2.2.3 Para a Circunferência \mathcal{C} , temos a aplicação $\gamma: [0, 2\pi]$ definida por $\gamma(t) = (\cos(t), \sin(t))$, é uma curva parametrizada diferenciável cujo traço é uma circunferência de centro na origem e raio igual a 1. Caso a circunferência não esteja centrada na origem, mas em (x_0, y_0) e tenha raio $r > 0$ qualquer definimos as equações paramétricas como $\gamma(t) = (x_0 + r\cos(t), y_0 + r\sin(t))$ onde (x_0, y_0) é o centro e r é o raio.

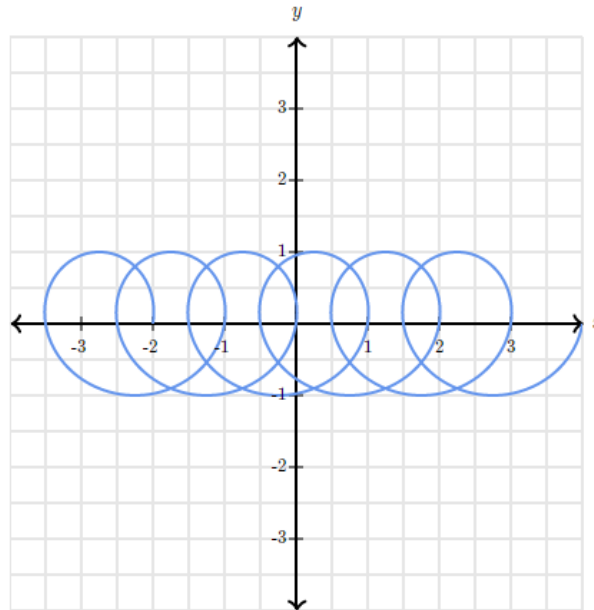
Figura 3 - Circunferência de centro na origem e raio igual a 1.



Fonte: elaborado pelo autor, no Geogebra.

Podemos associar a ideia do exemplo 2.2.3 à parametrização de uma curva em *loop* como representada na figura 4. A parametrização desta curva requer uma noção de como será seu desenho. Intuitivamente pode-se relacioná-la a um círculo desenhado no sentido anti-horário enquanto a mão é deslocada para a direita em velocidade constante. Como vimos a parametrização do círculo é da forma $(\cos(t), \sin(t))$. Iniciamos em $(1,0)$ e círculo de raio unitário, percebemos que a curva em análise se inicia em $(-2,0)$, assim convêm-se o ajuste do deslocamento usando o os valores de $x = -3$. A nova parametrização fica $(\cos(t) - 3, \sin(t))$. O deslocamento será constante ao longo do eixo x , para formalizar, definiremos que esse movimento ocorre c vezes. Assim a fórmula fica $(\cos(t) - 3 + ct, \sin(t))$. Nossa função completa um *loop* quando t varia de 0 a 2π , com isso verificamos que $2\pi c = 1$, logo $c = \frac{1}{2\pi}$. Isso significa que a aplicação será da forma $(\cos(t) - 3 + \frac{1}{2\pi}t, \sin(t))$. Bastando finalizar apenas com a informação de quantas voltas a curva dará.

Figura 4 – Curva em Loop

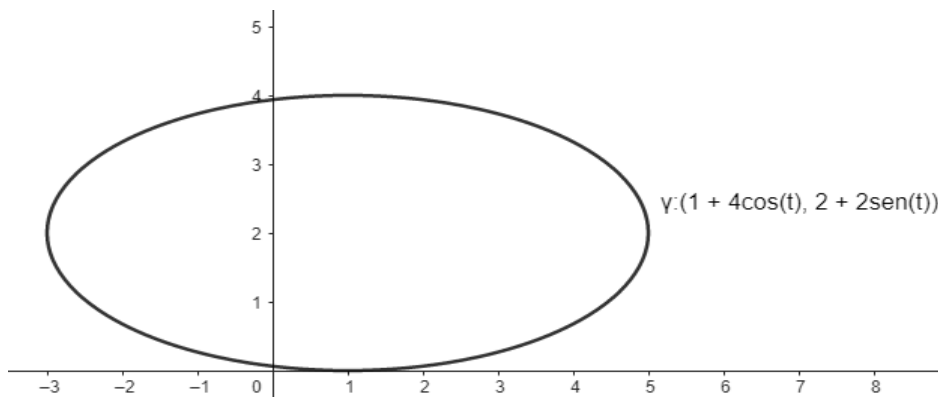


Fonte: elaborado pelo autor.

As curvas como as circunferências, ilustradas anteriormente, são um exemplo de cônicas, outras curvas que se enquadram na mesma categoria são as elipses, as hipérboles e as parábolas, cujas equações paramétricas serão representadas nos próximos exemplos.

Exemplo 2.2.4 A aplicação $\gamma: \mathbb{R} \rightarrow \mathbb{R}^2$ dada por $\gamma(t) = (x(t), y(t)) = (x_0 + acost, y_0 + bsen(t))$ é uma parametrização da elipse de equação $\mathcal{E}: \frac{(x - x_0)^2}{a^2} + \frac{(y - y_0)^2}{b^2} = 1$.

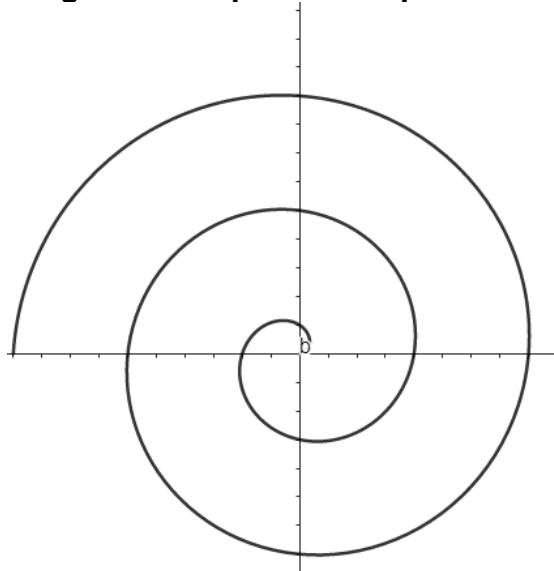
Figura 5 – Exemplo de Elipse.



Fonte: elaborado pelo autor, no Geogebra.

Exemplo 2.2.5 A curva $\gamma: [0, 2\pi] \rightarrow \mathbb{R}^2$ de equação $\gamma(t) = (at\cos(t), at\sin(t))$, $a > 0$, $0 \leq t \leq 2\pi$, é uma curva parametrizada diferenciável cujo traço recebe o nome de espiral de Arquimedes sendo definida em $D = [0, +\infty)$.

Figura 6 – Espiral de Arquimedes



Fonte: elaborado pelo autor, no Geogebra

É importante observar que determinadas curvas apresentam propriedades que não se caracterizam como curva parametrizável diferenciável, pois podem existir pontos onde a curva não é diferenciável como exemplo a aplicação $\gamma: \mathbb{R} \rightarrow \mathbb{R}^2$ dada por $\gamma(t) = (t, |t|)$ não é uma curva parametrizada diferenciável, já que $|t|$ não é diferenciável em $t = 0$. Porém a restrição de γ , a qualquer intervalo que não contem o ponto $t = 0$, é uma curva parametrizada diferenciável.

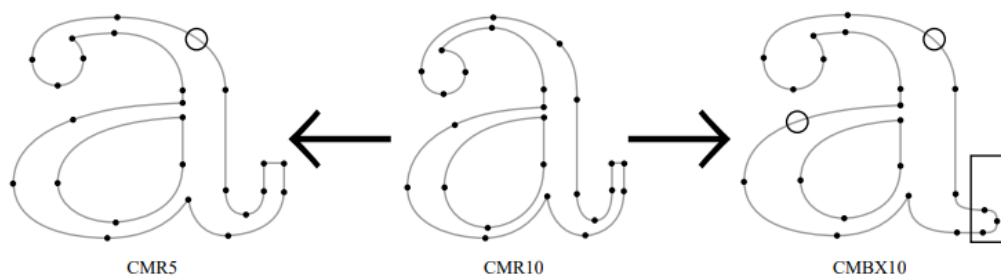
A análise das curvas, em especial as planas, apresentam vários desafios, em geral pelas características de cada uma. Um fato instigador é encontrar uma curva que se ajuste a uma série de pontos ou a determinadas propriedades específicas e parâmetros. Esses ajustes podem envolver a ideia de interpolação, onde é necessária uma regulagem exata aos dados ou ainda apenas pela suavização da aproximação dos dados.

3 INTERPOLAÇÃO LINEAR

3.1 Conceitos Iniciais

Adaptar uma curva por pontos específicos em um determinado plano, na análise de dados experimentais, frequentemente encontram problemas nas relações entre as variáveis. Essa análise pode ser bem exemplificada pela elaboração de fontes de letras em computadores e impressoras. Um exemplo pode ser visto na Figura 7: as letras são retiradas das versões PostScript das fontes CMR5, CMR10, CMBX10.

Figura 7 – Modelos de Fontes de Letras



Fonte: elaborado pelo autor.

Percebe-se que associar esses pontos a uma função pode ser uma tarefa árdua. Desse modo é conveniente converter essas funções em polinômios que a represente, uma vez que o trabalho com polinômios tende a apresentar relativa facilidade, pois existem vários métodos que nos auxiliam a determinar suas raízes, derivadas e integrais.

Para (Ruggiero, 1997), a “interpolação nos ajuda no sentido de que podemos aproximar uma função $f(x)$ por outra função $g(x)$, escolhida entre uma classe de funções definida a priori de modo que satisfaça algumas propriedades”.

A ideia de se usar polinômios para poder representar funções de complexidade maior se dá pelo fato de que pelo Teorema de Weierstrass toda função contínua $f: [a, b] \rightarrow \mathbb{R}$ pode ser uniformemente aproximada por polinômios.

3.2 Forma de Lagrange

Consideremos um conjunto de pontos x_0, x_1, \dots, x_n (chamados nós da interpolação) associados, respectivamente, aos valores de uma função $f(x)$ aplicados nesses pontos: $f(x_0), f(x_1), \dots, f(x_n)$ o que estamos nos propondo a determinar é o polinômio $p(x)$ de tal modo que:

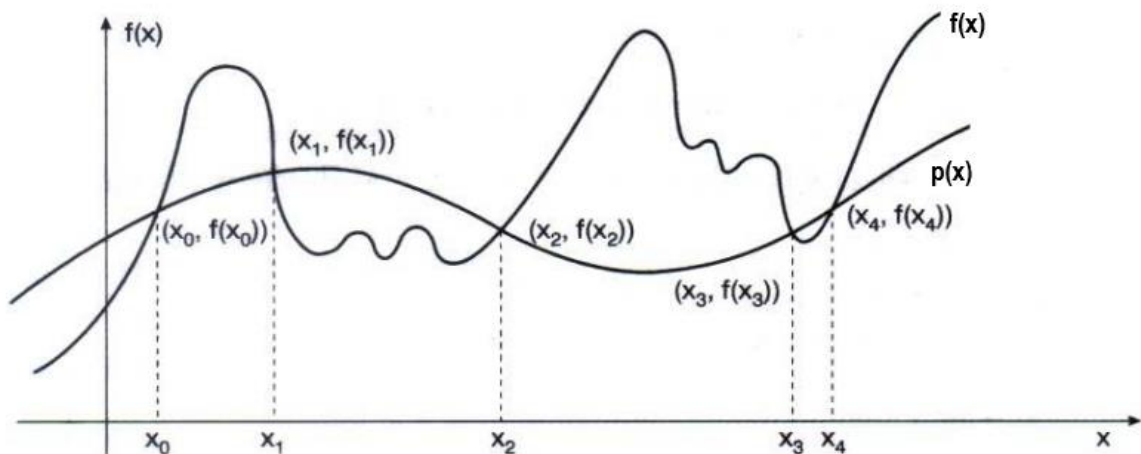
$$\begin{cases} p(x_0) = f(x_0) \\ p(x_1) = f(x_1) \\ p(x_2) = f(x_2) \\ \vdots \end{cases}$$

Assim, generalizando as igualdades acima teremos:

$$p(x_i) = f(x_i); \quad i = 0, \dots, n$$

Definição 3.2.1 Uma curva que passa por um conjunto de pontos no plano é dita curva interpoladora desses pontos.

Figura 8 – Modelo de Interpolação para $i = 4$



Fonte: (RUGGIERO e LOPES, 1997, com adaptações)

Tomando $p(x) = a_m x^m + a_{m-1} x^{m-1} + \dots + a_1 x + a_0$, obteremos o sistema abaixo

$$\begin{cases} f(x_0) = a_m x_0^m + a_{m-1} x_0^{m-1} + \dots + a_1 x_0 + a_0 \\ \vdots \\ f(x_n) = a_m x_n^m + a_{m-1} x_n^{m-1} + \dots + a_1 x_n + a_0 \end{cases} \quad (1)$$

Para que o sistema acima seja possível e determinado devemos ter ao menos a condição necessária de $m = n$. Dessa forma o sistema (1) pode ser reescrito na forma matricial abaixo

$$\begin{bmatrix} x_0^m & x_0^{m-1} & \dots & x_0 & 1 \\ x_1^m & x_1^{m-1} & \dots & x_1 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_n^m & x_n^{m-1} & \dots & x_n & 1 \end{bmatrix} \begin{bmatrix} a_m \\ a_{m-1} \\ \vdots \\ a_0 \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_n) \end{bmatrix}$$

A existência e unicidade do polinômio interpolador consiste em assegurar a solução do sistema com a matriz de Vandermonde acima, para todos x_0, \dots, x_n distintos.

O método descrito a seguir foi desenvolvido por Joseph-Louis Lagrange, (1736 – 1813), matemático e astrônomo italiano da era do iluminismo. Inicialmente tomemos x_0, x_1, \dots, x_n , $(n + 1)$ pontos distintos e $f(x_i) = y_i$; $i = 0, \dots, n$.

Seja $p_n(x)$ o polinômio com grau menor ou igual a n que interpola f em x_0, \dots, x_n . Representaremos $p_n(x)$ da forma $p_n(x) = y_0L_0(x) + y_1L_1(x) + \dots + y_nL_n(x)$, onde cada $L_k(x)$, $k = 0, 1, \dots, n$, são polinômios de grau n . Queremos que a condição $p_n(x_i) = y_i$ seja satisfeita para cada i , ou seja:

$$p_n(x_i) = y_0L_0(x_i) + y_1L_1(x_i) + \dots + y_nL_n(x_i) = y_i.$$

Definiremos $L_k(x)$ por

$$L_k(x) = \frac{(x - x_0)(x - x_1) \dots (x - x_{k-1})(x - x_{k+1}) \dots (x - x_n)}{(x_k - x_0)(x_k - x_1) \dots (x_k - x_{k-1})(x_k - x_{k+1}) \dots (x_k - x_n)}.$$

Observe que com essas definições valem as igualdades $L_k(x_k) = 1$ e $L_k(x_i) = 0$ se $i \neq k$. Como o numerador da expressão acima é formado pelo produto de n fatores do tipo $(x - x_i)$ com $i = 0, \dots, n, i \neq k$, então o polinômio $L_k(x)$ tem grau n , assim $p_n(x)$ tem grau menor ou igual a n .

Definição 3.2.2 Dada uma função $f: I \rightarrow \mathbb{R}$, $x_0, x_1, \dots, x_n \in I$. O polinômio interpolador de Lagrange é definido como:

$$p_n(x) = \sum_{k=0}^n y_k L_k(x)$$

Onde $L_k(x) = \frac{\prod_{j=0, j \neq k}^n (x - x_j)}{\prod_{j=0, j \neq k}^n (x_k - x_j)}$ e $y_k = f(x_k)$.

