

Relatório Técnico

Curvas e Superfícies

Rafael Viana de Carvalho
Ernesto Prado Lopes
Julio Tadeu Carvalho da Silveira
Eliana P. L. Aude

rvcarvalho@ufrj.br

lopes@cos.ufrj.br

julio@nce.ufrj.br

elaude@nce.ufrj.br

03 de dezembro de 2007

Resumo

Este trabalho apresenta uma exposição dos conceitos teóricos da computação gráfica referente às curvas e superfícies. Inicialmente é feita uma apresentação da conceituação de curvas e superfícies da geometria e da geometria diferencial e em seguida discute-se a adaptação destes conceitos às necessidades da computação gráfica. Mais precisamente, são apresentadas as diversas formas de representação das curvas e superfícies como curvas e superfícies paramétricas polinomiais por partes.

Lista de Figuras

3.1	Junção de segmentos interpolados	12
3.2	Interpolação dos polinômios das funções de mistura	13
3.3	continuidade no ponto de junção de curvas Hermite	16
3.4	polinômio de mistura que influencia a Bézier cúbica	17
3.5	Curva de Bézier contida no fecho convexo formado por seus pontos de controle.	18
3.6	Ajustador (<i>Blending function</i>) B_{kd} para $d = 0$	20
3.7	Ajustador (<i>Blending function</i>) B_{kd} para $d = 1$	20
3.8	Ajustador (<i>Blending function</i>) B_{kd} para $d = 2$	21
3.9	Ajustador (<i>Blending function</i>) B_{kd} para $d = 3$	21
3.10	<i>Blending functions</i> consideradas entre dois nós consecutivos, $d = 3$	22
3.11	Ajustador (Curva NURBS com variação do peso associado ao ponto de controle $\mathbf{p}(1)$)	24
4.1	Trecho de superfície	26
4.2	Trecho de superfície B-Spline	28

Sumário

1	Introdução	3
2	Representação de Curvas e Superfícies na Geometria e na Geometria Diferencial	5
2.1	Representação explícita	5
2.2	Representação implícita	6
2.3	Representação paramétrica	6
3	Tipos de Curvas	8
3.1	Curvas Parametrizadas Cúbicas	8
3.1.1	Interpolação	10
3.1.2	Curvas Hermite	14
3.1.3	Curvas Bézier	15
3.1.4	Curvas B-splines	17
3.1.5	Curvas NURBS	22
4	Tipos de Superfícies	25
4.1	Superfícies Parametrizadas	25
4.1.1	Superfícies de Bézier	26
4.1.2	Superfícies B-Splines	27
4.1.3	Superfícies NURBS	27
5	Conclusão	29

Capítulo 1

Introdução

A Computação Gráfica é um exemplo de ciência que depende da capacidade de processamento dos computadores. Através dela, é possível representar, graficamente, situações complexas em diversas áreas da vida real. Ela se fundamenta no uso do computador para transformar regras de geometria e física em imagens que têm significado para seres humanos (Glassner, 1995). Seu uso estende-se a aplicações científicas, como simulação de fenômenos físicos, visualização científica e aplicações domésticas como: filmes, comerciais, diversão eletrônica, dentre outras.

No passado, as representações de objetos do mundo real eram feitas através de estruturas simples como os *wire frame*¹ que davam ao usuário a impressão do objeto, mas obviamente insatisfatória. Em busca de uma melhor representação gráfica, os algoritmos tiveram que evoluir para possibilitar a representação completa das superfícies.

A modelagem de objetos deformáveis tem sido alvo de estudo pela comunidade de computação gráfica por mais de duas décadas. Ela tem evoluído através das aplicações, destacando-se aí a pura arte, que busca gerar imagens e animações em cenas virtuais de fina beleza e realismo. Os modelos vem sendo criados e usados nas mais diversas áreas. Poderíamos citar como exemplos: a manufatura e projeto de vestuário (animação de vestuário sobre figuras humanas, comportamento de tecidos sobre outros objetos ou suspensos em ganchos e cabides, etc); nas animações de expressões faciais de figuras humanas e de animais; na análises de imagens médicas (incluindo segmentação, representação, registro e captura de movimento da forma); na simulação dos movimentos dos músculos; em simulações cirúrgicas e treinamento de pessoal na área de cirurgia, etc... A exigência de processamento em tempo real e modelagem fisicamente realística é uma constante em quase todos esses exemplos. Para sumários do uso de modelos deformáveis em

¹Representação 3D de um objeto através de vértices e arestas interligados por retas ou curvas formando polígonos cujo as faces não são preenchidas ou coloridas

análises médica, recomenda-se a referência (McInerny, 1996). Aplicações e panorama de modelagem e animação de vestuários são apresentados em (McInerny, 1996; Volino, 2000; Etezemu, 2001). Para uma revisão sobre superfícies deformáveis: topologia, geometria e deformação, recomenda-se (Montagnat, 2001).

Muitos objetos do mundo real apresentam formas ou contornos suaves. Para modelar os objetos deformáveis em computação gráfica, é necessário a utilização de curvas e superfícies suaves na sua representação computacional (Angel, 2006).

Neste trabalho foi realizado um estudo dos tipos de representações de curvas e superfícies que melhor se adaptam aos ambientes tridimensionais. O conteúdo deste trabalho está disposto da seguinte forma: no capítulo 2 é feita uma apresentação rápida da conceituação de curvas e superfícies da geometria e da geometria diferencial, no capítulo 3, são apresentados os tipos de curvas paramétricas mais comumente utilizadas em computação gráfica, um pré-requisito para o entendimento das superfícies em computação gráfica, que são abordadas no capítulo 4. O último capítulo é dedicado aos comentários finais deste trabalho.

Capítulo 2

Representação de Curvas e Superfícies na Geometria e na Geometria Diferencial

Nesta seção apresentaremos rapidamente os principais conceitos concernentes a representação de curvas e superfícies na Geometria e na Geometria Diferencial. Nestas disciplinas, fundamentalmente, destacam-se três formas de representação de curvas e superfícies: a explícita, a implícita e a paramétrica.

2.1 Representação explícita

A representação explícita de uma curva ou superfície é possível quando ela pode ser expressa como o gráfico de uma função. Ou seja, como um conjunto dado por:

$$G = \{(x, y) | y = f(x)\} \subset \mathfrak{R}^m, m = n + 1 \text{ ou } 3;$$

onde f é uma função da **variável independente** $x \in \mathfrak{R}^n, n = 1, 2$.

Este tipo de representação não é adequado para os sistemas gráficos. Espera-se da representação de uma curva ou superfície, que, no mínimo, ela defina uma bijeção entre um aberto de um espaço euclidiano (no caso o R^1 ou o R^2) e parte da curva ou superfície que se quer representar. Isto é fundamental, por exemplo, para que seja possível identificar os pontos da superfície e mapear sobre eles as propriedades óticas da superfície ou curva em cada ponto. Em geral, é extremamente difícil achar os abertos e as formas funcionais explícitas que estabeleçam esta relação biunívoca cobrindo toda superfície, mesmo quando elas existem.