Exemplo 3.2.1 Determinar o polinômio interpolador de Lagrange para a função $f(x) = \frac{1}{x}$, usando os pontos $x_0 = 2$; $x_1 = 2,5$ e $x_2 = 4$.

Solução: Primeiro passo é determinar $L_0(x)$, $L_1(x)$ e $L_2(x)$. Usando a definição anterior teremos

$$\begin{aligned} L_0(x) &= \frac{(x - 2,5)(x - 4)}{(2 - 2,5)(2 - 4)} = (x - 2,5)(x - 4) \\ L_1(x) &= \frac{(x - 2)(x - 4)}{(2,5 - 2)(2,5 - 4)} = -\frac{(x - 2)(x - 4)}{0,75} \\ L_2(x) &= \frac{(x - 2)(x - 2,5)}{(4 - 2)(4 - 2,5)} = \frac{(x - 2)(x - 2,5)}{3} \end{aligned}$$

Como $f(x_0) = f(2) = 0,5$; $f(x_1) = f(2,5) = 0,4$; $f(x_2) = f(4) = 0,25$ temos que

$$P(x) = 0,5 [(x - 2,5)(x - 4)] + 0,4 \left[-\frac{(x - 2)(x - 4)}{0,75} \right] + 0,5 \left[\frac{(x - 2)(x - 2,5)}{3} \right] \text{ logo,}$$

$P(x) = 0,05x^2 - 0,425x + 1,15$. Usando o polinômio $P(x)$ calculado, podemos estimar o valor da função $f(x) = \frac{1}{x}$ em um ponto qualquer com um erro estimado, quanto mais nós tiver a interpolação melhor a aproximação.

3.3 Interpolação Linear

Tomaremos o caso $n = 1$ para o resultado anterior. Teremos,

$$p_1(x) = y_0 L_0(x) + y_1 L_1(x), \text{ como } L_0(x) = \frac{(x - x_1)}{(x_0 - x_1)} \text{ e } L_1(x) = \frac{(x - x_0)}{(x_1 - x_0)},$$

fazendo as substituições resulta em

$$\begin{aligned} p_1(x) &= y_0 \frac{(x - x_1)}{(x_0 - x_1)} + y_1 \frac{(x - x_0)}{(x_1 - x_0)} \\ p_1(x) &= y_0 \frac{(x_1 - x)}{(x_1 - x_0)} + y_1 \frac{(x - x_0)}{(x_1 - x_0)} \end{aligned}$$

Esse modelo é chamado de interpolação linear.

Teorema 3.3.1 Dados quaisquer $n + 1$ pontos no plano xy que tem coordenadas x distintas, existe um único polinômio de grau n ou inferior cujo gráfico passa por esses pontos.

Demonstração: Inicia-se pela unicidade.

Supondo que existem mais de um polinômio interpolador, sem perda de generalidade, suponhamos dois, p e q de grau menor ou igual a n , então teremos $p(x) - q(x)$ com grau menor ou igual a n e $n + 1$ raízes, já que se são polinômios interpoladores obedecem a condição:

$$p(x_i) = f(x_i) = q(x_i) \text{ para } i = 0, \dots, n.$$

Portanto, como o polinômio $p(x) - q(x)$ tem $n + 1$ raízes e grau menor ou igual a n terá que ser nulo e com isso $p = q$.

A demonstração continua com a prova da existência. A existência desse polinômio se prova a partir da construção polinômio interpolador de Lagrange.

■

3.4 Polinômios De Bernstein

Os estudos de Sergei Natanovich Bernstein (1880 – 1968), matemático soviético que se destacou com seus trabalhos centrado-se na aproximação de funções e teoria da probabilidade, nos ajudarão a definir os polinômios que levam seu nome e as propriedades mais relevantes. Esses polinômios têm grande importância na construção das Curvas de Bézier que são os objetivos de estudo deste trabalho.

Definição 3.4.1 Um polinômio de Bernstein (B_i^n) é definido por

$$B_i^n(t) = \binom{n}{i} t^i (1 - t)^{n-i} \text{ com } i = 0, \dots, n$$

Notemos que o polinômio descrito na definição 3.4.1, é uma parcela do desenvolvimento binomial

$$1 = (t + (1 - t))^n = \sum_{i=0}^n \binom{n}{i} t^i (1 - t)^{n-i}$$

Uma condição importante é uma definição recursiva dos polinômios de Bernstein. Os polinômios de Bernstein de grau n podem ser definidos somando dois polinômios de Bernstein como representados na proposição abaixo usando a relação de Stiffel.

Proposição 3.4.1 O i -ésimo polinômio de grau n -ésimo de Bernstein pode ser escrito como

$$\mathcal{B}_i^n(t) = (1 - t) \mathcal{B}_i^{n-1}(t) + t \mathcal{B}_{i-1}^{n-1}(t)$$

Demonstração: Para mostrar isso, precisamos apenas usar a definição dos polinômios de Bernstein e alguma álgebra simples:

$$\begin{aligned} (1 - t) \mathcal{B}_i^{n-1}(t) + t \mathcal{B}_{i-1}^{n-1}(t) &= (1 - t) \binom{n-1}{i} (1 - t)^{n-1-i} + t \binom{n-1}{i-1} t^{i-1} (1 - t)^{n-1-(i-1)} \\ &= \binom{n-1}{i} t^i (1 - t)^{n-i} + \binom{n-1}{i-1} t^i (1 - t)^{n-i} \\ &= [\binom{n-1}{i} + \binom{n-1}{i-1}] t^i (1 - t)^{n-i} \\ &= \binom{n}{i} t^i (1 - t)^{n-i} \text{ (por Stiffel)} \\ &= \mathcal{B}_i^n(t) \quad \blacksquare \end{aligned}$$

Definição 3.4.2 Dizemos que um conjunto de funções $f_i(t)$ particiona a unidade se somam um para todos os valores de t .

Proposição 3.4.2 Os $n + 1$ polinômios Bernstein de grau n formam uma partição de unidade.

Demonstração: Para mostrar que isso é verdade, é mais fácil primeiro mostrar um fato ligeiramente diferente: para cada n , a soma dos $n + 1$ polinômios de Bernstein de grau n é igual à soma dos n polinômios de Bernstein de grau $n - 1$. Isto é,

$$\sum_{i=0}^n \mathcal{B}_i^n(t) = \sum_{i=0}^{n-1} \mathcal{B}_i^{n-1}(t)$$

Este cálculo é direto, usando a definição recursiva e reorganizando as somas de forma convenientes:

$$\begin{aligned}
 \sum_{i=0}^n \mathcal{B}_i^n(t) &= \sum_{i=0}^n [(1-t)\mathcal{B}_i^{n-1}(t) + t\mathcal{B}_{i-1}^{n-1}(t)] \\
 &= (1-t) \left[\sum_{i=0}^{n-1} \mathcal{B}_i^{n-1}(t) + \mathcal{B}_n^{n-1}(t) \right] + t \left[\sum_{i=1}^n \mathcal{B}_{i-1}^{n-1}(t) + \mathcal{B}_{-1}^{n-1}(t) \right] \\
 &= (1-t) \sum_{i=0}^{n-1} \mathcal{B}_i^{n-1}(t) + t \sum_{i=1}^n \mathcal{B}_{i-1}^{n-1}(t) \\
 &= (1-t) \sum_{i=0}^{n-1} \mathcal{B}_i^{n-1}(t) + t \sum_{i=0}^{n-1} \mathcal{B}_i^{n-1}(t) \\
 &= \sum_{i=0}^{n-1} \mathcal{B}_i^{n-1}(t)
 \end{aligned}$$

Utilizamos por convenção $\mathcal{B}_n^{n-1}(t) = \mathcal{B}_{-1}^{n-1}(t) \equiv 0$. Estabelecida essa igualdade, podemos escrever

$$\sum_{i=0}^n \mathcal{B}_i^n(t) = \sum_{i=0}^{n-1} \mathcal{B}_i^{n-1}(t) = \sum_{i=0}^{n-2} \mathcal{B}_i^{n-2}(t) = \dots = \sum_{i=0}^1 \mathcal{B}_i^1(t) = (1-t) + t = 1 \quad \blacksquare$$

Uma observação em relação a partição da unidade é sua importância quando se utiliza polinômios de Bernstein em geometria, modelagem e computação gráfica. Em particular, para qualquer conjunto de pontos P_0, P_1, \dots, P_n , no espaço, e para qualquer t , a expressão

$$P(t) = P_0\mathcal{B}_0^n(t) + P_1\mathcal{B}_1^n(t) + \dots + P_n\mathcal{B}_n^n(t)$$

é uma combinação linear do conjunto de pontos P_0, P_1, \dots, P_n e se $0 \leq t \leq 1$, é uma combinação convexa dos pontos.

Proposição 3.4.3 Os polinômios de Bernstein são positivos em $(0,1)$.

Demonstração: De fato, visto que $t \in (0,1)$, temos $t^i(1-t)^{i-k} \geq 0$, portanto da definição temos,

$$\mathcal{B}_i^n(t) = \binom{n}{i} t^i (1-t)^{i-k} \geq 0, \forall n, i \in \mathbb{N}, \quad i \neq n \quad \blacksquare$$

Daqui em diante sempre consideraremos, a não ser menção contrária, que os polinômios de Bernstein possuem o domínio no intervalo $(0,1)$.

Apresentamos agora alguns exemplos de polinômios de Bernstein:

Polinômios de Bernstein de grau 1: Usando a definição temos,

$$\mathcal{B}_0^1 = \binom{1}{0} t^0 (1-t)^1 = 1 \cdot 1 \cdot (1-t) = (1-t)$$

$$\mathcal{B}_1^1 = \binom{1}{1} t^1 (1-t)^0 = 1 \cdot t \cdot 1 = t$$

Polinômios de Bernstein de grau 2: Usando a definição temos,

$$\mathcal{B}_0^2 = \binom{2}{0} t^0 (1-t)^2 = 1 \cdot 1 \cdot (1-t)^2 = (1-t)^2$$

$$\mathcal{B}_1^2 = \binom{2}{1} t^1 (1-t)^1 = 2 \cdot t \cdot (1-t) = 2t(1-t)$$

$$\mathcal{B}_2^2 = \binom{2}{2} t^2 (1-t)^0 = 1 \cdot t^2 \cdot 1 = t^2$$

Polinômios de Bernstein de grau 3: Usando a definição temos,

$$\mathcal{B}_0^3 = \binom{3}{0} t^0 (1-t)^3 = 1 \cdot 1 \cdot (1-t)^3 = (1-t)^3$$

$$\mathcal{B}_1^3 = \binom{3}{1} t^1 (1-t)^2 = 3 \cdot t \cdot (1-t)^2 = 3t(1-t)^2$$

$$\mathcal{B}_2^3 = \binom{3}{2} t^2 (1-t)^1 = 3 \cdot t^2 \cdot (1-t)^1 = 3t^2(1-t)$$

$$\mathcal{B}_3^3 = \binom{3}{3} t^3 (1-t)^0 = 1 \cdot t^3 \cdot (1-t)^0 = t^3$$

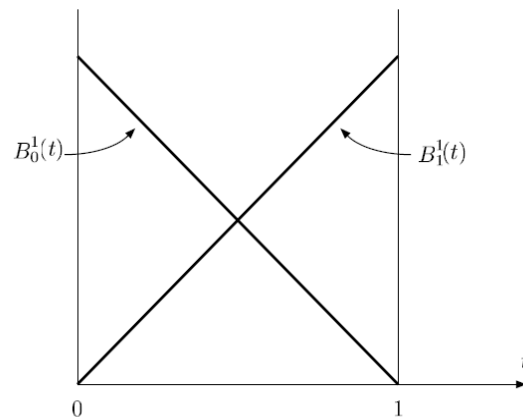
Podemos agrupar os polinômios de modo a se organizar com a mesma estrutura do triângulo de Pascal. De fato, os Polinômios de Bernstein satisfazem uma relação de recorrência.

Tabela 2 – Exemplos de Polinômios de Bernstein

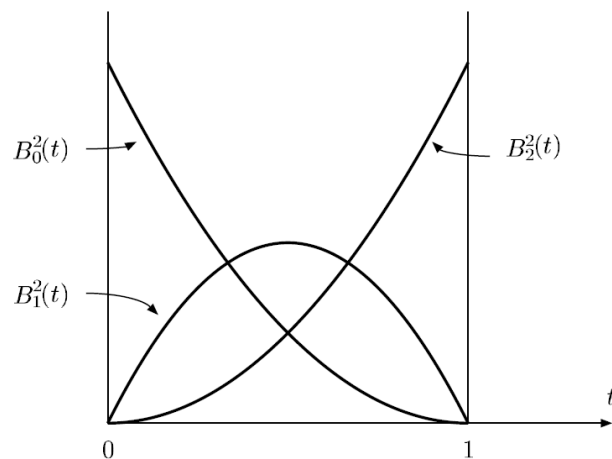
i / n	0	1	2	3	4	5	6
0	1	$(1-t)$	$(1-t)^2$	$(1-t)^3$	$(1-t)^4$	$(1-t)^5$	$(1-t)^6$
1		t^1	$2t(1-t)$	$3t(1-t)^2$	$4t(1-t)^3$	$5t(1-t)^4$	$6t(1-t)^5$
2			t^2	$3t^2(1-t)$	$6t^2(1-t)^2$	$10t^2(1-t)^3$	$15t^2(1-t)^4$
3				t^3	$4t^3(1-t)$	$10t^3(1-t)^2$	$20t^3(1-t)^3$
4					t^4	$5t^4(1-t)$	$15t^4(1-t)^2$
5						t^5	$6t^5(1-t)$
6							t^6

Fonte: elaborado pelo autor,

A seguir representamos as curvas polinomiais de Bernstein de graus 1, 2 e 3.

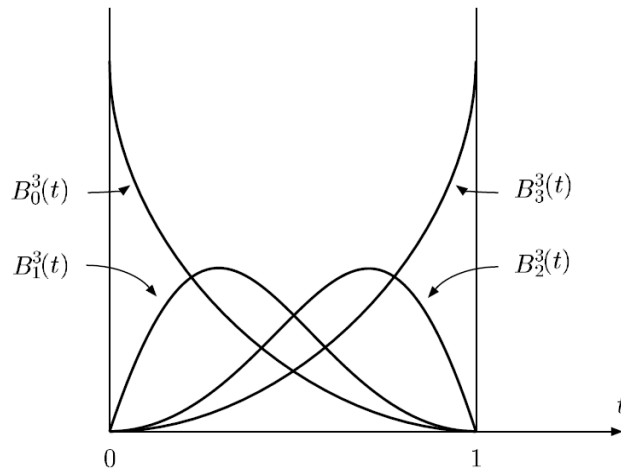
Figura 9 - Polinômio de Bernstein de grau 1

Fonte: elaborado pelo autor,

Figura 10 - Polinômio de Bernstein de grau 2

Fonte: elaborado pelo autor

Figura 11 - Polinômio de Bernstein de grau 3



Fonte: elaborado pelo autor.

Proposição 3.4.4 As derivadas dos polinômios de Bernstein de grau n são polinômios de grau $n - 1$. Além disso, a derivada pode ser escrita como uma combinação linear de polinômios de Bernstein. Em particular

$$\frac{d}{dt} B_i^n(t) = n(B_{i-1}^{n-1}(t) - B_i^{n-1}(t)) \text{ para } 0 \leq i \leq n.$$

Demonstração: Demonstraremos usando as condições de diferenciação de forma imediata, assim temos

$$\begin{aligned} \frac{d}{dt} B_i^n(t) &= \frac{d}{dt} \binom{n}{i} t^i (1-t)^{n-i} \\ &= \frac{d}{dt} \left(\frac{n!}{(n-i)! i!} t^i (1-t)^{n-i} \right) \\ &= \frac{in!}{(n-i)! i!} t^{i-1} (1-t)^{n-i} + \frac{(n-i)n!}{(n-i)! i!} t^i (1-t)^{n-i-1} \\ &= \frac{in(n-1)!}{(n-i)! i(i-1)!} t^{i-1} (1-t)^{n-i} \\ &\quad + \frac{(n-i)n(n-1)!}{(n-i)(n-i-1)! i!} t^i (1-t)^{n-i-1} \\ &= \frac{n(n-1)!}{(n-i)! (i-1)!} t^{i-1} (1-t)^{n-i} + \frac{n(n-1)!}{(n-i-1)! i!} t^i (1-t)^{n-i-1} \end{aligned}$$

$$\begin{aligned}
&= n \left(\frac{(n-1)!}{(n-i)!(i-1)!} t^{i-1} (1-t)^{n-i} + \frac{(n-1)!}{(n-i-1)!i!} t^i (1-t)^{n-i-1} \right) \\
&= n \left(\binom{n-1}{i-1} t^{i-1} (1-t)^{n-i} + \binom{n-1}{i} t^i (1-t)^{n-i-1} \right) \\
&= n \left(B_{i-1}^{n-1}(t) - B_i^{n-1}(t) \right). \quad \blacksquare
\end{aligned}$$

Para concluirmos esse momento inicial precisamos compreender que os polinômios de Bernstein de grau n formam uma base para o espaço dos polinômios de grau menor ou igual a n , uma vez que qualquer polinômio pode ser escrito como uma combinação linear de polinômios de Bernstein.

Proposição 3.4.5 Os polinômios de Bernstein de grau n formam uma base para o espaço vetorial dos polinômios de grau menor ou igual a n .

Demonstração: Temos a dimensão deste espaço vetorial valendo $n+1$, dessa forma provaremos o resultado verificando que as $n+1$ polinômios de Bernstein de grau n são linearmente independentes.

Tome $c_i \in \mathbb{R}$, tal que:

$$\sum_{i=0}^n c_i t^i (1-t)^{n-i} = 0 \quad (2)$$

para qualquer $t \in (0,1)$. Vamos dividir a expressão (2) por $(1-t)^n$ e escrever

$$\sum_{i=0}^n c_i t^i \frac{(1-t)^{n-i}}{(1-t)^n} = 0 \Leftrightarrow \sum_{i=0}^n c_i t^i (1-t)^{-i} = 0 \quad (3)$$

Definimos $y := \frac{t}{1-t}$ dessa forma a expressão (3) se reduz a $\sum_{i=0}^n c_i y^i = 0$, sendo essa última expressão de tal modo que $\sum_{i=0}^n c_i y^i = 0, \forall y > 0$, resta por consequência que para termos a igualdade verdadeira deveremos ter

$$c_0 = c_1 = c_2 = \dots = c_n = 0. \quad \blacksquare$$

De posse dessas condições podemos reescrever os polinômios de Bernstein na forma matricial, que pode ser útil em diversas aplicações, em especial na computação.

Seja $\mathcal{B}(t)$ um polinômio no espaço dos polinômios de grau no máximo n . Como os polinômios de Bernstein de grau n formam uma base desse espaço podemos escrever $\mathcal{B}(t) = c_0\mathcal{B}_0^n(t) + c_1\mathcal{B}_1^n(t) + \dots + c_n\mathcal{B}_n^n(t)$. Escrevendo-o como um produto de dois vetores temos:

$$\mathcal{B}(t) = [\mathcal{B}_0^n(t) \quad \mathcal{B}_1^n(t) \quad \dots \quad \mathcal{B}_n^n(t)] \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{bmatrix}$$

Escrevendo os polinômios de Bernstein na base canônica $1, t, t^2, \dots, t^n$, obteremos

$$B_i^n = \sum_{j=0}^n b_i^j t^j$$

logo,

$$\mathcal{B}(t) = [1 \quad t \quad \dots \quad t^n] \begin{bmatrix} b_0^0 & 0 & 0 & 0 & 0 \\ b_0^1 & b_1^1 & 0 & 0 & 0 \\ b_0^2 & b_1^2 & b_2^2 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_0^n & b_1^n & b_2^n & \dots & b_n^n \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{bmatrix}.$$

Exemplo 3.4.1 No caso quadrático ($n = 2$), a representação matricial é

$$\mathcal{B}(t) = [1 \quad t \quad t^2] \begin{bmatrix} 1 & 0 & 0 \\ -2 & 2 & 0 \\ 1 & -2 & 1 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix}$$

Exemplo 3.4.2 No caso cúbico ($n = 3$), a representação matricial é

$$\mathcal{B}(t) = [1 \quad t \quad t^2 \quad t^3] \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

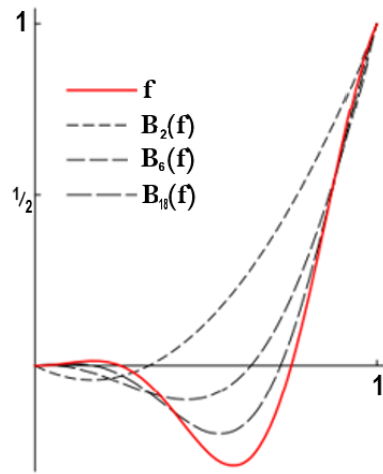
Definição 3.4.3 Seja f uma função de valor real definida e limitada no intervalo $[0, 1]$, seja $B_n(f)(t)$ o polinômio em $[0, 1]$ que atribui a t o valor

$$\sum_{i=0}^n \binom{n}{i} t^i (1-t)^{n-i} f\left(\frac{i}{n}\right).$$

$B_n(f)(t) = B_n(f; t)$ é a n -ésimo **aproximação polinomial de Bernstein**.

Para exemplificar esse modelo temos na figura abaixo, $f: [0, 1] \rightarrow \mathbb{R}$ sendo uma função definida por $f(t) = t^2 \cos(2\pi t)$. Podemos perceber que os gráficos dos polinômios de Bernstein vão se aproximando de f .

Figura 12 – Aplicação do polinômio de Bernstein



Fonte: elaborado pelo autor.

A partir da definição (3.4.3) temos a proposição abaixo, algumas obtidas de forma imediata outras requerendo mais atenção.

Proposição 3.4.6 Seja B_n a aproximação polinomial de Bernstein então são válidas.

$$I) \quad B_n(1) = \sum_{i=0}^n \binom{n}{i} t^i (1-t)^{n-i} = 1$$

$$II) \quad B_n(t) = \sum_{i=0}^n \binom{n}{i} \frac{i}{n} t^i (1-t)^{n-i} = t$$

$$III) \quad B_n(t^2) = \sum_{i=0}^n \binom{n}{i} \frac{i^2}{n^2} t^i (1-t)^{n-i} = \frac{(n-1)t^2}{n} + \frac{t}{n}$$

$$IV) \quad B_n(t^3) = \sum_{i=0}^n \binom{n}{i} \frac{i^3}{n^3} t^i (1-t)^{n-i} = \frac{(n-1)(n-2)t^3}{n^2} + \frac{3(n-1)t^2}{n^2} + \frac{t}{n^2}$$

Provaremos o item (III) as demais podem ser verificadas de forma análoga. Dessa forma seguem as igualdades.

$$\begin{aligned}
 B_n(t^2) &= \sum_{i=0}^n \binom{n}{i} \frac{i^2}{n^2} t^i (1-t)^{n-i} \\
 &= \sum_{i=0}^n \binom{n}{i} \frac{i}{n} \cdot \frac{i}{n} t^i (1-t)^{n-i} \\
 &= \sum_{i=0}^n \binom{n-1}{i-1} \frac{i}{n} t^i (1-t)^{n-i} \\
 &= t \sum_{i=0}^n \binom{n-1}{i-1} \frac{i}{n} t^{i-1} (1-t)^{n-i-(i-1)} \\
 &= t \sum_{i=0}^{n-1} \binom{n-1}{i} \frac{1+i}{n} t^i (1-t)^{n-1-i} \\
 &= t \left[\sum_{i=0}^{n-1} \binom{n-1}{i} \frac{i}{n} t^i (1-t)^{n-1-i} + \sum_{i=0}^{n-1} \binom{n-1}{i} \frac{1}{n} t^i (1-t)^{n-1-i} \right] \\
 &= t \left[\frac{n-1}{n} \sum_{i=0}^{n-1} \binom{n-1}{i} \frac{i}{n-1} t^i (1-t)^{n-1-i} + \sum_{i=0}^{n-1} \binom{n-1}{i} \frac{1}{n} t^i (1-t)^{n-1-i} \right] \\
 &= t \left(\frac{n-1}{n} t + \frac{1}{n} \right) \\
 &= t^2 + \frac{1}{n} (t - t^2) \quad \blacksquare
 \end{aligned}$$

O estudo das interpolações e dos polinômios de Bernstein estão interligados, já que é possível usar interpolação para encontrar pontos de controle para criar o que conheceremos como curva de Bézier, enquanto os polinômios de Bernstein são usados para definir a posição desses pontos de controle. Esses conceitos são extremamente importantes na matemática e na ciência da computação, permitindo a criação de curvas suaves e precisas para representar uma ampla variedade de fenômenos.

Nesse contexto, dentre os métodos possíveis, para funções contínuas, os polinômios de Bernstein se destacam por oferecerem uma aproximação uniforme. Como sempre, há um custo a ser suportado: a convergência não é rápida, quando comparada a outros métodos de aproximação polinomial. Contudo a apropriação das aplicações dos polinômios de Bernstein é atrativa, pois são de grande utilidade nos casos em que se necessita aproximar uma função em todo um intervalo. Verifica-se que como ferramenta de ajustes esses polinômios apresentam vantagens principalmente no desenho assistido por computador.

4 CURVAS DE BÉZIER

4.1 Introdução

As técnicas de aproximação de curvas são muito usadas em ambientes de projeto (CAD) interativos, por serem mais intuitivas do que técnicas de ajuste, por esse fato é muito comum que as curvas de Bézier seja muito difundida em áreas como engenharia mecânica e mecatrônica, computação gráfica, em design de produtos, arquitetura, entre outros. Um método adequado para o design de curvas e superfícies de forma livre em ambientes interativos foi desenvolvido como veremos por Pierre Bézier. Uma curva de Bézier é determinada por um conjunto de pontos de controle (polígono de controle) essas curvas podem ser lineares, quadráticas, cúbicas, etc., dependendo do número de pontos de controle e da ordem do polinômio usado para obter a aproximação. No decorrer desse trabalho apresentamos, definimos, caracterizamos a construímos diversas propriedades de conceitos matemáticos, que auxiliarão no desenvolvimento das curvas de Bézier em especial os polinômios de Bernstein e as curvas paramétricas.

4.2 Principais Aspectos Históricos das curvas de Bézier e sua relação com a computação.

Desenvolvido em 1962, leva o nome de quem publicou o primeiro trabalho sobre esse modelo de curva, o francês Pierre Etienne Bézier (Figura 12), funcionário da Renault, que o utilizou para projetos de carros. Foi construído de acordo com o algoritmo de Paul de Casteljaou da Citroën em 1957, cuja importância daremos a seguir, e formalizado nos anos 60.

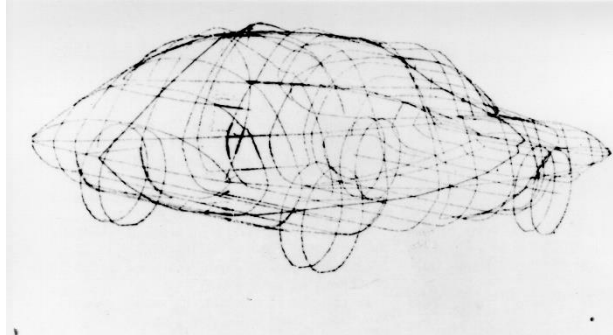
Figura 13 – Pierre Etienne Bézier



Fonte: elaborado pelo autor.

Bézier (1910 – 1999) foi um engenheiro precursor da área de modelagem e também na área de representação de curvas, de forma mais direta e especial no sistema de CAD/CAM – o “CA” é a abreviação do termo Computer-Aided (assistido por computador), o que significa que os sistemas foram criados para ajudar a alcançar o seu objetivo da forma mais rápida possível, usando o poder dos computadores para processamento. Em CAD, a última abreviação vem do termo Design (desenho, planta) e para o CAM – Manufacturing (manufatura). Foi pioneiro na gestão e transformação de manufaturas utilizando ferramentas de computação e em transformação de design utilizando computador e modelagem tridimensional. Abaixo uma ilustração da conferência de Pierre Bézier de 23 de outubro de 1964.

Figura 14 - Croqui do corpo em CAD pelo sistema UNISURF



Fonte: <https://journals.openedition.org/artefact/6711#ftn19> (Acesso em 11.01.2023)

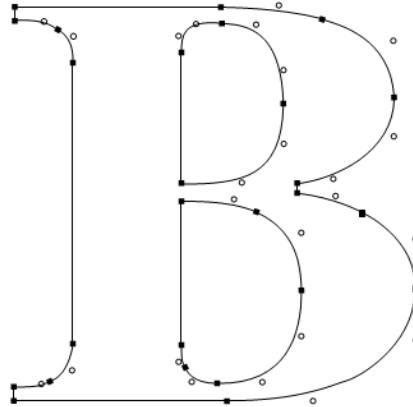
Vale salientar que Bézier patenteou e difundiu as curvas que levam seu nome, porém ele

não as criou. Essa popularização dada por ele, propiciou que a maioria dos sistemas de computação gráfica atuais e de design auxiliados por computador use as técnicas como referência. Segundo (FARIN, 1997) “Bézier e Casteljau desenvolveram as curvas com o intuito de modelar as formas aerodinâmicas dos automóveis modernos”.

Como foi comentado anteriormente, as curvas de Bézier foram e são utilizadas para diversos fins que vão além da aerodinâmica de carros e projetos automobilísticos. Seu uso é muito difundido em softwares de edição de imagens e em diversas modelagens de tipos de contornos de fontes de texto e de linguagens da computação. No tópico de interpolação falamos brevemente sobre a fonte PostScript, outra que tem grande representatividade é a TrueType que foi

desenvolvida pela Apple Computer e posteriormente licenciada para a Microsoft Corporation.

Figura 15 – Modelo TrueType



Fonte: elaborado pelo autor.

Não diferente dos tempos atuais, no século passado, os segredos industriais eram mantidos em sigilo absoluto. De tal forma que grande parte dos trabalhos de Bézier são frutos de um estudo inicial atribuído a Paul de Faget de Casteljaud (1930 – 2022). Casteljaud (Figura 14), nascido na França, estudou matemática e física na École Normale Supérieure de Paris, entrou na empresa automobilística Citroën em 1958 onde trabalhou até 1992 dedicado ao grupo de pesquisa Groupe Détermination Mathématique des Carrosseries da Citroën. Seu empenho resultou no desenvolvimento do estudo e conseqüentemente na criação de um algoritmo para a aplicação das curvas e superfícies com pólos. Esses estudos permaneceram ocultos por força de patente até a exibição de trabalho semelhante apresentado por Bézier na década de 1960.

Figura 16 – Paul de Faget de Casteljaud



Fonte: Google imagens

Casteljau usou os Polinômios de Bernstein, que serviram como funções de base para superfície e curvas polinomiais paramétricas. Seu trabalho resultou numa ferramenta fundamental para entender e trabalhar com curvas definidas por dois ou mais pontos de controle. Sua contribuição foi imprescindível na área de polinômios multilineares que forneceram um algoritmo de avaliação eficiente e estável. Seu estudo conseguiu estabelecer uma relação próxima entre as formas polares e pontos de controle chamados B-spline.