2.2 Representação implícita

Na representação implícita a superfície é representada pelos pontos do $(x, y) \in \mathbb{R}^{n+m}$, $n + m = 2$ ou 3 , que satisfazem a equação:

$$f(x, y) = 0 \in \mathbb{R}^m, m = 1 \text{ ou } 2.$$

Por exemplo, uma reta ou uma esfera centrada na origem podem ser representados, respectivamente, por:

$$ax + by + c = 0 \quad \text{e} \quad x^2 + y^2 + z^2 - r^2 = 0$$

Nas condições do teorema da função implícita existe sempre uma representação explícita local para uma curva ou superfície definida implicitamente. No entanto, curvas e superfícies no espaço tridimensional não são fáceis de ser representadas de forma implícita. As vantagens desta representação, quando ela é possível, são a possibilidade de representar curvas e superfícies infinitas, p.ex. um plano dado pela equação $ax + by + cz = 0$, e a facilidade para se avaliar se um ponto pertence ou não a uma curva ou a uma superfície.

2.3 Representação paramétrica

A representação paramétrica expressa, localmente, os valores das coordenadas de cada ponto de uma superfície, com relação ao espaço em que ela está imersa, como função de uma variável independente u , denominada parâmetro, definida em um aberto $U \subset \mathbb{R}^n$, onde n é a dimensão da superfície.

Por exemplo, em um espaço tridimensional, cada ponto de uma curva parametrizada pode ser representado por:

$$\mathbf{x}(u) = (x, y, z), \quad \text{onde } x = x(u), \quad y = y(u), \quad z = z(u), \quad u \in I = (a, b).$$

Na geometria diferencial, que usa o cálculo diferencial para estudar as propriedades das superfícies, restringe-se o estudos das superfícies às chamadas superfícies regulares, que são definidas como: (Carmo, 1976)

Um subconjunto $S \subset \mathbb{R}^3$ é uma superfície regular se, para cada $p \in S$, existe uma vizinhança V no \mathbb{R}^3 e uma aplicação $\mathbf{x} : U \rightarrow V \cap S$ de um aberto $U \subset \mathbb{R}^2$ sobre $V \cap S \subset \mathbb{R}^3$ tal que:

1. \mathbf{x} é C^∞ ;
2. $\mathbf{x}^{-1}: V \cap S \rightarrow U$ existe e é contínua como restrição de uma função contínua, definida num aberto do \mathbb{R}^3 contendo $V \cap S$ e a valores no \mathbb{R}^2 ;

3. $\forall q \in U$, a derivada de \mathbf{x} no ponto q , $\mathbf{x}'(q): \mathbb{R}^2 \rightarrow \mathbb{R}^3$, é injetiva.

Por esta definição uma superfície pode ser representada por um conjunto de aplicações diferenciáveis com inversas contínuas em todo seus pontos. Portanto, cada ponto da superfície pode ser referenciado de maneira única pela escolha correta de uma destas aplicações. Os domínios destas aplicações podem ser tomados sempre como sendo o conjunto aberto $(0, 1)$ ou $(0, 1) \times (0, 1)$, de acordo com a dimensão da superfície.

No caso das curvas (superfícies de dimensão 1) define-se o vetor velocidade por $\mathbf{x}'(q) = \frac{d\mathbf{x}(q)}{dq} = (x'(q), y'(q), z'(q))$, ele é tangente à curva em cada ponto. Pode-se representar $\mathbf{x}(q)$ e $\mathbf{x}'(q)$, na base do espaço em que a curva está imersa, como um vetor coluna:

$$\mathbf{x}(q) = \begin{bmatrix} x(q) \\ y(q) \\ z(q) \end{bmatrix} \quad \text{e} \quad \frac{d\mathbf{x}(q)}{dq} = \begin{bmatrix} \frac{dx(q)}{dq} \\ \frac{dy(q)}{dq} \\ \frac{dz(q)}{dq} \end{bmatrix}.$$

Na representação das superfícies (dimensão 2) se $q = (u, v)$, temos:

$$\mathbf{x}(u, v) = (x(u, v), y(u, v), z(u, v)).$$

O plano tangente em cada ponto $p = \mathbf{x}(q)$ da superfície é determinado pelos vetores linearmente independentes tangentes a superfície $\frac{\partial \mathbf{x}}{\partial u}(q)$ e $\frac{\partial \mathbf{x}}{\partial v}(q)$. Na forma matricial:

$$\mathbf{x}(u, v) = \begin{bmatrix} x(u, v) \\ y(u, v) \\ z(u, v) \end{bmatrix}, \quad \frac{\partial \mathbf{x}}{\partial u} = \begin{bmatrix} \frac{\partial x(u, v)}{\partial u} \\ \frac{\partial y(u, v)}{\partial u} \\ \frac{\partial z(u, v)}{\partial u} \end{bmatrix} \quad \text{e} \quad \frac{\partial \mathbf{x}}{\partial v} = \begin{bmatrix} \frac{\partial x(u, v)}{\partial v} \\ \frac{\partial y(u, v)}{\partial v} \\ \frac{\partial z(u, v)}{\partial v} \end{bmatrix}.$$

A representação paramétrica de curvas e superfícies é a forma mais flexível e robusta para sua modelagem computacional. Ela não apresenta as limitações inerentes às formas implícita e explícita, permitindo o uso da álgebra linear nos cálculos necessários para a concatenação de fragmentos, prolongando-os em curvas ou superfícies contínuas e suaves, como requerido em modelagem de objetos. Isto, em geral, é feito pela escolha dos pontos de junção e pela imposição de restrições aos vetores tangentes e normais de cada fragmento de curva ou superfície. Uma questão importante, que será discutida nos capítulos a seguir, são os métodos de escolha da forma aproximada das funções paramétricas que são usadas em computação gráfica.

Capítulo 3

Tipos de Curvas

Muitos objetos do mundo real apresentam formas ou contornos suaves. Para modelar tais objetos em computação gráfica, é necessário a utilização de curvas e superfícies suaves em sua representação computacional. Este tipo de modelagem é utilizado em diversas áreas como Medicina, Biologia, Química, na indústria do entretenimento, como no desenvolvimento de jogos em mundos virtuais, e nas mais diversas de aplicações CAD/CAM¹.

A representação de uma curva como uma sucessão de trecho retos pode ser suficiente para várias aplicações. No entanto, curvas e superfícies complexas normalmente demandam uma maneira mais eficiente de representação. Tal representação é normalmente mais compacta que as formas discretas. Definir uma curva que passe por um conjunto determinado de pontos é um problema de **interpolação**, enquanto a definição de uma curva que passe próximo a um conjunto determinado de pontos é um problema de **aproximação**. Este capítulo irá tratar dos tipos de curvas mais comumente utilizadas como base para a construção de superfícies.

3.1 Curvas Parametrizadas Cúbicas

A modelagem de objetos envolve a definição de curvas e superfícies que obedecem determinadas condições de posição, tangência e curvatura, expressas em um conjunto de pontos denominados pontos de controle. Grande parte dos modelos geométricos utilizados em computação gráfica são baseados em curvas e superfícies

¹**Computer Aided Design (CAD)**, ou desenho auxiliado por computador, é o nome genérico de sistemas computacionais (software) utilizados pela engenharia, geologia, arquitetura, e design para facilitar o projeto e desenho técnicos. O **Computer Aided Manufacturing (CAM)**, ou Manufatura Auxiliada por Computador, trabalham tendo como base modelos matemáticos provenientes do sistema CAD e estão ligados ao processo de produção.

cies parametrizadas que utilizam funções polinomiais, cujas principais propriedades são (Persiano, 1996):

- Polinômios de grau n admitem n raízes r_i (alguns pares podem ser complexo conjugados), e podem ser expressos como:

$$p(u) = p_0(u - r_1)(u - r_2) \dots (u - r_{n-1})(u - r_n);$$

- Polinômios possuem derivadas de qualquer ordem que são também polinômios de grau menor;
- A integral indefinida de um polinômio de grau n é um polinômio de grau $n + 1$;
- Polinômios de grau $n > 0$ são funções ilimitadas;
- Polinômios de grau n admitem no máximo $n - 1$ extremos (máximos ou mínimos) locais.