Devido a sua grande e valiosa participação nos estudos das curvas e modelagens, Casteljau, recebeu em 2012 o prêmio Bézier da Solid Modeling Association (SMA) e vale destacar que em seu anúncio a premiação ressalta:

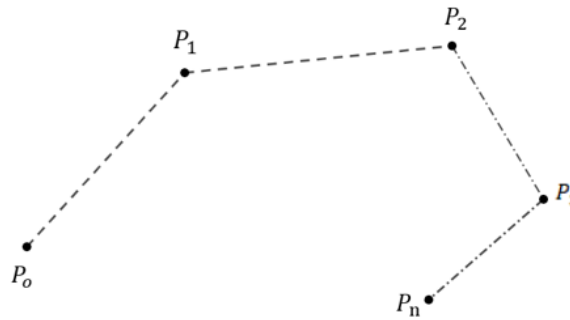
As contribuições de Paul de Casteljau são menos conhecidas do que deveriam, porque ele não foi capaz de publicá-las até que ideias equivalentes fossem reinventadas independentemente por outros, às vezes de uma forma bastante diferente, mas agora reconhecidamente relacionadas. Como ele não teve permissão para publicar seus primeiros trabalhos, agora chamamos os polinômios com base de Bernstein de “polinômios de Bézier”, embora o próprio Bézier não usasse pontos de controle, mas seus primeiros vetores de diferença como coeficientes. Também chamamos os polinômios multilineares de “florescimento”, seguindo Lyle Ramshaw, que por sua vez creditou a Casteljau a “abordagem polar” subjacente à teoria matemática de splines. (SOLID MODELING ASSOCIATION (SMA), 2012, tradução nossa.)

4.3 Definição de Curvas de Bézier via Algoritmo de Casteljau e polinômio de Bernstein

O algoritmo descrito abaixo tem alta importância no campo do design de curvas e superfícies, provavelmente o de maior relevância nessa área de estudo. Embora apresente um nível de significância elevada, se revela uma aplicação relativamente simples. Sua condução apresenta uma interação entre álgebra e geometria que nos levará a resultados poderosos.

Definição 4.3.1 Dizemos que os pontos $P_0, P_1, P_2, \dots, P_n \in \mathbb{R}^2$ formam um polígono de controle se os segmentos $\overline{P_0P_1} \cup \overline{P_1P_2} \cup \dots \cup \overline{P_{i-1}P_i}$ formarem uma linha poligonal contínua. Denotaremos o polígono de controle por $P = [P_0, P_1, P_2, \dots, P_n]$.

Figura 17 – Polígono de Controle



Fonte: elaborado pelo autor, no Geogebra.

O método descrito pelo algoritmo de Casteljau gera as curvas de Bézier, que de forma simplificada, são obtidas a partir da aplicação de interpolações lineares repetidas vezes. Com isso obteremos um segmento de reta passando por um par de pontos de controle consecutivos. Ao aplicar esse processo $(n - 1)$ vezes obteremos o ponto pertencente à curva. O método de Casteljau, para quatro pontos de controle (para mais pontos segue mesmo modelo) funciona da seguinte forma:

- I) Primeiro, forme r_0, r_1, r_2 por interpolação linear a partir dos pontos de controle (P_0, P_1, P_2, P_3) por $r_i(t) = (1 - t)P_i + t(P_{i+1})$.
- II) Em seguida, defina s_0 e s_1 por interpolação linear dos r_i 's por $s_i(t) = (1 - t)r_i + t(r_{i+1})$.
- III) Por fim, defina v_0 por interpolação linear de s_0 e s_1 por $v_0(t) = (1 - t)s_0 + t(s_1)$.

Definição 4.3.2 Uma curva de Bézier é uma aplicação $B(t): [0, 1] \rightarrow \mathbb{R}^2$ associada a um polígono de controle obtido a partir do Algoritmo de Casteljau aplicado de forma recursiva.

Para melhor assimilação representaremos o algoritmo em questão, em etapas ou passos lógicos. Porém, deveremos antes, supor $P = [P_0, P_1, P_2, \dots, P_n]$ um polígono de controle onde serão aplicadas as recursões.

Em primeiro momento fixamos um parâmetro t definido no intervalo $[0, 1]$, em seguida, tomamos $P_i^0(t) \equiv P_i$ para $i = 0, 1, 2, \dots, n$. Em terceiro momento definimos

$$P_i^j(t) = (1 - t) \cdot P_{i-1}^{j-1}(t) + t \cdot P_i^{j-1}(t); j = 1, 2, \dots, n \text{ e } i = 1, \dots, n - j,$$

por fim teremos $\mathcal{B}(t) = P_n^n(t)$. Assim todas as coordenadas da aplicação $\mathcal{B}(t)$ podem ser obtidas por um polinômio de grau n sob o parâmetro t .

Devido sua interpretação geométrica, iniciaremos com o exemplo de uma curva de Bézier de grau 2, ou seja, quadrática. A partir da de grau 2 conseguimos interpretar geometricamente as outras curvas de Bézier por recursividades.

As curvas de Bézier de grau 2 são obtidas por três pontos de controle P_0, P_1 e P_2 no plano. São definidas pela associação de dois segmentos de reta que são P_0P_1 e P_1P_2 . O procedimento consiste em algumas etapas, descreveremos o procedimento intuitivo abaixo.

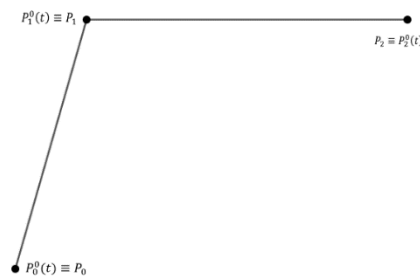
- 1) Construir sequencialmente os segmentos P_0P_1 e P_1P_2 .
- 2) Definir os pontos $P_1^1(t)$ e $P_1^2(t)$ a partir da divisão desses segmentos em mesma razão.
- 3) De $P_1^1(t)$ e $P_1^2(t)$ construir o segmento $P_1^1(t)P_1^2(t)$
- 4) Tomar o ponto $P_2^2(t)$ a partir da divisão do segmento $P_1^1(t)P_1^2(t)$ na mesma razão dos segmentos anteriores.

Vale ressaltar que a razão a que nos referimos na divisão do segmento será dado pelo parâmetro $t \in [0, 1]$.

Exemplo 4.3.1 Determinar a curva de Bézier quadrática (grau 2).

Usaremos o algoritmo de Casteljau. Nesse caso definiremos o polígono verificador, ou seja, $P = [P_0, P_1, P_2]$. O passo inicial mostra que $P_0^0(t) \equiv P_0$, $P_1^0(t) \equiv P_1$, $P_2^0(t) \equiv P_2$

Figura 18 – Primeira Etapa Do Algoritmo De Casteljau para a Curva De Bézier Quadrática



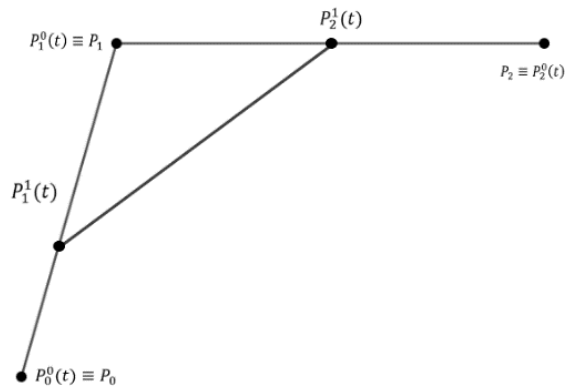
Fonte: elaborado pelo autor, no Geogebra.

Na segunda etapa do desenvolvimento temos

$$P_1^1(t) = (1 - t)P_0^0(t) + tP_1^0(t) = (1 - t)P_0 + tP_1 ,$$

$$P_2^1(t) = (1 - t)P_1^0(t) + tP_2^0(t) = (1 - t)P_1 + tP_2.$$

Figura 19 – Segunda Etapa Do Algoritmo De Casteljau para a Curva De Bézier Quadrática

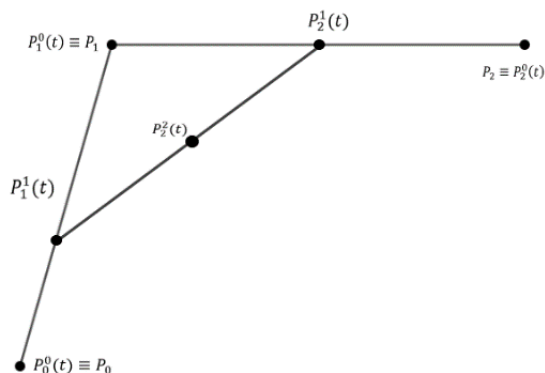


Fonte: elaborado pelo autor, no Geogebra.

Na terceira etapa do algoritmo temos,

$$\begin{aligned} P_2^2(t) &= (1 - t)P_1^1(t) + tP_2^1(t) = (1 - t)[(1 - t)P_0 + tP_1] + t[(1 - t)P_1 + tP_2] \\ &= (1 - t)P_1^1(t) + tP_2^1(t) = (1 - t)^2P_0 + 2t(1 - t)P_1 + t^2P_2. \end{aligned}$$

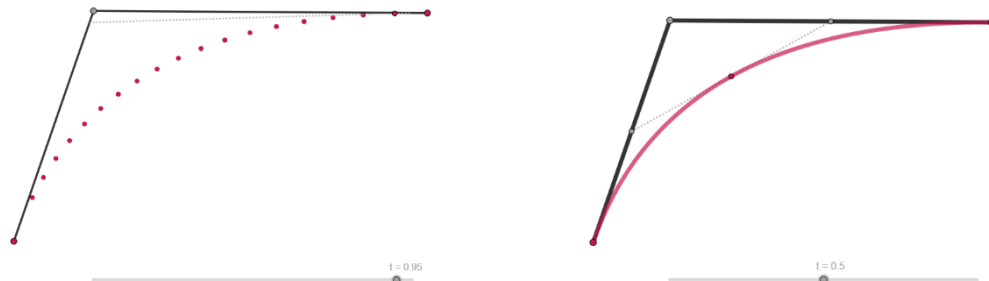
Figura 20 – Terceira Etapa Do Algoritmo De Casteljau para a Curva De Bézier Quadrática



Fonte: elaborado pelo autor, no Geogebra.

A imagem abaixo mostra o processo de deslocamento de $P_2^2(t)$ para forma a curva de Bézier $\mathcal{B}(t)$. Perceba que a variação de $t \in [0, 1]$ surge o traço da curva.

Figura 21 – Bézier Quadrática



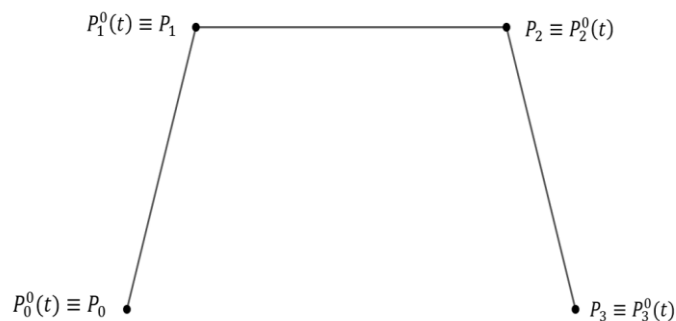
Fonte: elaborado pelo autor.

Uma observação a se fazer é que podemos perceber que pela construção geométrica do algoritmo fica verificada a obrigatoriedade de trabalharmos no intervalo $[0, 1]$. Uma vez que para valores menores que 0 ou maiores que 1 teremos um ponto fora do segmento formado pelos pontos de controle do algoritmo.

Exemplo 4.3.2 Determinar a curva de Bézier cúbica (grau 3).

Solução: Novamente usaremos o algoritmo de Casteljau. Nesse caso definimos o polígono verificador, ou seja, $P = [P_0, P_1, P_2, P_3]$. O passo inicial mostra que $P_0^0(t) \equiv P_0$, $P_1^0(t) \equiv P_1$, $P_2^0(t) \equiv P_2$, $P_3^0(t) \equiv P_3$

Figura 22 – Primeira Etapa Do Algoritmo De Casteljau para a Curva De Bézier Cúbica



Fonte: elaborado pelo autor.

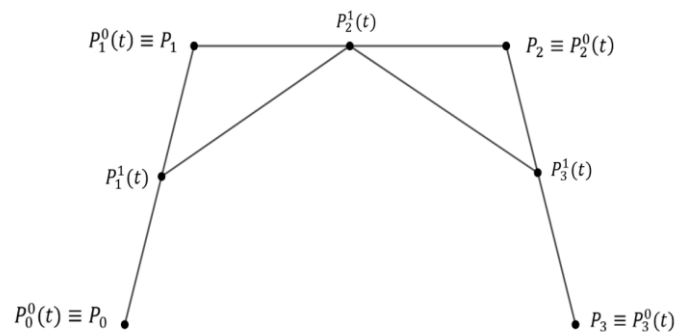
Na segunda etapa do desenvolvimento temos

$$P_1^1(t) = (1 - t)P_0^0(t) + tP_1^0(t) = (1 - t)P_0 + tP_1 ,$$

$$P_2^1(t) = (1 - t)P_1^0(t) + tP_2^0(t) = (1 - t)P_1 + tP_2 ,$$

$$P_3^1(t) = (1 - t)P_2^0(t) + tP_3^0(t) = (1 - t)P_2 + tP_3 .$$

Figura 23 – Segunda Etapa Do Algoritmo De Casteljaú para a Curva De Bézier Cúbica



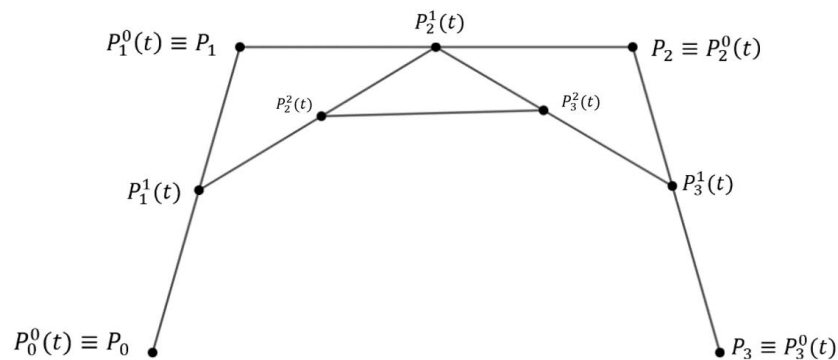
Fonte: elaborado pelo autor.

Seguimos para a terceira etapa do algoritmo, de tal modo que temos,

$$\begin{aligned} P_2^2(t) &= (1 - t)P_1^1(t) + tP_2^1(t) = (1 - t)[(1 - t)P_0 + tP_1] + t[(1 - t)P_1 + tP_2] \\ &= (1 - t)^2P_0 + 2t(1 - t)P_1 + t^2P_2. \end{aligned}$$

$$\begin{aligned} P_3^2(t) &= (1 - t)P_2^1(t) + tP_3^1(t) = (1 - t)[(1 - t)P_1 + tP_2] + t[(1 - t)P_2 + tP_3] \\ &= (1 - t)^2P_1 + 2t(1 - t)P_2 + t^2P_3 . \end{aligned}$$

Figura 24 – Terceira Etapa Do Algoritmo De Casteljaú para a Curva De Bézier Cúbica

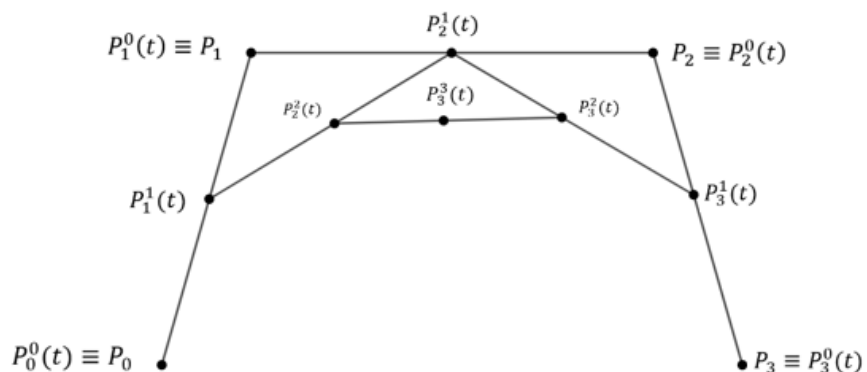


Fonte: elaborado pelo autor.