Um fator a ser considerado na representação paramétrica com funções polinomiais é a definição do grau dos polinômios. Quanto maior o grau, mais parâmetros serão necessários para definir o formato da curva, podendo torná-la mais oscilante, e exigindo mais processamento computacional. Por outro lado, polinômios de menor grau podem não fornecer parâmetros suficientes para o ajuste de sua curvatura. Uma solução comumente utilizada na computação gráfica é a utilização de polinômios cúbicos (de grau 3).

Uma curva paramétrica cúbica por partes é definida pela junção de vários segmentos de curvas cúbicas. A suavidade de toda a curva é obtida pela continuidade em seus pontos de junção e das derivadas nestes pontos.

Um segmento de curva paramétrica cúbica pode ser escrita como $\mathbf{p}(u) = (x(u), y(u), z(u))$ onde:

$$x(u) = c_{0x} + c_{1x}u + c_{2x}u^2 + c_{3x}u^3,$$

$$y(u) = c_{0y} + c_{1y}u + c_{2y}u^2 + c_{3y}u^3,$$

$$z(u) = c_{0z} + c_{1z}u + c_{2z}u^2 + c_{3z}u^3.$$

Na forma matricial o polinômio cúbico paramétrico é escrito da seguinte forma:

$$[x(u) \ y(u) \ z(u)] = [1 \ u \ u^2 \ u^3] \cdot \begin{bmatrix} c_{0x} & c_{0y} & c_{0z} \\ c_{1x} & c_{1y} & c_{1z} \\ c_{2x} & c_{2y} & c_{2z} \\ c_{3x} & c_{3y} & c_{3z} \end{bmatrix}$$

ou

$$\mathbf{p}(u) = \mathbf{c}_0 + \mathbf{c}_1 u + \mathbf{c}_2 u^2 + \mathbf{c}_3 u^3 = \sum_{k=0}^3 \mathbf{c}_k u^k = \mathbf{u}^T \cdot \mathbf{c}$$

onde

$$\mathbf{p}(u)^T = \begin{bmatrix} x(u) \\ y(u) \\ z(u) \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} 1 \\ u \\ u^2 \\ u^3 \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} \mathbf{c}_0 \\ \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{bmatrix}, \quad \text{sendo } \mathbf{c}_k = \begin{bmatrix} c_{kx} \\ c_{ky} \\ c_{kz} \end{bmatrix}^T$$

Portanto, a forma básica que expressa o segmento de curva polinomial parametrizada é dada pela equação

$$\mathbf{p}(u) = \mathbf{u}^T \cdot \mathbf{c},$$

que será usada para a gerar os pontos do segmento de curva quando o parâmetro u varia no intervalo $(0, 1)$.

Resta ainda determinar os coeficientes que compõem a matriz \mathbf{c} . Para funções polinômiais cúbicas são necessárias quatro condições, em geral expressas por

$$\mathbf{p}(u_i) = \mathbf{u}_i^T \cdot \mathbf{c}, \quad i = 0, 1, 2, 3.$$

O vetor \mathbf{u}_i é composto por quatro valores específicos do parâmetro u . Em geral, os valores $\mathbf{p}(u_i)$ são determinados pelos pontos de controle, correspondendo aos valores u_i escolhidos. Este conjunto de quatro equações corresponde a um sistema que pode ser escrito da seguinte forma

$$\mathbf{p} = \mathbf{A} \cdot \mathbf{c} \quad \Rightarrow \quad \mathbf{c} = \mathbf{A}^{-1} \cdot \mathbf{p}.$$

Esta forma básica apresenta versões específicas nos vários tipos de curvas polinômiais estudadas a seguir.

3.1.1 Interpolação

Uma curva cúbica por partes **interpolada** passa por uma sequência $\{\mathbf{p}_k\}$ de pontos de controle, onde

$$\mathbf{p}_k = \begin{bmatrix} p_{kx} \\ p_{ky} \\ p_{kz} \end{bmatrix}.$$

Como são necessárias quatro condições (equações) para estabelecer os coeficientes \mathbf{c} de cada segmento de cúbica, eles serão determinados de forma que a curva polinomial $\mathbf{p}(u) = \mathbf{u}^T \mathbf{c}$ interpole quatro pontos sucessivos \mathbf{p}_k , \mathbf{p}_{k+1} , \mathbf{p}_{k+2} e \mathbf{p}_{k+3} .

Para $u \in [0, 1]$, uma possível interpolação poderia utilizar os valores

$$u_k = 0, \frac{1}{3}, \frac{2}{3}, 1. \quad (3.1)$$

$$\begin{aligned} \mathbf{p}_0 &= \mathbf{p}(0) = \mathbf{c}_0 \\ \mathbf{p}_1 &= \mathbf{p}\left(\frac{1}{3}\right) = \mathbf{c}_0 + \frac{1}{3}\mathbf{c}_1 + \frac{1}{3}^2\mathbf{c}_2 + \frac{1}{3}^3\mathbf{c}_3 \\ \mathbf{p}_2 &= \mathbf{p}\left(\frac{2}{3}\right) = \mathbf{c}_0 + \frac{2}{3}\mathbf{c}_1 + \frac{2}{3}^2\mathbf{c}_2 + \frac{2}{3}^3\mathbf{c}_3 \\ \mathbf{p}_3 &= \mathbf{p}(1) = \mathbf{c}_0 + \mathbf{c}_1 + \mathbf{c}_2 + \mathbf{c}_3 \end{aligned}$$

Reescrevendo as equações na forma matricial, obtêm-se o sistema $\mathbf{p} = \mathbf{A}\mathbf{c}$, onde:

$$\mathbf{p} = \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix} \quad \text{e} \quad \mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & \frac{1}{3} & \left(\frac{1}{3}\right)^2 & \left(\frac{1}{3}\right)^3 \\ 1 & \frac{2}{3} & \left(\frac{2}{3}\right)^2 & \left(\frac{2}{3}\right)^3 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

A matriz dos coeficientes \mathbf{c} pode então ser obtida da seguinte forma:

$$\mathbf{p} = \mathbf{A} \cdot \mathbf{c} \quad \Rightarrow \quad \mathbf{c} = \mathbf{A}^{-1}\mathbf{p} = \mathbf{M}_I\mathbf{p}.$$

A matriz

$$\mathbf{M}_I = \mathbf{A}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -5.5 & 9 & -4.5 & 1 \\ 9 & -22.5 & 18 & -4.5 \\ -4.5 & 13.5 & -13.5 & 4.5 \end{bmatrix},$$

é chamada matriz de **interpolação geométrica**, relativa aos valores u_k em (3.1). Cada segmento de cúbica $\mathbf{p}(u) = \mathbf{u}^T \mathbf{c}$, $u \in [0, 1]$, interpola quatro pontos sucessivos.

A interpolação de todos os pontos é composta por segmentos cúbicos que interpolam subsequências de quatro pontos de controle, cada um determinando uma parte da curva cúbica por partes.

Para garantir a continuidade nos pontos de junção, os extremos de segmentos vizinhos devem ser coincidentes. Ou seja, o último ponto \mathbf{p}_k de um segmento deve coincidir com o primeiro ponto do próximo segmento. A Figura 3.1 ilustra a interpolação composta por dois segmentos de cúbicas adjacentes. Embora a curva cúbica por partes formada seja contínua, a derivada nos pontos de junção pode não existir.

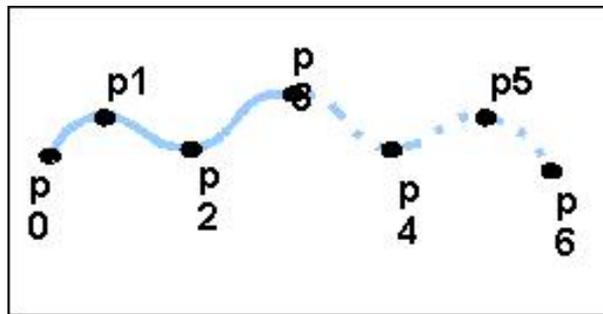


Figura 3.1: Junção de segmentos interpolados

Pode-se avaliar a influência de um ponto de controle sobre qualquer trecho do segmento que o interpola. Para isso, reescreve-se a equação

$$\mathbf{p}(u) = \mathbf{u}^T \mathbf{c} = \mathbf{u}^T \mathbf{M}_I \mathbf{p},$$

ou

$$\mathbf{p}(u) = \mathbf{b}(u)^T \mathbf{p},$$

onde $\mathbf{b}(u)^T = \mathbf{u}^T \mathbf{M}_I$.

O vetor $\mathbf{b}(u) = (\mathbf{u}^T \mathbf{M}_I)^T = \mathbf{M}_I^T \mathbf{u}$ é um conjunto de funções cúbicas conhecidas como funções de mistura (*blending functions*):

$$\mathbf{b}(u) = \begin{bmatrix} b_0(u) \\ b_1(u) \\ b_2(u) \\ b_3(u) \end{bmatrix} = \begin{bmatrix} -4.5u^3 + 9u^2 - 5.5u + 1 \\ 13.5u^3 - 22.5u^2 + 9u \\ -13.5u^3 + 18u^2 - 4.5u \\ 4.5u^3 - 4.5u^2 + u \end{bmatrix}.$$

Reescrevendo $\mathbf{p}(u)$, obtêm-se:

$$\mathbf{p}(u) = b_0(u)\mathbf{p}_0 + b_1(u)\mathbf{p}_1 + b_2(u)\mathbf{p}_2 + b_3(u)\mathbf{p}_3 = \sum_{i=0}^3 b_i(u)\mathbf{p}_i.$$

Cada $b_i(u)$ expressa a influência do seu respectivo ponto de controle sobre a curva gerada. Pode-se depreender da figura 3.2, que quando u distancia-se de $\frac{i}{3}$ (para $i = 0, 1, 2, 3$) o ponto \mathbf{p}_i diminui sua influencia sobre a curva. As funções de misturas são polinômios de grau 3, portanto podem ser escritas como produtos envolvendo suas raízes. Observa-se que as raízes de cada b_i coincidem com os valores u_k definidos em (3.1).

$$\begin{aligned} b_0(u) &= -\frac{9}{2}\left(u - \frac{1}{3}\right)\left(u - \frac{2}{3}\right)(u - 1), \\ b_1(u) &= \frac{27}{2}u\left(u - \frac{2}{3}\right)(u - 1), \\ b_2(u) &= -\frac{27}{2}u\left(u - \frac{1}{3}\right)(u - 1), \\ b_3(u) &= \frac{9}{2}u\left(u - \frac{1}{3}\right)\left(u - \frac{2}{3}\right). \end{aligned}$$

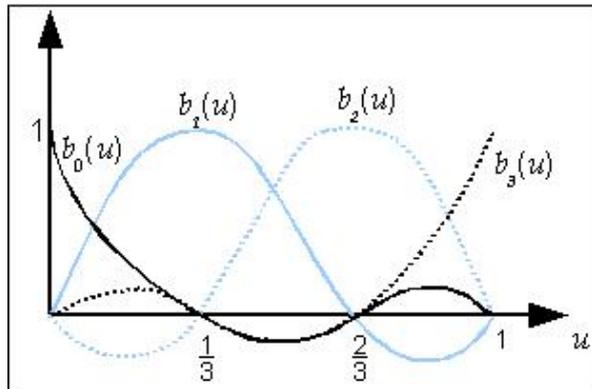


Figura 3.2: Interpolação dos polinômios das funções de mistura

3.1.2 Curvas Hermite

Para resolver o problema da não existência da derivada nos pontos de junção dos segmentos de cúbica no método anterior Hermite propôs usar apenas dois pontos de controle. As quatro condições necessárias para determinação dos coeficientes de \mathbf{c} deveriam ser dadas pela interpolação do segmento de cúbica nos pontos de entrada e saída da curva, de forma que os vetores tangentes à curva neste pontos tivessem valores pré-estabelecidos. Notaremos $(\mathbf{p}_0, \mathbf{p}_3)$ e $(\mathbf{p}'_0, \mathbf{p}'_3)$, respectivamente, os pontos de entrada e saída do segmento de cúbica e a derivada nestes pontos.

Considerando $u \in [0, 1]$ os pontos extremos são $\mathbf{p}(0)$ e $\mathbf{p}(1)$ e os vetores tangentes são determinados da seguinte forma

$$\mathbf{p}'(u) = \begin{bmatrix} \frac{dx}{du} \\ \frac{dy}{du} \\ \frac{dz}{du} \end{bmatrix} = \mathbf{c}_1 + 2u\mathbf{c}_2 + 3u^2\mathbf{c}_3.$$

Logo, as condições de Hermite são:

$$\begin{aligned} \mathbf{p}_0 &= \mathbf{p}(0) = \mathbf{c}_0, \\ \mathbf{p}_3 &= \mathbf{p}(1) = \mathbf{c}_0 + \mathbf{c}_1 + \mathbf{c}_2 + \mathbf{c}_3, \\ \mathbf{p}'_0 &= \mathbf{p}'(0) = \mathbf{c}_1, \\ \mathbf{p}'_3 &= \mathbf{p}'(1) = \mathbf{c}_1 + 2\mathbf{c}_2 + 3\mathbf{c}_3, \end{aligned}$$

Matricialmente representados por:

$$\begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_3 \\ \mathbf{p}'_0 \\ \mathbf{p}'_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 3 \end{bmatrix} \mathbf{c},$$

onde

$$\mathbf{q} = \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_3 \\ \mathbf{p}'_0 \\ \mathbf{p}'_3 \end{bmatrix} \quad \text{e} \quad \mathbf{A}_H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 3 \end{bmatrix}.$$

A matriz dos coeficientes \mathbf{c} é obtida resolvendo-se a equação $\mathbf{c} = \mathbf{M}_H \mathbf{q}$, onde \mathbf{M}_H é chamada matriz geométrica de Hermite

$$\mathbf{M}_H = \mathbf{A}_H^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix}.$$

O segmento de cúbica é dada por $\mathbf{p}(u) = \mathbf{u}^T \mathbf{c}$, logo $\mathbf{p}(u) = \mathbf{u}^T \mathbf{M}_H \mathbf{q}$.