Para finalizarmos, seguimos para a última etapa do algoritmo para obtenção de curva de Bézier cúbica, desse modo

$$\begin{aligned} \mathcal{B}(t) &= P_3^3(t) = (1-t)P_2^2(t) + tP_3^2(t) \\ &= (1-t)[(1-t)^2P_0 + 2t(1-t)P_1 + t^2P_2] + t[(1-t)^2P_1 + 2t(1-t)P_2 + t^2P_3] \\ &= (1-t)^3P_0 + 3t(1-t)^2P_1 + 3t^2(1-t)P_2 + t^3P_3. \end{aligned}$$

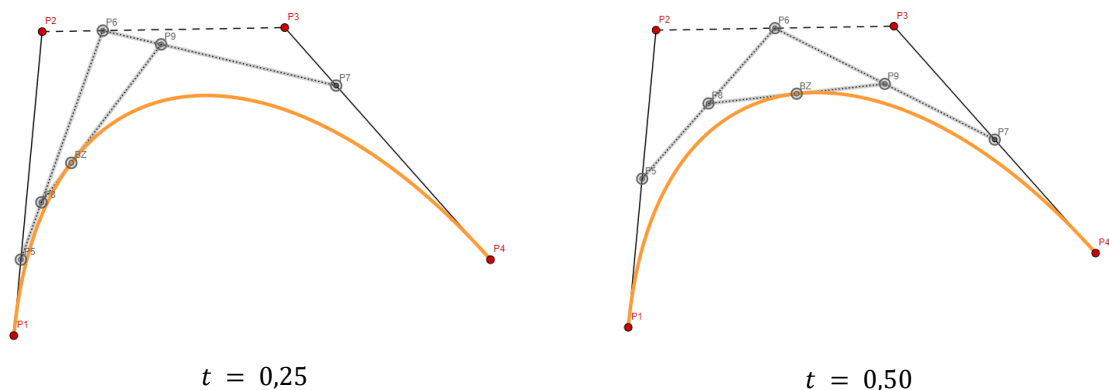
Figura 25 – Última etapa do algoritmo de Casteljau para a curva de Bézier cúbica

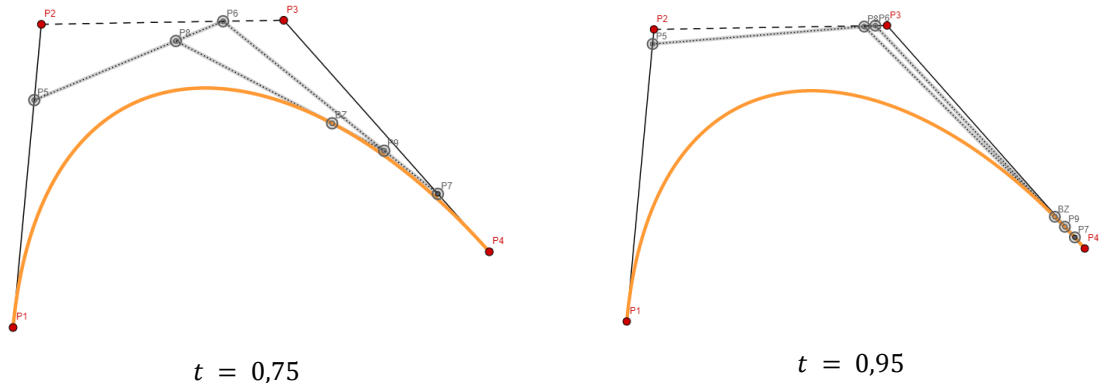


Fonte: elaborado pelo autor.

A última expressão representa o polinômio que gera a curva de Bézier cúbica. Devemos, portanto, estabelecer um valor para o parâmetro t a fim de reproduzir o traço da curva, como $t \in [0,1]$ podemos supor, a critério de exemplo o valor $t = \{0,25; 0,5; 0,75; 0,95\}$. Dessa forma o traço fica definido como na figura (26) abaixo.

Figura 26 – Bézier Cúbica





Fonte: elaborado pelo autor.

Definimos a seguir as curvas de Bézier, diferentemente da anterior que usa algoritmo de Casteljau, como uma relação direta com os polinômios de Bernstein. Como vimos se desejarmos uma aproximação de uma curva polinomial entre dois pontos quaisquer conhecidos, P_0 e P_1 , a solução mais eficiente é pensarmos em um segmento de reta que passa pelos pontos dados. A equação parametrizada mais comum é $P(t) = (1 - t)P_0 + tP_1$, de tal modo que é viável pensar em $P(t)$ como uma média ponderada entre P_0 e P_1 . Para BIEZUNER (2014 p.16)

Definição 4.3.3 Utilizando polinômios de Bernstein definimos as curvas de Bézier como:

$$C(t) = \sum_{i=0}^n P_i B_i^n(t)$$

onde B_i^n são os polinômios de Bernstein de grau n e P_i são os pontos de controle escolhidos previamente. O grau do polinômio $C(t)$ é o grau da curva de Bézier. Assim podemos determinar as coordenadas da seguinte forma:

$$x(t) = \sum_{i=0}^n x_i B_i^n(t) \quad e \quad y(t) = \sum_{i=0}^n y_i B_i^n(t)$$

Algumas observações sobre as curvas de Bézier são de grande importância podemos destacar:

I – As curvas de Bézier com n pontos de controle tem grau $(n - 1)$.

II – Vale relatar que curvas com grau muito alto são difíceis de desenhar, possuem certa complexidade e estão sujeitas a erros de precisão.

III – Para se trabalhar com grau qualquer da curva a solução elementar é concatenar curvas polinomiais de grau menor.

4.4 Propriedades das Curvas de Bézier

Evidenciamos nesse momento as principais propriedades extraídas das definições anteriores. Em geral, grande parte dessas propriedades podem ser percebidas como similares aos que já trabalhamos quando mencionamos as características dos polinômios de Bernstein.

I) Propriedade de invariância afim: Qualquer transformação nos pontos de controle implica em transformação na curva. Para (BUSS, 2003) “a curva é uma aproximação mais suave do polígono de controle. Assim, a quantidade de vezes que os segmentos do polígono de controle interceptam a curva será no máximo a quantidade destes segmentos”. Já de acordo com De Oliveira (2018): “A curva é invariante sob transformações geométricas afins (rotação, translação, escala, ...), assim, transformar os pontos de controle e desenhar a curva é equivalente a desenhar a curva transformada”.

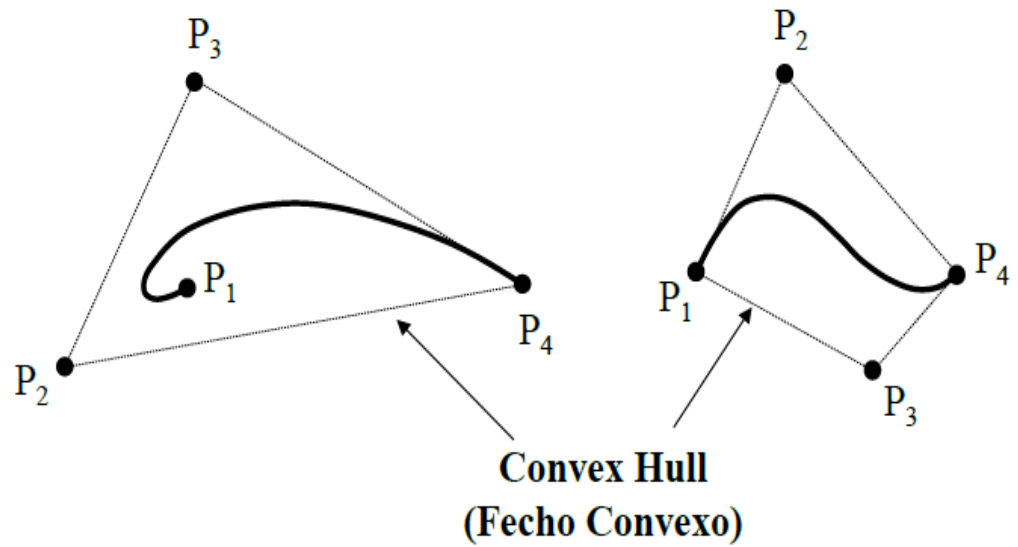
II) Propriedade da invariância sob transformações de parâmetros afins: Algebricamente podemos representar essa propriedade pela relação

$$\sum_{i=0}^n \mathcal{P}_i \mathcal{B}_i^n(t) = \sum_{i=0}^n \mathcal{P}_i \mathcal{B}_i^n\left(\frac{u-a}{b-a}\right)$$

Essa condição é importante, pois se pode pensar na curva como sendo definida sobre qualquer intervalo arbitrário $a < u < b$ na reta real, onde $t = \frac{u-a}{b-a}$. Esta propriedade é obtida a partir do processo de interpolação linear.

III) Propriedade do fecho convexo: Segundo, Farin (1997), essa condição é válida, pois “para todo $t \in [0,1]$ os polinômios de Bernstein são não negativos e somam um”. Em outras palavras, a curva sempre estará contida no fecho convexo do polígono de controle.

Figura 27 – Fecho Convexo

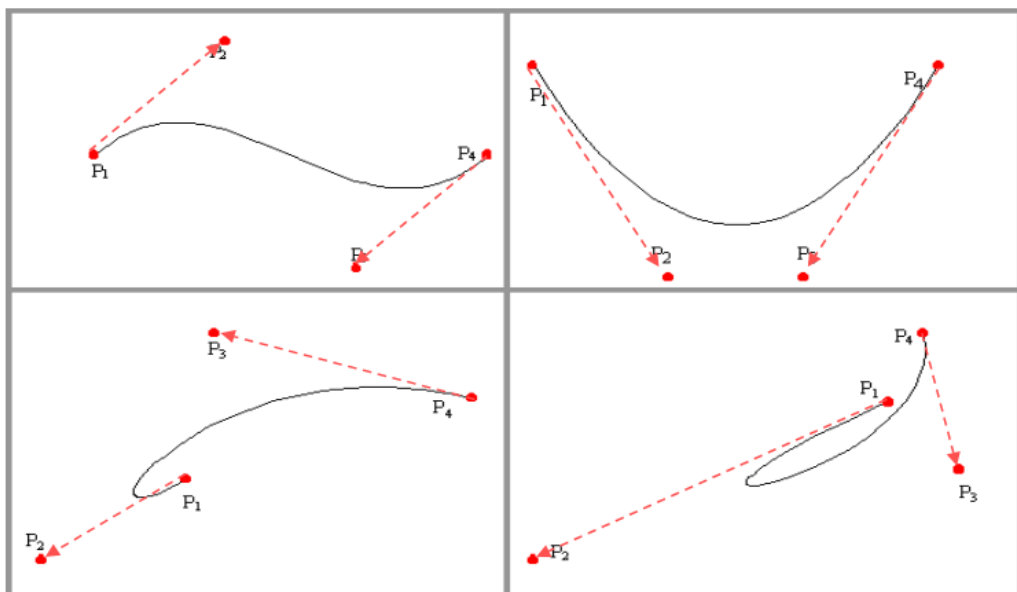


Fonte: (DE OLIVEIRA, 2018)

IV) Interpolação dos pontos extremos: a curva de Bézier passa sempre através dos extremos \mathcal{P}_0 e \mathcal{P}_n , isso decorre da propriedade das raízes dos polinômios de Bernstein, pois: $\mathcal{C}(0) = \mathcal{P}_0$ e $\mathcal{C}(1) = \mathcal{P}_n$, a critério de modelo podemos retornar ao exemplo (4.3.3).

V) Design interativo: Forma da curva ‘acompanha’ a forma do polígono de definição o que implica formulação adequada ao ‘design’ interativo de formas

Figura 28 – Relação entre forma das curvas e posição dos pontos de controle



Fonte: (WANGENHEIM; WAGNER, [2005], p. 102)

VI) A figura (25) também pode ilustrar a propriedade chamada falso controle local: Seja uma curva de Bézier com P_n pontos de controle. Quando movemos apenas um dos vértices do polígono de controle, P_i por exemplo, a curva sofre uma mudança mais acentuada na região próxima ao ponto, um tanto previsível, embora a alteração afete toda a curva.

4.5 Propriedade da Subdivisão Recursiva e Bézier por Partes

Subdivisão recursiva é a técnica que consiste em dividir uma curva de Bézier em outras curvas menores que chamaremos de subcurvas. A ideia de subdividir uma curva de Bézier se dá, entre outros fatores, ao fato de ao seccionarmos uma curva como essa podemos trabalhar por aproximação de segmentos de retas.

Suponhamos a existência de uma curva cúbica de Bézier $\mathcal{C}(t)$, com os pontos de controle P_0, P_1, P_2, P_3 . Se particionarmos $\mathcal{C}(t)$ em $\mathcal{C}_1(t) = \mathcal{C}\left(\frac{t}{2}\right)$ e $\mathcal{C}_2(t) = \mathcal{C}\left(\frac{t+1}{2}\right)$ teremos duas curvas cúbicas. Suporemos, ainda, o domínio dentro do intervalo $[0, 1]$. Fica claro que se $0 < t < 1$ termos $\mathcal{C}_1(t)$ como a primeira parte de $\mathcal{C}(t)$ em $0 < t < \frac{1}{2}$ da mesma forma $\mathcal{C}_2(t)$ será a curva que é a segunda parte de $\mathcal{C}(t)$.

Se definirmos (ver figura 30)

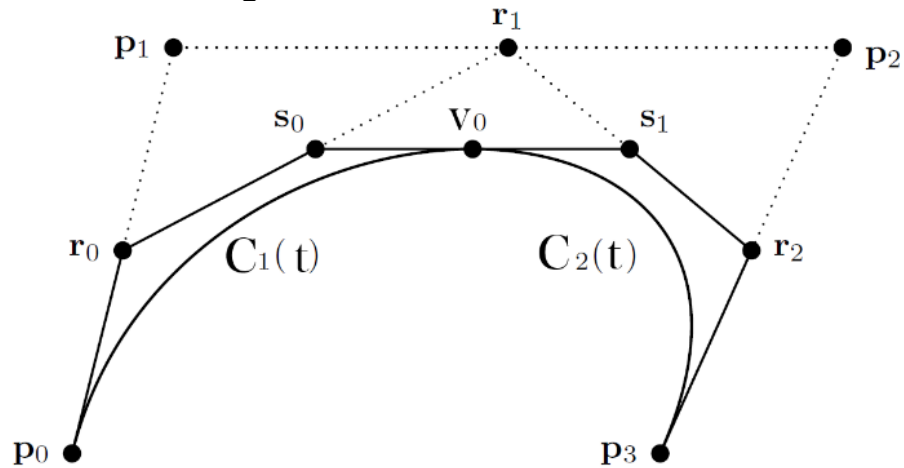
$$r_i = (1 - t) \cdot P_i + t \cdot P_{i+1},$$

$$s_i = (1 - t) \cdot r_i + t \cdot r_{i+1} \quad e$$

$$v_0 = (1 - t) \cdot s_0 + t \cdot s_1$$

que em particular para $t = \frac{1}{2}$ teremos $r_i = \frac{P_i + P_{i+1}}{2}$, $s_i = \frac{r_i + r_{i+1}}{2}$ e $v_0 = \frac{s_0 + s_1}{2}$. Tomando $\mathcal{C}(t)$, $\mathcal{C}_1(t)$ e $\mathcal{C}_2(t)$ como acima. Então a curva $\mathcal{C}_1(t) = \mathcal{C}(t)$ nos pontos de controle P_0, r_0, s_0, v_0 e $\mathcal{C}_2(t) = \mathcal{C}(t)$ com os pontos de controle P_3, r_2, s_1, v_0 .

Figura 29 – Subdivisão Recursiva



Fonte: (buss, 2003)

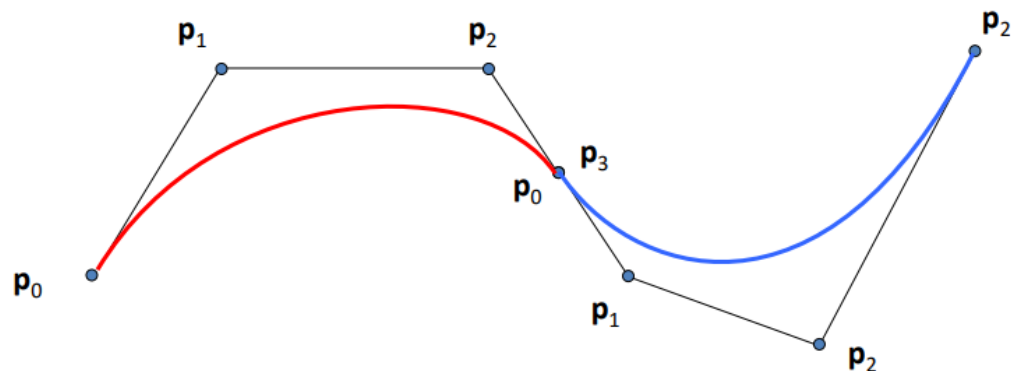
A verificação da condição acima pode ser feita de forma direta, uma vez que $C_1(t)$ e $C_2(t)$ estão definidas e especificadas com os pontos de controle r_i e s_i , mas esses pontos foram definidos em função de P_i . Da mesma forma $C_1(t)$ e $C_2(t)$ são definidas como $C_1(t) = c\left(\frac{t}{2}\right)$ e $C_2(t) = c\left(\frac{t+1}{2}\right)$. Dessa forma a afirmação acima pode ser concluída por cálculo direto. Essa análise foi pontual é possível fazer a generalização, porém vai além de nossas intenções.

A ideia oposta é a propriedade de associar várias curvas de Bézier de grau menores para obter curvas mais complexas, aqui chamaremos de aproximação por partes. Uma vez que, uma grande desvantagem das curvas de Bézier é que não é possível definir uma curva de Bézier cúbica para aproximar ou representar um conjunto de n pontos sem utilizar múltiplos segmentos de curva. Queremos afirmar nesse momento que curvas complexas podem ser obtidas ‘concatenando’ várias curvas de grau baixo.

Para assegurar a continuidade entre segmentos de curva, definem-se restrições adicionais de continuidade.

A figura abaixo ilustrará uma situação em seguida indicaremos as condições que a compõem.

Figura 30 – Concatenando Curvas Bézier



Fonte: Autor, feito no Geogebra.

Pontuamos o procedimento da forma abaixo com suas definições:

- I) **Definição 4.5.1** Continuidade paramétrica de ordem 0 (C^0): Último ponto da primeira será igual ao primeiro ponto da segunda.
- II) **Definição 4.5.2** Continuidade paramétrica de ordem 1 (C^1): C^0 e segmento $\overline{P_2P_3}$ da primeira com mesma direção e comprimento que o segmento $\overline{P_0P_1}$ da segunda.
- III) **Definição 4.5.3** Continuidade paramétrica de ordem (C^2): C^1 e as restrições sobre pontos P_1 da primeira e P_2 da segunda.

4.6 Derivadas de Curva de Bézier

Vamos abordar esse tópico das curvas de Bézier recordando sua definição para grau três, usando polinômios de Bernstein. Essa abordagem nos dará embasamento para podermos entender a ideia para um grau qualquer. Uma propriedade inicial é de que essas curvas têm diferenciabilidade infinita isso afirma, em outras palavras, que todas as derivadas são contínuas.

As curvas Bézier de grau três são definidas parametricamente por uma função $\mathcal{C}(t)$: conforme t varia de 0 a 1, os valores de $\mathcal{C}(t)$ varrem a curva. A fórmula para uma curva Bézier de grau três como vimos é:

$$\mathcal{C}(t) = P_0B_0^3(t) + P_1B_1^3(t) + P_2B_2^3(t) + P_3B_3^3(t),$$

onde cada $\mathcal{B}_i^n(t)$ são funções de valor escalar e como vimos $\mathcal{B}_i^3(t) = \binom{3}{i}t^i(1-t)^{3-i}$ e $\binom{n}{i} = \frac{n!}{i!(n-i)!}$. Expandir as definições fornece polinômios de grau três. De fato, pela Tabela (2):

$$\mathcal{B}_0^3 = (1-t)^3 = 1 - 3t + 3t^2 - t^3$$

$$\mathcal{B}_1^3 = 3t(1-t)^2 = 3t - 6t^2 + 3t^3$$

$$\mathcal{B}_2^3 = 3t^2(1-t) = 3t^2 - 3t^3$$

$$\mathcal{B}_3^3 = t^3.$$

Se calcularmos manualmente as derivadas das quatro funções acima, é claro que descobriremos que suas derivadas são polinômios de grau dois, bastando apenas usar as regras de derivação. Vale salientar que essas condições são dadas anteriormente. Agora avaliando as derivadas em $t = 0$ e $t = 1$ verificamos que:

$$\mathcal{B}_0^{3'}(0) = -3 \text{ e } \mathcal{B}_0^{3'}(1) = 0$$

$$\mathcal{B}_1^{3'}(0) = 3 \text{ e } \mathcal{B}_1^{3'}(1) = 0$$

$$\mathcal{B}_2^{3'}(0) = 0 \text{ e } \mathcal{B}_2^{3'}(1) = -3$$

$$\mathcal{B}_3^{3'}(0) = 0 \text{ e } \mathcal{B}_3^{3'}(1) = 3$$

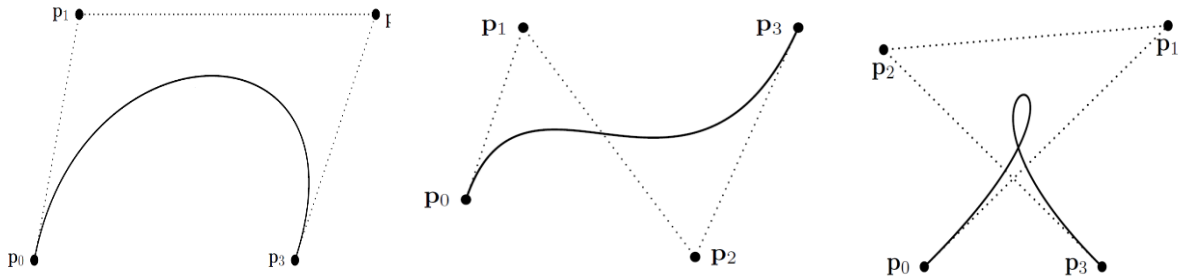
Como os pontos de controle são constantes e independentes da variável t , a derivada da curva reduz-se a:

$$\mathcal{C}'(t) = \mathcal{P}_0\mathcal{B}_0^{3'}(t) + \mathcal{P}_1\mathcal{B}_1^{3'}(t) + \mathcal{P}_2\mathcal{B}_2^{3'}(t) + \mathcal{P}_3\mathcal{B}_3^{3'}(t),$$

daí verificamos que no valor inicial e no valor final as derivadas tem os seguintes valores: $\mathcal{C}'(0) = 3(\mathcal{P}_1 - \mathcal{P}_0)$ e $\mathcal{C}'(1) = 3(\mathcal{P}_3 - \mathcal{P}_2)$.

No gráfico, isso significa que a curva $\mathcal{C}(t)$ começa em $t = 0$ percorrendo na direção do vetor de \mathcal{P}_0 para \mathcal{P}_1 . De forma análoga, no final, onde $t = 1$, a curva $\mathcal{C}(t)$ é tangente ao vetor de \mathcal{P}_2 a \mathcal{P}_3 . De acordo com (BUSS, 2003) isso corresponde ao “fato de que a curva começa em \mathcal{P}_0 inicialmente tangente ao segmento de linha que une o primeiro ponto de controle ao segundo ponto de controle e termina em \mathcal{P}_3 tangente ao segmento de linha que une o terceiro e o quarto pontos de controle”. Esse fato pode ser visualizado, por exemplo, na imagem abaixo.

Figura 31 – Comportamento de $\mathcal{C}(t)$



Fonte: (BUSS, 2003)

Proposição 4.6.1 Da proposição (3.4.4): $\frac{d}{dt}\mathcal{B}_i^n(t) = \mathcal{B}_i^{n-1}(t) - \mathcal{B}_{i-1}^{n-1}(t)$, então podemos escrever a derivada $\mathcal{C}(t)$ como:

$$\mathcal{C}'(t) = n \cdot \sum_{i=0}^{n-1} \mathcal{B}_i^{n-1}(t)(\mathcal{P}_{i+1} - \mathcal{P}_i)$$

Demonstração: Usaremos a definição do polinômio de Bernstein, assim temos:

$$\frac{d}{dt} \mathcal{B}_i^n(t) = \binom{n}{i} i t^{i-1} (1-t)^{n-1} - \binom{n}{i} (n-i) t^i (1-t)^{n-i-1}$$

Observe que o primeiro termo é zero se $i = 0$ e o segundo é zero se $i = n$. Assim, a derivada de $\mathcal{C}(t)$ é igual a

$$\begin{aligned} & \sum_{i=0}^n \binom{n}{i} i t^{i-1} (1-t)^{n-i} \mathcal{P}_i - \sum_{i=0}^n \binom{n}{i} (n-i) t^i (1-t)^{n-1-i} \mathcal{P}_i \\ &= \sum_{i=1}^n \binom{n}{i} i t^{i-1} (1-t)^{n-i} \mathcal{P}_i - \sum_{i=0}^{n-1} \binom{n}{i} (n-i) t^i (1-t)^{n-1-i} \mathcal{P}_i \end{aligned}$$

(Aqui removemos os termos zeros do somatório)

$$= \sum_{i=1}^n n \binom{n-1}{i-1} t^{i-1} (1-t)^{n-i} \mathcal{P}_i - \sum_{i=0}^{n-1} n \binom{n-1}{i} t^i (1-t)^{n-1-i} \mathcal{P}_i$$

(Aqui usamos o fato de $\binom{n}{i} i = n \binom{n-1}{i-1}$ e $\binom{n}{i} (n-i) = n \binom{n-1}{i}$)

$$= \sum_{i=0}^{n-1} n \binom{n-1}{i} t^i (1-t)^{n-1-i} \mathcal{P}_{i+1} - \sum_{i=0}^{n-1} n \binom{n-1}{i} t^i (1-t)^{n-1-i} \mathcal{P}_i$$

(Aqui usamos o conceito de mudança de variável no primeiro somatório)

$$= \sum_{i=0}^{n-1} n \binom{n-1}{i} t^i (1-t)^{n-1-i} (\mathcal{P}_{i+1} - \mathcal{P}_i)$$

$$= n \sum_{i=0}^{n-1} \mathcal{B}_i^{n-1}(t) (\mathcal{P}_{i+1} - \mathcal{P}_i) \quad \blacksquare$$

A seguir apresentaremos alguns modelos computacionais onde foram aplicadas as curvas de Bézier. Perceberemos que apesar da álgebra aplicada em seu desenvolvimento, seu uso é relativamente simples e intuitivo. Suas possibilidades de uso são inúmeras e ficarão a cargo de cada necessidade.

5 APLICAÇÕES DA FERRAMENTA BÉZIER

5.1 Introdução

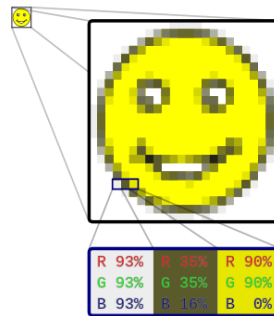
Muitos softwares de edição de imagens, vídeos e até de criação 3D possuem o recurso de criação de curvas Bézier. Profissionais gráficos e desenhistas em geral, recorrem com frequência à ferramenta de traçado de curvas em software de edição gráfica. Este tipo de ferramenta permite precisão ao criar ou editar desenhos e imagens, condição fundamental para ilustradores, por exemplo.

A edição vetorial é o processo pelo qual se pode trabalhar em imagens construídas a partir de formas básicas como curvas, polígonos, entre outros. Diferentemente de se trabalhar com bitmap (mapa de bits), os vetores permitem manipulações sem que a qualidade da arte seja comprometida, ou seja, é possível manter a escala da ilustração.

Além disso, esses gráficos requerem menor esforço computacional, como processamento e espaço, o que os tornam mais utilizados. É possível ainda isolar objetos e áreas da imagem facilitando o trabalho de tratamento de objetos de forma independente. Um exemplo dessa aplicação é a criação de vídeos em motion graphics, que é a técnica de design gráfico que combina animação, design e linguagem para criar conteúdo visual dinâmico.

As imagens em computadores são, comumente, representadas por matrizes que contém em cada ponto intensidade diferente de cores vermelho, verde e azul. Porém alterar uma imagem com essa estrutura causará uma destruição da originalidade do objeto. Um exemplo inicial são as imagens com dados que são formados por pixel, na figura abaixo são exibidos os percentuais de cada cor primária por pixel.

Figura 32 – Imagem bitmap ampliada



Fonte: Google imagens. (Acesso em 11.01.2023)

Os gráficos de vetor trabalham com relações matemáticas entre cada ponto e trajeto, quando são conectados podemos ver a imagem. Os gráficos vetoriais são de grande importância para os designers se eles visam mudar a cor específica na imagem, não perder a resolução ao mudar o tamanho de uma imagem e muitos outros benefícios úteis. Podemos verificar na imagem abaixo a diferença de formatos.

Figura 33 – Imagem Matricial x Imagem Vetorial



Fonte: Google imagens. (Acesso em 11.01.2023)

Uma desvantagem para o modelo vetorial de edição de imagens é que os dados vetoriais normalmente são armazenados como longas listas de coordenadas de nós e de vértices de fácil representação computacional, mas de difícil edição por parte do usuário. Conhecer o modo de ler e de operar um arquivo de dados pode demandar tempo, e não ser trivial.

No próximo tópico apresentaremos a utilização da ferramenta Bézier presente em alguns softwares de computador, bem como o resultado de sua aplicabilidade.

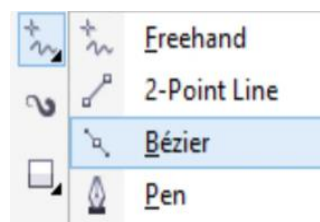
5.2 Aplicações da Ferramenta Bézier em Alguns Softwares e em Games

As situações mais presentes para a aplicabilidade da ferramenta Bézier estão relacionadas ao processo de vetorização das imagens. Listamos alguns programas de computador que utilizam a ferramenta e o resultado de sua prática.