Na interpolação de Hermite, não só os pontos de junção de dois segmento de cúbica adjacentes são compartilhados, mas também o valor das derivadas nestes pontos. Para cada par de segmentos de cúbica adjacentes, o ponto de saída de um segmento e o vetor tangente neste ponto coincide com o ponto de entrada do próximo segmento e o vetor tangente neste ponto. Isto garante que a curva total seja de classe C^1 . Ou seja, o controle preciso das tangentes de entrada e de saída dos segmentos da curva é essencial para a "suavização" da curva total. Para isso, os vetores tangentes de entrada e saída em cada ponto de controle devem possuir a mesma direção e módulo.

As funções de mistura são obtidas reescrevendo-se $\mathbf{p}(u)$ como

$$\mathbf{p}(u) = \mathbf{b}(u)^T \mathbf{q}, \text{ onde}$$

$$\mathbf{b}(u) = \mathbf{M}_H^T \mathbf{u} = \begin{bmatrix} 2u^3 - 3u^2 + 1 \\ -2u^3 + 3u^2 \\ u^3 - 2u^2 + u \\ u^3 - u^2 \end{bmatrix}.$$

A figura 3.3 ilustra a continuidade da curva obtida pelo alinhamento das tangentes no ponto de união.

3.1.3 Curvas Bézier

A curva de Bézier foi desenvolvida por Pierre Bézier durante seus trabalhos em projetos de automóveis para a Renault francesa no início da década de 1970. Bézier baseou sua curva no método proposto por Hermite, adicionando à definição original uma metodologia para a determinação das tangentes \mathbf{p}'_0 e \mathbf{p}'_3 . Em seu método é usado mais dois pontos de controle \mathbf{p}_1 e \mathbf{p}_2 , para obter uma aproximação linear de \mathbf{p}'_0 e \mathbf{p}'_3 definida da seguinte forma:

$$\begin{aligned} \mathbf{p}'_0 &= \mathbf{p}'(0) \approx \frac{\mathbf{p}_1 - \mathbf{p}_0}{\frac{1}{3}} = 3(\mathbf{p}_1 - \mathbf{p}_0), \\ \mathbf{p}'_3 &= \mathbf{p}'(1) \approx \frac{\mathbf{p}_3 - \mathbf{p}_2}{\frac{1}{3}} = 3(\mathbf{p}_3 - \mathbf{p}_2). \end{aligned}$$

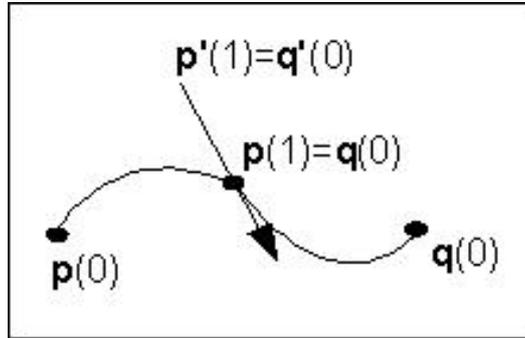


Figura 3.3: continuidade no ponto de junção de curvas Hermite

Através dessas aproximações, obtêm-se duas condições de controle, que, juntamente com as condições dos pontos extremos, determinam $\mathbf{p}(u) = \mathbf{u}^T \mathbf{c}$, onde:

$$\begin{aligned} \mathbf{p}_0 &= \mathbf{p}(0) = \mathbf{c}_0 \\ \mathbf{p}_3 &= \mathbf{p}(1) = \mathbf{c}_0 + \mathbf{c}_1 + \mathbf{c}_2 + \mathbf{c}_3 \\ \mathbf{p}'_0 &= \mathbf{p}'(0) \approx 3\mathbf{p}_1 - 3\mathbf{p}_0 = \mathbf{c}_1, \\ \mathbf{p}'_3 &= \mathbf{p}'(1) \approx 3\mathbf{p}_3 - 3\mathbf{p}_2 = \mathbf{c}_1 + 2\mathbf{c}_2 + 3\mathbf{c}_3. \end{aligned}$$

A matriz dos coeficientes \mathbf{c} é obtida através da equação $\mathbf{c} = \mathbf{M}_B \mathbf{p}$, onde \mathbf{M}_B é chamada matriz geométrica Bézier e dada por:

$$\mathbf{M}_B = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}.$$

A polinomial cúbica de Bézier é dada por $\mathbf{p}(u) = \mathbf{u}^T \mathbf{M}_B \mathbf{p}$.

As funções de mistura do método de Bézier, são polinômios de Bernstein, que se escrevem:

$$\mathbf{b}(u) = \mathbf{M}_B^T \mathbf{u} = \begin{bmatrix} (1-u)^3 \\ 3u(1-u)^2 \\ 3u^2(1-u) \\ u^3 \end{bmatrix}.$$

Os polinômios de Bernstein são dados por:

$$b_{id}(u) = \frac{d!}{i!(d-i)!} u^i (1-u)^{d-i}.$$

e tem as seguintes propriedades:

- b_{id} não tem raízes no intervalo $(0, 1)$.
- $0 < b_{id}(u) \leq 1, \forall u \in (0, 1)$
- $\sum_{i=0}^d b_{id}(u) = 1, \forall u \in [0, 1]$

A influência das funções de mistura na curva de Bézier são ilustradas na figura 3.4.

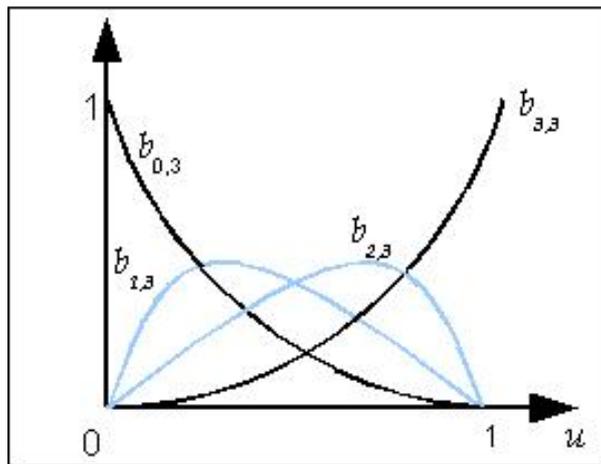


Figura 3.4: polinômio de mistura que influencia a Bézier cúbica

A curva de Bézier pode ser escrita como: $\mathbf{p}(u) = \sum_{i=0}^3 b_{i3}(u)\mathbf{p}_i$, onde b_{i3} são os polinômios de Bernstein de grau 3 e \mathbf{p}_i são os pontos de controle.

Para cada $u \in [0, 1]$, $\mathbf{p}(u)$ é uma combinação convexa dos quatro pontos de controle. Ou seja, a curva está contida no interior do fecho convexo definido por esses pontos, como mostra a figura 3.5.

3.1.4 Curvas B-splines

O termo *spline*, em inglês, se refere originalmente às tiras de metal ou madeira que eram utilizadas no projeto de embarcações, carros, etc. Pesos eram utilizados para ajustar a curvatura das *splines* às formas desejadas.

Inspirada nas splines físicas, a *Spline Cúbica Natural* é um polinômio cúbico por partes de classe C^2 que interpola todos os seus pontos de controle. Embora

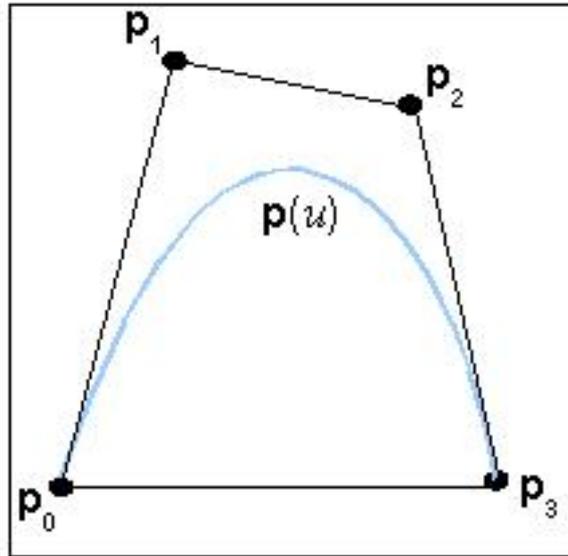


Figura 3.5: Curva de Bézier contida no fecho convexo formado por seus pontos de controle.

com suavidade maior do que as curvas Hermite e Bézier, sua principal desvantagem é que todos os pontos de controle devem ser considerados no cálculo dos coeficientes polinomiais: qualquer ajuste em um ponto de controle afeta toda a curva, exigindo muito cálculo computacional.

A B-Spline é uma versão simplificada da *Spline Cúbica Natural*, que implementa o controle local da curva. Qualquer alteração em um ponto de controle se propaga apenas para os vizinhos mais próximos. Entretanto, a B-Spline não interpola os pontos de controle, apenas permite o ajuste de proximidade da curva gerada em relação a estes pontos.

O problema geral da *Spline Natural* é: dado um conjunto de pontos de controle $\{\mathbf{p}_i\}$, $i = 0, 1, \dots, m$; deseja-se encontrar uma função suave

$$\mathbf{p}(u) = [x(u) \ y(u) \ z(u)], \quad u \in [u_{min}, u_{max}];$$

que aproxime, de alguma forma, dos pontos de controle. A partir de uma sequência não decrescente de valores u_k , $k = 0, 1, \dots, n$; denominados nós, tais que

$$u_{min} = u_0 \leq u_1 \leq \dots \leq u_n = u_{max};$$

define-se então $\mathbf{p}(u)$ como uma função polinomial de grau d , entre cada par consecutivo de nós:

$$\mathbf{p}(u) = \sum_{j=0}^d c_{jk} u^j, \quad u_k < u < u_{k+1}.$$

Para definir a spline de grau d , é necessário calcular os $n(d+1)$ coeficientes c_{jk} . Desta forma, uma cúbica polinomial ($d = 3$) é definida por $4n$ coeficientes. Para calcular os coeficientes é necessário definir tantas condições quantos coeficientes existem.

A B-Spline, assim como a curva Bézier, utiliza ajustadores (ou *blending functions*) no cálculo da curva gerada. Cada ajustador $B_{id}(u)$ expressa a influência do ponto de controle \mathbf{p}_i na geração de toda a curva, dada por:

$$\mathbf{p}(u) = \sum_{i=0}^m B_{id}(u) \mathbf{p}_i.$$

Cada função $B_{id}(u)$ é um polinômio de grau d por partes no seu intervalo de atuação ($u_{i_{min}}, u_{i_{max}}$) e é igual a zero fora dele. O intervalo de atuação é determinado pelo grau d do polinômio, conforme veremos a seguir. O nome B-Splines vem do termo *basis splines*, porque para uma determinada sequência de nós e um grau k fixado, as funções $B_{id}(u)$ formam uma base.

Um dos conjuntos de bases B-splines mais utilizadas e definida por recorrência é a proposta por **Cox-deBoor** (Foley, 1990):

$$B_{k,0}(u) = \begin{cases} 1, & u_k \leq u \leq u_{k+1} \\ 0, & \text{em outro caso} \end{cases} \quad (3.2)$$

$$B_{kd}(u) = \frac{u - u_k}{u_{k+d} - u_k} B_{k,d-1}(u) + \frac{u_{k+d+1} - u}{u_{k+d+1} - u_{k+1}} B_{k+1,d-1}(u).$$

A análise da recorrência acima permite um melhor entendimento sobre o papel de cada ponto de controle na geração da curva segmentada nos intervalos entre nós consecutivos.

A base da recorrência em (3.2), determina $B_{k,0}(u) = 1$ em (u_k, u_{k+1}) , e nulo fora do intervalo, como ilustrado na Figura 3.6

Para $d = 1$, obtêm-se:

$$B_{k,1}(u) = \frac{u - u_k}{u_{k+1} - u_k} B_{k,0}(u) + \frac{u_{k+2} - u}{u_{k+2} - u_{k+1}} B_{k+1,0}(u).$$

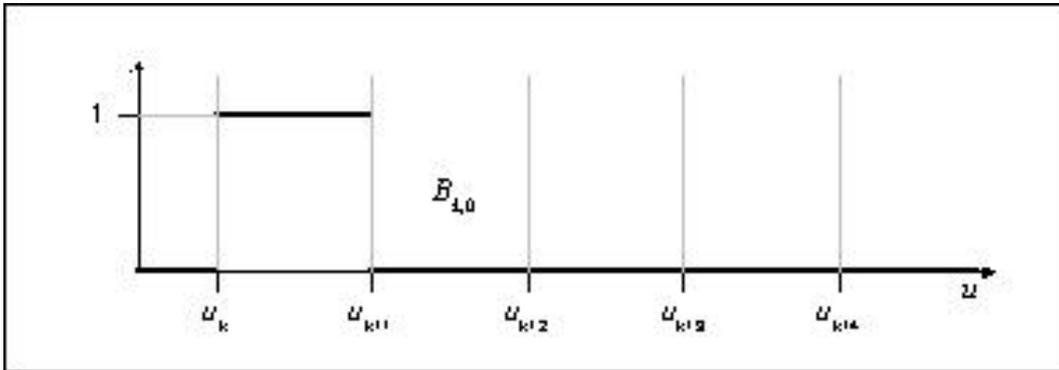


Figura 3.6: Ajustador (*Blending function*) B_{kd} para $d = 0$

O que resulta em:

$$B_{k,1}(u) = \begin{cases} \frac{u-u_k}{u_{k+1}-u_k}, & u_k \leq u \leq u_{k+1} \\ \frac{u_{k+2}-u}{u_{k+2}-u_{k+1}}, & u_{k+1} \leq u \leq u_{k+2} \\ 0, & \text{em outro caso} \end{cases}$$

A função $B_{k,1}(u)$ é ilustrada na Figura 3.7.