5.2.1 Corel Draw

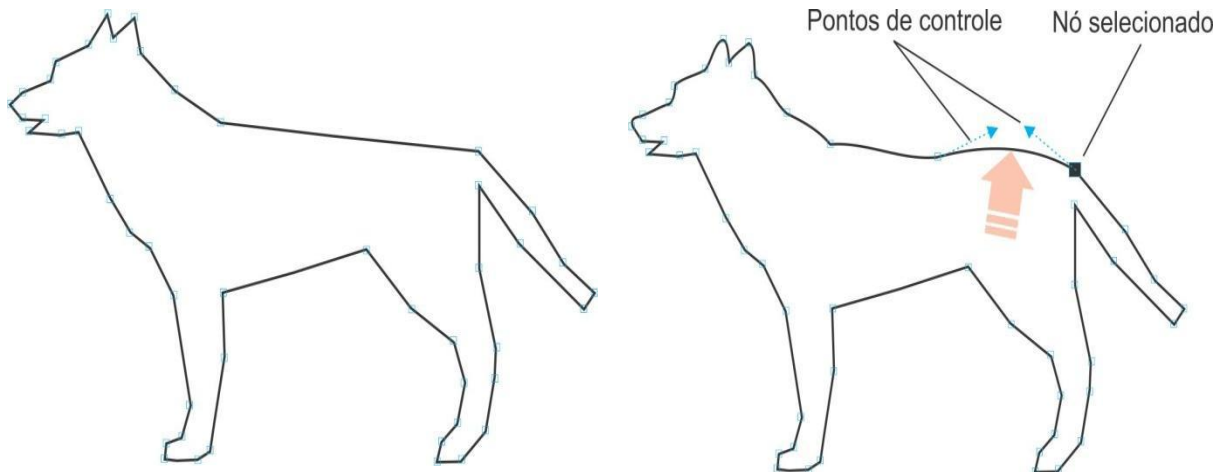
O CorelDraw é um programa de desenho vetorial bidimensional para design gráfico desenvolvido pela Corel Corporation, possibilita a criação e a manipulação de vários produtos, como por exemplo: desenhos artísticos, publicitários, logotipos, capas de revistas, livros, etc.

Figura 34 – Ícone da Ferramenta Bézier no CorelDraw



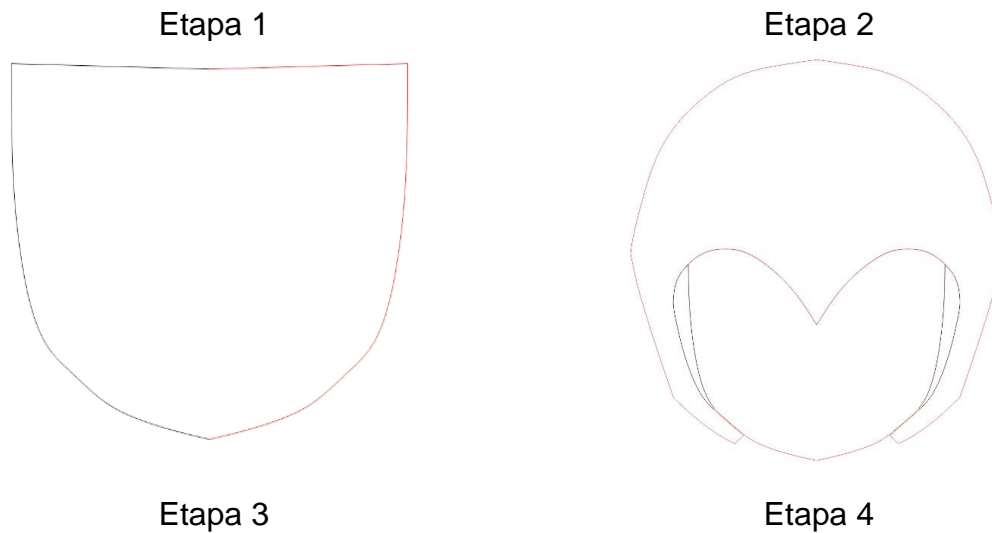
Fonte: elaborado pelo autor.

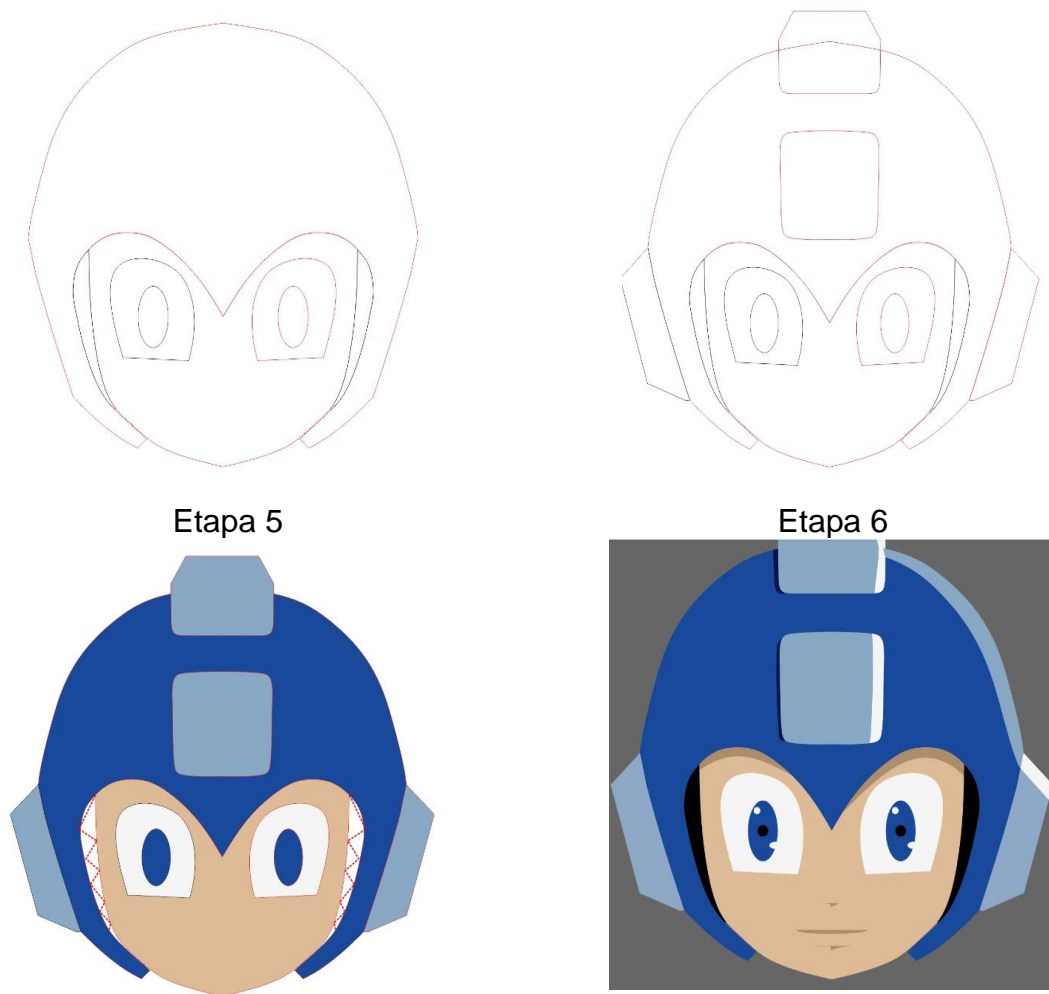
Na imagem a seguir é possível verificar a utilidade prática da ferramenta. No lado esquerdo a imagem é composta apenas por linhas retas. No lado direito podemos verificar a imagem com contornos mais suavizados. O procedimento consiste em arrastar a linha ou um dos pontos de controle com o mouse para encurvar a linha. Quando satisfeito com a curvatura da linha, selecione o nó seguinte e repita a operação.

Figura 35 – Aplicação da Ferramenta Bézier I

Fonte: Site: <http://cursocorelgratis.blogspot.com/>. (Acesso em 11.01.2023)

Outro exemplo do uso da ferramenta é a construção de imagens e objetos como descreveremos nos passos a seguir. Mostraremos os passos para desenvolver a representação de uma animação.

Figura 36 – Aplicação da Ferramenta Bézier II



Fonte: Site: <https://davidcreator.com/>. (Acesso em 11.01.2023)

5.2.2 Adobe Photoshop

Photoshop é um software da Adobe Inc. usado para edição de imagens, criação de arte digital, design gráfico e animações. A multiplicidade de usos do software faz com que o Photoshop seja uma ferramenta de trabalho para diversos tipos de profissionais. O programa pode ser utilizado por designers gráficos e web, fotógrafos, artistas, entre outros que necessitam de ferramentas para edição e criação de imagens digitais.

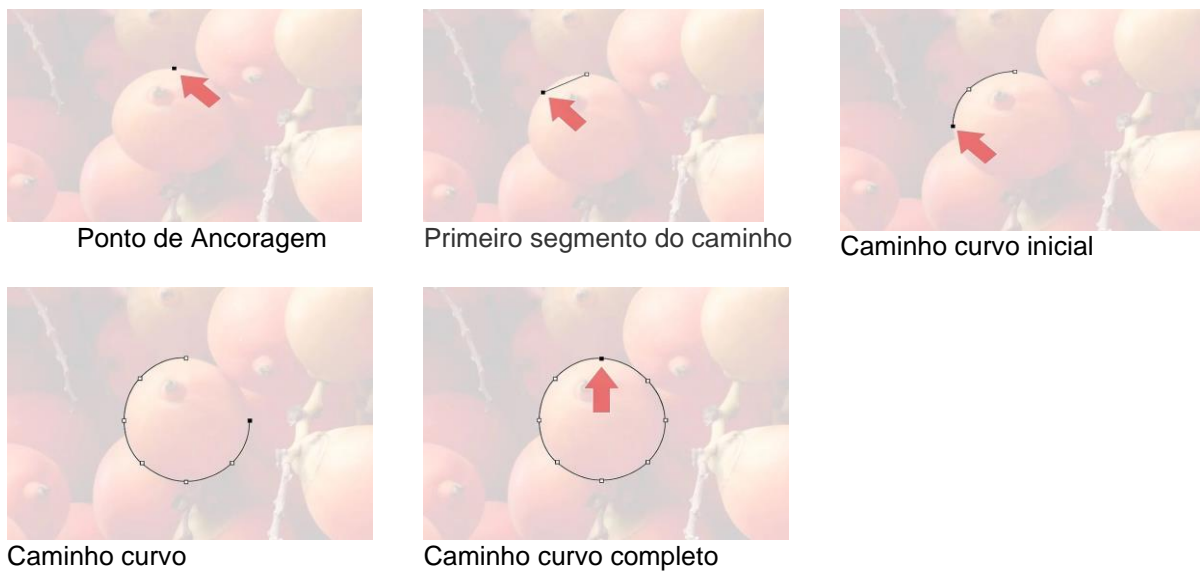
O Photoshop oferece múltiplas ferramentas de Bézier para cada necessidade criativa, são elas:

- I. A ferramenta Bézier de curvatura permite desenhar segmentos curvos e retos de maneira intuitiva.

- II. A ferramenta Bézier padrão permite desenhar segmentos curvos e retos com grande precisão.
- III. A ferramenta Traço sensível ao conteúdo permite automatizar o processo para traçar imagens.

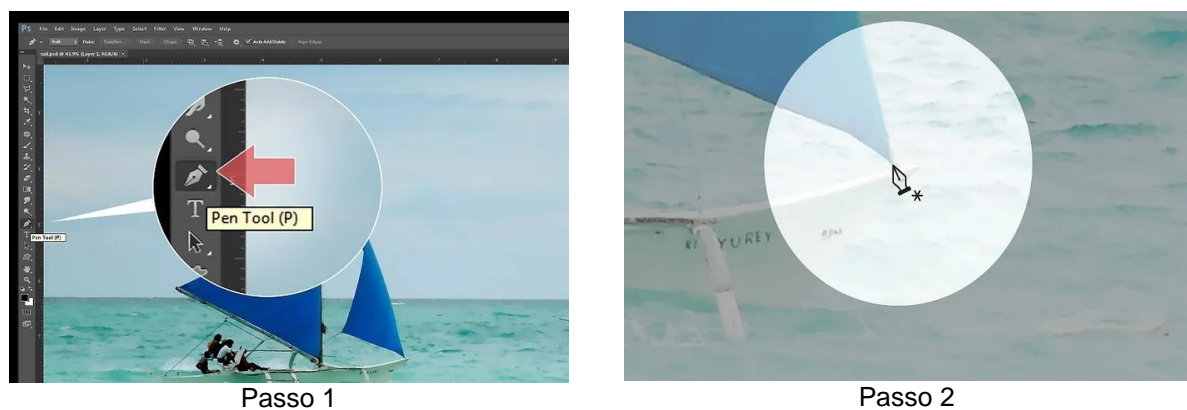
Abaixo algumas figuras que ilustram cada aplicação de Bézier. Cada ferramenta possui, como dito, especificações distintas, porém todas atuam com o mesmo princípio em relação a suavização dos contornos das imagens.

Figura 37 – Ferramenta Bézier de curvatura



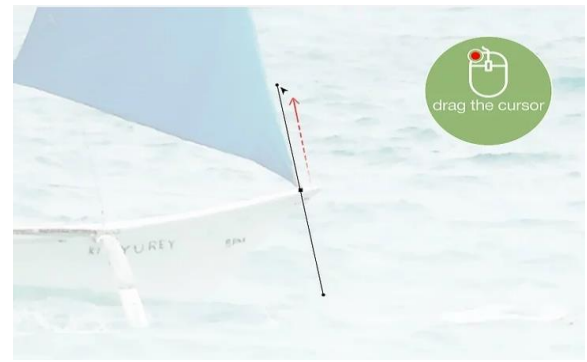
Fonte: <https://pt.wikihow.com/Desenhar-Linhas-Curvas-no-Photoshop>. (Acesso em 11.01.2023)

Figura 38 – Ferramenta Bézier padrão

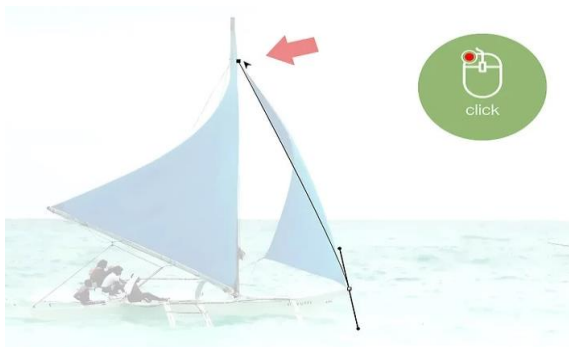




Passo 3



Passo 4



Passo 5



Passo 6



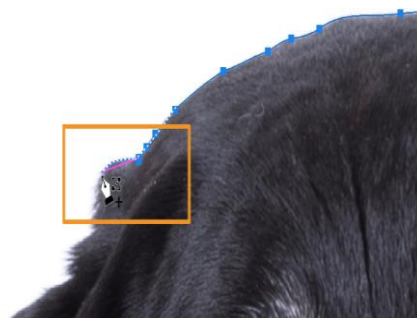
Passo 7

Fonte: <https://pt.wikihow.com/Desenhar-Linhas-Curvas-no-Photoshop>. (Acesso em 11.01.2023)

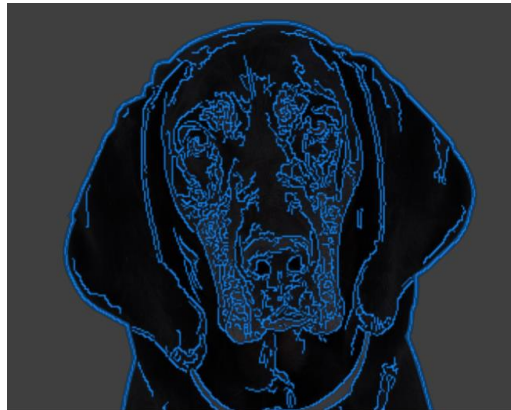
Figura 39 – Ferramenta Traço Sensível ao Conteúdo



Seção realçada



Demarcador



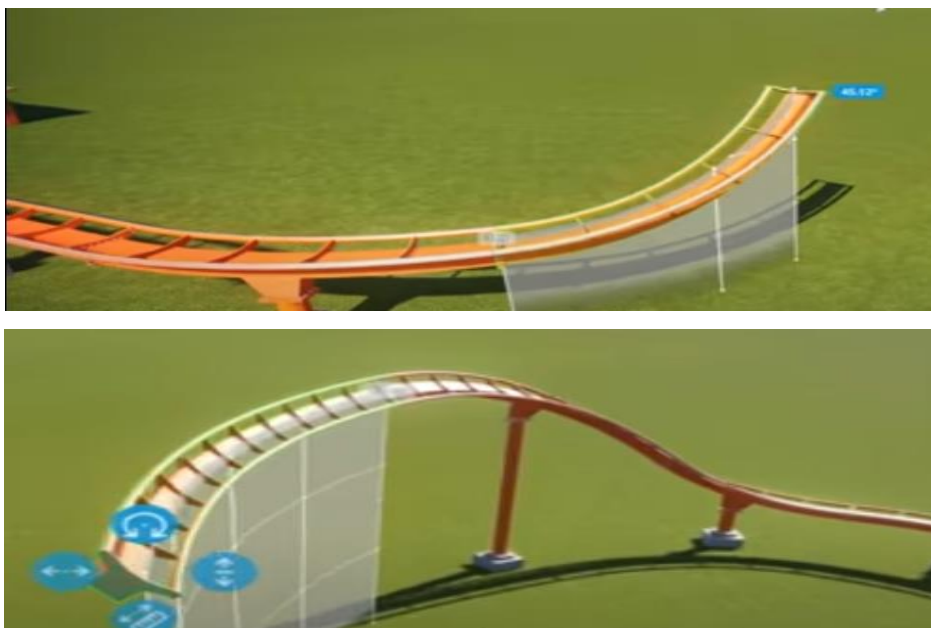
Controle deslizante Detalhe

Fonte: <https://helpx.adobe.com/br/photoshop/using/drawing-pen-tools.html>. (Acesso em 11.01.2023)

5.2.3 Aplicação da Ferramenta Bézier em Jogos e Animações

Curvas de Bézier e, mais geralmente, splines, são onipresentes em jogos. Em geral encontramos em quaisquer animações. A figura abaixo mostra a construção de um caminho de uma montanha-russa ao longo de um jogo de videogame. Esses tipos de caminhos estão por toda parte, não apenas para animações, mas também modelos 3D, nesse caso percebemos se possível manipular dobrando, torcendo ou estendendo-a.

Figura 40 – Construção de Animações Para jogos I



Fonte: <https://www.youtube.com/watch?v=aVwxzDHniEw>. (Acesso em 11.01.2023)

Figura 41 – Construção de Animações Para jogos II



Fonte: <https://www.youtube.com/watch?v=aVwxzDHniEw>. (Acesso em 11.01.2023)

6 PRODUTO EDUCACIONAL

Há muitas opções para criar um produto educacional que envolva o tema curvas de Bézier. O importante é que possamos definir um objetivo claro para o produto, a fim de torná-lo interessante e atraente para o público-alvo e utilizar recursos eficazes para ensinar e explorar o conceito. Nesse contexto colocamos como exemplos as seguintes opções:

Um aplicativo poderia fornecer uma interface amigável que permita que os usuários escolham o número de pontos de controle e ajuste sua posição e orientação. O aplicativo também poderia mostrar o resultado em tempo real, permitindo que os usuários vejam como cada mudança afeta a forma final da curva.

Outra ideia seria um jogo que usa curvas de Bézier como parte do gameplay. O jogo poderia desafiar os jogadores a traçar a curva correta para alcançar um objetivo específico, como mover um personagem para um determinado ponto no mapa ou coletar um item específico.

Também é possível criar um curso online que explique as curvas de Bézier em detalhes, incluindo seu histórico, matemática envolvida e exemplos de aplicação prática. O curso pode incluir vídeos, animações e exercícios práticos para ajudar os alunos a entender melhor o conceito e como usá-lo.

É fato que os modelos acima podem apresentar certa complexidade na sua criação e execução. Desse modo propomos como atividade uma ação que permita correlacionar o ensino de polinômios e a aplicação das curvas de Bézier para ilustrar sua utilização direta na edição de imagens para redes sociais. Alguns alunos não têm a percepção do potencial que os polinômios apresentam na solução de tarefas recorrentes de seus cotidianos. Sugerimos algo que possa ser aplicado em sala de aula e que ao fim possa despertar um interesse em relação ao assunto.

O ensino de polinômios é uma parte fundamental do currículo de matemática nas escolas. Polinômios são uma ferramenta matemática extremamente útil e versátil, e o estudo deles ajuda os alunos a desenvolver habilidades importantes, como a capacidade de modelar e resolver problemas quantitativos. Eles são amplamente utilizados em muitas áreas da matemática, ciência e engenharia, como cálculo, álgebra linear, estatística, física e engenharia, que frequentemente usam polinômios como uma ferramenta fundamental.

Na escola, o estudo dos polinômios geralmente começa com o aprendizado sobre as propriedades básicas, como grau, coeficientes e termos. Em seguida, os alunos aprendem sobre como manipular e simplificar polinômios usando as propriedades de adição, subtração, multiplicação e divisão. Eles também aprendem sobre como fatorar polinômios, o que envolve a decomposição de um polinômio em fatores menores.

Uma vez que os alunos dominem esses conceitos básicos, eles podem começar a aplicar os polinômios a problemas do mundo real. Por exemplo, eles podem usá-los para modelar fenômenos físicos, como movimento, crescimento populacional e dinâmica de sistemas, ou para resolver problemas de matemática financeira, como juros compostos e investimentos.

Dessa forma, como as curvas de Bézier são definidas por polinômios, podemos inserir nesse contexto seu uso para edição de imagens para redes sociais. É muito comum nos dias de hoje, encontrar pessoas que utilizam aplicativos de edição de fotos para melhorar suas imagens antes de publicá-las em redes sociais como o Instagram, Snapchat, TikTok, entre outras. Esses aplicativos permitem aos usuários melhorar a qualidade das fotos, corrigir imperfeições e adicionar efeitos para torná-las mais atraentes para o público.

Na edição de imagens, as curvas Bézier são usadas principalmente para criar máscaras e seleções precisas. Ao desenhar uma seleção com a ferramenta Pen, (uma ferramenta disponível em diversos programas de edição de imagem, como Adobe Photoshop, Adobe Illustrator e CorelDraw) o usuário pode ajustar as curvas Bézier para criar uma seleção suave e precisa em torno de um objeto. A ferramenta Pen é geralmente usada em conjunto com a ferramenta Path (também disponível em programas de edição de imagem como Adobe Photoshop e Adobe Illustrator), que permite que o usuário edite as curvas e formas criadas com a ferramenta Pen.

Abaixo detalharemos um plano de aula que orienta uma abordagem para a utilização de edição de imagens, usando a ferramenta Bézier, como meio para o ensino de polinômios para alunos do ensino médio.

6.1 Plano de Aula

Título da aula: Polinômios e Curvas de Bézier

Objetivos:

- a) Apresentar os conceitos básicos de polinômios;
- b) Identificar as propriedades das curvas de Bézier e sua relação com os polinômios;
- c) Explorar a aplicação das curvas de Bézier na representação gráfica de polinômios.

Recursos didáticos:

- a) Computadores com software de edição de imagens (Adobe Illustrator, Inkscape, CorelDraw, etc.);
- b) Projetor para apresentação;
- c) Exemplos de imagens que utilizam curvas de Bézier.

Desenvolvimento:

1) Preparação:

Revisão dos conceitos básicos de álgebra (equações, variáveis, etc.);

Perguntas para a turma sobre o que eles já sabem sobre polinômios.

2) Exposição:

Apresentação dos conceitos básicos de polinômios (termos, graus, etc.);

Identificação das propriedades dos polinômios.

3) Atividade:

Os alunos irão criar uma curva de Bézier usando o software de edição de imagens;

Os alunos irão explorar a manipulação dos pontos de controle para criar diferentes curvas.

Os alunos irão criar um projeto gráfico utilizando a ferramenta aprendida na exposição;

O projeto pode ser um cartaz, banner, capa de revista, entre outros.

4) Discussão:

Discussão em grupo sobre as atividades realizadas;

Sanar possíveis dúvidas.

5) Aplicação:

Os alunos irão pesquisar exemplos de imagens que utilizam curvas de Bézier para representar polinômios e apresentar para a turma;

Os alunos irão discutir a aplicação das curvas de Bézier na representação gráfica de polinômios.

6) Encerramento:

Recapitulação dos principais conceitos abordados na aula;

Reflexão sobre a importância da representação gráfica de polinômios na compreensão dos conceitos.

Avaliação:

A avaliação será feita por meio da participação dos alunos nas atividades propostas e nas discussões em grupo;

Será verificado se os alunos conseguiram criar e manipular curvas de Bézier no software de edição de imagens para representar polinômios;

Será avaliado o entendimento dos alunos sobre a aplicação das curvas de Bézier na representação gráfica de polinômios.

Observações:

O tempo de cada etapa pode ser adaptado conforme a necessidade;

O plano de aula deve ser adaptado à faixa etária e ao nível de conhecimento dos alunos;

O professor pode acrescentar outros recursos didáticos ou atividades que julgar pertinentes.

7 CONCLUSÃO

Com base nas informações apresentadas neste trabalho, podemos concluir que as curvas de Bézier são uma ferramenta valiosa e poderosa para modelar formas complexas e suaves no mundo da computação gráfica. São definidas matematicamente através de pontos de controle, que são usados para determinar a direção e o comprimento das curvas. Essas curvas têm sido amplamente utilizadas em aplicações de design gráfico, jogos, animação e outras áreas.

As curvas de Bézier permitem uma grande flexibilidade na criação de curvas, permitindo aos usuários criar uma variedade infinita de formas e designs. Com a possibilidade de ajustar os pontos de controle, essas curvas permitem que os usuários criem formas suaves e precisas, dando-lhes controle total sobre o design.

Além disso, são altamente eficientes em termos de desempenho, o que as torna ideais para aplicações em tempo real, como jogos e animação.

No geral, as curvas de Bézier se mostra uma ferramenta valiosa para quem trabalha com computação gráfica. A flexibilidade, eficiência e precisão dessas curvas as tornam uma escolha popular para a criação de designs complexos e suaves. Com o avanço contínuo da tecnologia, é provável que continuem a ser uma ferramenta valiosa para designers e desenvolvedores de jogos e animação por muitos anos.

REFERÊNCIAS

- ALENCAR, Hilário; SANTOS, Walcy; NETO, Gregório Silva. **Geometria diferencial de curvas no R²**. Rio de Janeiro: SBM, 2020. 428 p. v. 4.
- ALMEIDA, Adriano Pinho; CAVALCANTI, Cleber Araújo. Curvas De Bézier e Algumas Aplicações. **Revista Multidebates**, Palmas, v. 4, n. 5, 2020. Disponível em: <https://revista.faculdadeitop.edu.br/index.php/revista/article/view/301>. Acesso em: 7 jan. 2023.
- ALVES, Alane Da Rocha. **Uma sequência didática para o ensino de curvas de Bézier usando Geogebra**. 2021. 145f. Dissertação (Mestrado em Matemática) - Universidade Federal de Alagoas, Alagoas, 2021. Disponível em: <https://www.repositorio.ufal.br/bitstream/123456789/8691/1/Uma%20sequ%C3%Aancia%20did%C3%A1tica%20para%20o%20ensino%20de%20curvas%20de%20B%C3%A9zier%20usando%20GeoGebra.pdf>. Acesso em: 7 jan. 2023.
- ANTON, Howard; RORRES, Chris. **Álgebra linear com aplicações**. 10. ed. Porto Alegre: Bookman, 2012.
- BIEZUNER, Rodney Josué; DE JESUS, Breno Felipe Rodrigues. Curvas de Bézier. In: SEMINÁRIO DE INICIAÇÃO CIENTÍFICA, 1., 2014, Belo Horizonte. **Anais...** Belo Horizonte: UFMG, 2014.
- BUSS, Samuel R. **3D Computer Graphics: A Mathematical Introduction with OpenGL**. 2. ed. [S. l.: s. n.], 2003.
- CAD, CAE E CAM: Qual a diferença entre eles?. [S. l.], 16 mar. 2017. Disponível em: <https://www.4ieng.com.br/single-post/2017/03/16/cad-cae-e-cam-qual-a-diferenca-entre-eles>. Acesso em: 11 jan. 2023.
- DELGADO, Jorge; FRENSEL, Katia; CRISSAFF, Lhaylla. **Geometria Analítica**. 2. ed. Rio de Janeiro: SBM, 2017. 373 p. v. 11.
- DE OLIVEIRA, M.C.F. **Curvas e Superfícies**. 2018. Disponível em: <http://wiki.icmc.usp.br/images/1/14/CurvasSuperficiesLivres.pdf>. Acesso em: 17 jan. 2023.
- DO CARMO, Manfredo Perdigão. **Geometria Diferencial de Curvas e Superfícies**. Rio de Janeiro: SBM, 2005.
- DYLEWSKI, Vanusa. Recuperando o traço de uma curva utilizando integração numérica e o teorema fundamental das curvas planas. In: ENCONTRO REGIONAL DE ESTUDANTES DE MATEMÁTICA DA REGIÃO SUL, 20., 2014, Bagé, **Anais....** Bagé: UNIPAMPA, 2014. Disponível em: https://eventos.unipampa.edu.br/eremat/files/2014/12/CC_Dylewski_032.869.680-30.pdf. Acesso em: 7 jan. 2023.
- FARIN, Gerald. **Curves and Surfaces for Computer Aided Design**. 4. ed. [S. l.: s. n.], 1997.

FLORES, Marcia. **Hélices, Curvas de Bertrand e Superfícies Regradas**. 2012. 124f. Dissertação (Mestrado em Matemática) - Universidade Federal de Santa Maria, Santa Maria, 2012. Disponível em:

<https://repositorio.ufsm.br/handle/1/9973#:~:text=Uma%20h%C3%A9lice%20circular%20%C3%A9%20caracterizada,s%20%E2%88%88%20I%20s%C3%A3o%20iguais>.

Acesso em: 7 jan. 2023.

MADUREIRA, Rodrigo Lopes Rangel. **Algoritmos de interseções de curvas de bézier com uma aplicação à localização de raízes de equações**. 2013. 145f.

Dissertação (Mestrado em Informática) – Universidade Federal do Rio Janeiro, Rio de Janeiro, 2013. Disponível em: <http://objdig.ufrj.br/15/teses/823525.pdf>. Acesso em: 7 jan. 2023.

MICHEL, Alain P. La carrière de Pierre Bézier chez Renault (1933-1975): La biographie, mesure du contraste entre les projets d'un ingénieur et la conduite d'une entreprise. **Artefact**: Techniques, Histoire et Sciences Humaines, Open Edition Journals, França, v. 13, p. 213-242, 2020. Disponível em:

<https://journals.openedition.org/artefact/67111#ftn19>. Acesso em: 11 jan. 2023.

PALUSZNY, Marco; PRAUTZSCH, Hartmut; BOEHM, Wolfgang. **Métodos de Bézier y B-splines**. Alemanha: Springer Verlag Berlin Heidelberg, 2005. 303 p.

RUGGIERO, Márcia A. Gomes; LOPES, Vera Lúcia da Rocha. **Cálculo Numérico: Aspectos Teóricos e Computacionais**. 2. ed. São Paulo: Pearson, 1997.

SOLID MODELING ASSOCIATION. **Paul de Faget de Casteljaou, ganhador do Prêmio Pierre Bézier 2012**. [S. l.]: Solid Modeling Association, 2012. Disponível em: <http://solidmodeling.org/awards/bezier-award/paul-de-faget-de-casteljaou/>. Acesso em: 11 jan. 2023.

TENENBLAT, Keti. **Introdução a Geometria Diferencial**. 2. ed. São Paulo: Blucher, 2008.

WANGENHEIM, A.V.; WAGNER, H.M. Curva de Bézier. In: WANGENHEIM, A.v.; WAGNER, H.M. **Curvas paramétricas**. [S. l.: s. n.], [2005]. cap. 4, p. 87-114.

Disponível em: <http://www.inf.ufsc.br/~aldo.vw/grafica/CG4.pdf>. Acesso em: 17 jan. 2023.