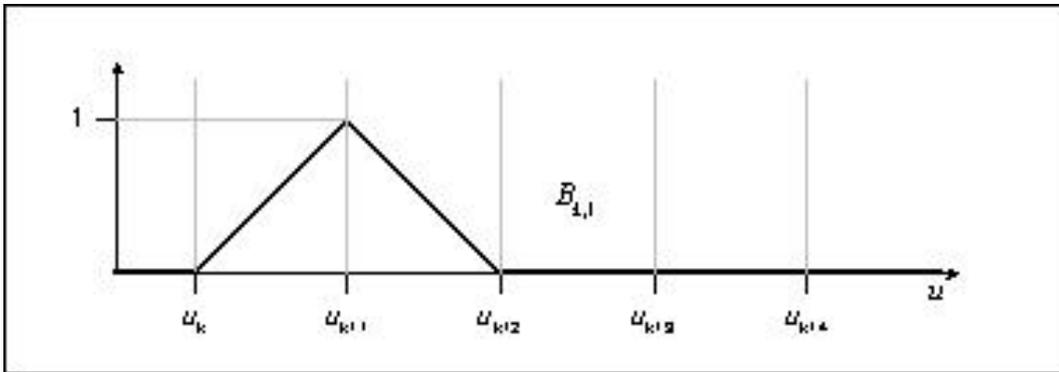


Figura 3.7: Ajustador (*Blending function*) B_{kd} para $d = 1$

Para $d = 2$:

$$B_{k,2}(u) = \frac{u - u_k}{u_{k+2} - u_k} B_{k,1}(u) + \frac{u_{k+3} - u}{u_{k+3} - u_{k+1}} B_{k+1,1}(u).$$

resultando

$$B_{k,2}(u) = \begin{cases} \frac{u-u_k}{u_{k+2}-u_k} \cdot \frac{u_{k+1}-u_k}{u_{k+1}-u_k}, & u_k \leq u \leq u_{k+1} \\ \frac{u-u_k}{u_{k+2}-u_k} \cdot \frac{u_{k+2}-u}{u_{k+2}-u_{k+1}} + \frac{u_{k+3}-u}{u_{k+3}-u_{k+1}} \cdot \frac{u-u_{k+1}}{u_{k+2}-u_{k+1}}, & u_{k+1} \leq u \leq u_{k+2} \\ \frac{u_{k+3}-u}{u_{k+3}-u_{k+1}} \cdot \frac{u_{k+3}-u}{u_{k+3}-u_{k+2}}, & u_{k+2} \leq u \leq u_{k+3} \\ 0, & \text{em outro caso} \end{cases}$$

A função $B_{k,2}(u)$ é ilustrada na Figura 3.8.

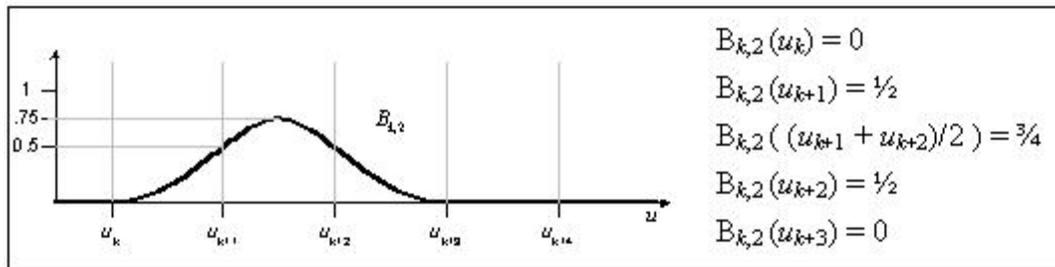


Figura 3.8: Ajustador (*Blending function*) B_{kd} para $d = 2$

Para $d = 3$, obtêm-se

$$B_{k,3}(u) = \frac{u - u_k}{u_{k+3} - u_k} B_{k,2}(u) + \frac{u_{k+4} - u}{u_{k+4} - u_{k+1}} B_{k+1,2}(u),$$

como ilustrado na Figura 3.9.

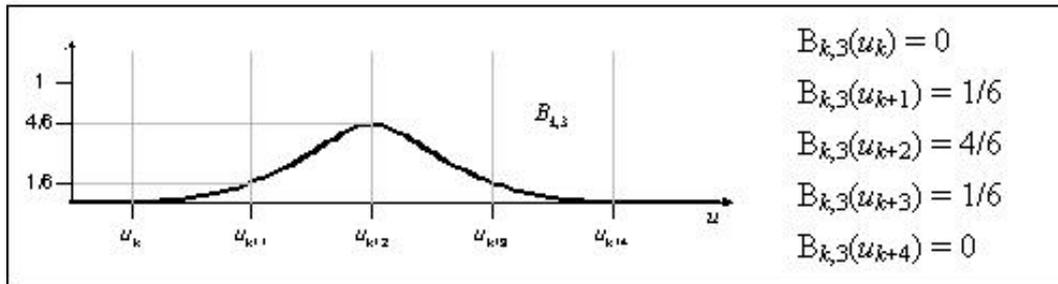


Figura 3.9: Ajustador (*Blending function*) B_{kd} para $d = 3$

A figura 3.9 mostra a influência do ponto de controle \mathbf{p}_k , determinada pela função $B_{k3}(u)$, em quatro trechos consecutivos (de $[u_k, u_{k+1}]$ a $[u_{k+3}, u_{k+4}]$), da polinomial cúbica gerada. No caso geral, a geração de um polinômio de grau d

determina a influência de cada ponto \mathbf{p}_k sobre $d + 1$ intervalos consecutivos, de $[u_k, u_{k+1}]$ a $[u_{k+d}, u_{k+d+1}]$.

Na polinomial cúbica, cada intervalo entre nós consecutivos é influenciado por quatro pontos de controle. A figura 3.10 ilustra tal situação, onde a geração da curva no intervalo $[u_k, u_{k+1}]$ leva em consideração quatro ajustadores consecutivos: $B_{k-3,d}$ a $B_{k,d}$.

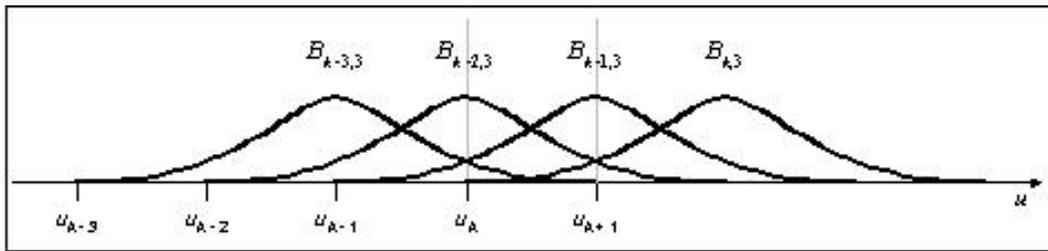


Figura 3.10: *Blending functions* consideradas entre dois nós consecutivos, $d = 3$

As bases B-splines de **Cox-deBoor** são de classe C^{d-1} nos nós e satisfazem a propriedade convexidade

$$0 \leq b_{id}(u) \leq 1 \text{ e } \sum_{i=0}^d b_{id}(u) = 1, \forall u \in [0, 1]$$

A figura 3.10 ilustra o caso em que os nós são igualmente espaçados pelo intervalo $[u_{min}, u_{max}]$; ou seja, todos os intervalos $[u_k, u_{k+1}]$, têm o mesmo valor $\Delta u = u_{k+1} - u_k$. As curvas em que todos os nós são igualmente espaçados são chamadas splines uniformes.

Pode-se também definir Splines não uniformes, que utilizam vetores de nós não uniformemente distribuídos. Esta flexibilidade permite até mesmo a definição de nós coincidentes ($u_k = u_{k+1}$). Neste caso, termos $0/0$ serão substituídos por 1. Por exemplo, o caso particular de uma spline com vetor de nós $u_k = \{0, 0, 0, 0, 1, 1, 1, 1\}$ define uma curva de Bézier (Angel, 2004).

3.1.5 Curvas NURBS

As Curvas NURBS (Non Uniform Rational B-Splines), são assim chamadas por se tratarem de B-Splines não uniformes definidas por Funções Básicas Racionais, onde a cada ponto de controle é associado um valor real que determina o seu peso na geração da curva. Dessa forma, quanto maior o peso de um ponto de controle, mais influência este ponto exerce sobre o trecho da curva no qual atua.

Uma curva NURBS de grau d é definida por

$$\mathbf{p}(u) = \frac{\sum_{i=0}^{n-1} w_i B_{i,d}(u) \mathbf{p}_i}{\sum_{i=0}^{n-1} w_i B_{i,d}(u)},$$

onde \mathbf{p}_i são os pontos de controle da curva, $B_{i,d}(u)$ são as Funções Básicas B-Splines de grau d e cada valor real w_i é um peso associado ao ponto de controle \mathbf{p}_i . Observe-se que as B-Splines são um caso particular das NURBS, onde $w_i = 1, \forall i$.

Utilizando coordenadas homogêneas, uma curva NURBS definida em \mathbb{R}^3 será representada como uma curva polinomial no espaço \mathbb{R}^4 . Pode-se incorporar o peso de cada ponto de controle em suas coordenadas homogêneas, definindo-se $\mathbf{p}_i^w = w_i \mathbf{p}_i = (w_i x_i, w_i y_i, w_i z_i, w_i)$.

Dessa forma, a equação das NURBS será

$$\mathbf{p}(u) = \frac{\sum_{i=0}^{n-1} B_{i,d}(u) \mathbf{p}_i^w}{\sum_{i=0}^{n-1} w_i B_{i,d}(u)}.$$

As curvas NURBS herdam todas as propriedades das curvas B-spline como envoltória convexa e serem de C^{d-1} . Além disso, incorporam a atribuição explícita de pesos aos pontos de controle. Este ajuste adicional tornam as NURBS uma poderosa ferramenta para modelagem de curvas e superfícies. A Figura 3.11 ilustra diferentes formas de uma curva, obtidas por variações do peso de um único ponto de controle.

Outra importante característica das NURBS trata da visualização em perspectiva. Determinadas operações como rotação, escala ou translação, podem ser obtidas aplicando-se transformações apenas em pontos específicos de uma forma geométrica. Por exemplo, pode-se rotacionar apenas os vértices de um cubo, e as novas coordenadas determinarão o novo cubo rotacionado. Tal processo é muito mais rápido do que rotacionar cada ponto do cubo previamente gerado.

Embora o mesmo processo se aplique a curvas e superfícies, a visualização em perspectiva para as B-Splines não segue o mesmo princípio: a aplicação da transformação apenas nos pontos de controle não produz o mesmo resultado. Segundo Angel (Angel, 2004), as NURBS não apresentam esta limitação.

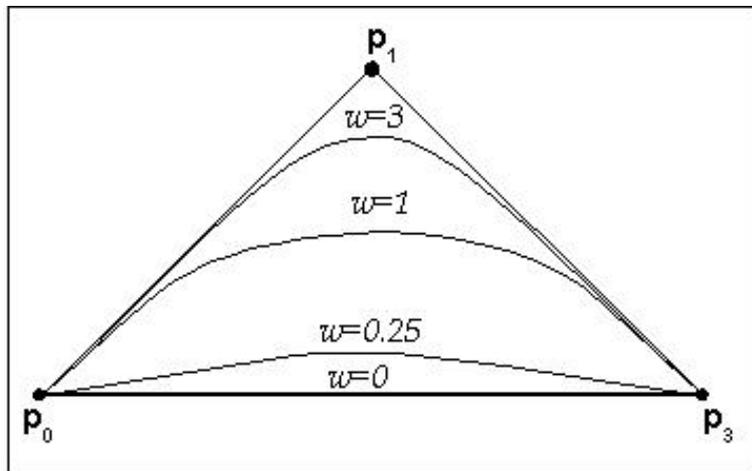


Figura 3.11: Ajustador (Curva NURBS com variação do peso associado ao ponto de controle $p(1)$)

Capítulo 4

Tipos de Superfícies

A modelagem e desenho de superfícies suaves são necessárias em várias aplicações de computação gráfica, seja na modelagem de objetos já existentes ou na criação de novos objetos. As superfícies são modeladas geralmente usando-se malhas de polígonos, superfícies bicúbicas paramétricas e superfícies quádraticas, tem tem como princípio para a sua construção as curvas paramétricas. Este capítulo trata dos conceitos que envolvem a modelagem de superfícies. Serão abordados principalmente os conceitos das superfícies paramétricas, além de três técnicas para modelagem de superfícies que derivam dos conceitos de modelagem de curvas já apresentados.

4.1 Superfícies Parametrizadas

Pode-se estender a definição de curvas para a geração de uma superfície parametrizada $\mathbf{p}(u, v)$:

$$\mathbf{p}(u, v) = \begin{bmatrix} x(u, v) \\ y(u, v) \\ z(u, v) \end{bmatrix} = \sum_{i=0}^n \sum_{j=0}^m \mathbf{c}_{ij} u^i v^j.$$

A especificação dos $3(n+1)(m+1)$ coeficientes determinará uma superfície $\mathbf{p}(u, v)$ específica, gerada a partir da variação dos dois parâmetros u e v . Geralmente define-se $n = m$, com $(u, v) \in [0, 1]^2$, conforme ilustrado na Figura 4.1.

Qualquer superfície pode ser vista como uma coleção de curvas, geradas fixando um dos parâmetros, u ou v , e variando-se o outro. Com isto, pode-se gerar superfícies a partir de curvas específicas (p.ex, polinomiais cúbicas), todas com as

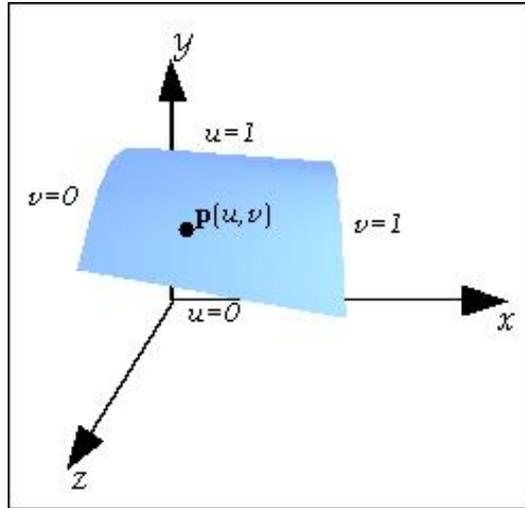


Figura 4.1: Trecho de superfície

mesmas características. Nesse trabalho serão consideradas as superfícies paramétricas definidas a seguir.

4.1.1 Superfícies de Bézier

Extendendo-se o conceito de curva de Bézier, pode-se definir uma superfície de Bézier a partir de um *array* bidimensional de pontos de controle, $\mathbf{P} = [\mathbf{p}_{ij}]_{4 \times 4}$, e dos polinômios de *Bernstein* associados. Então,

$$\mathbf{p}(u, v) = \sum_{i=0}^n \sum_{j=0}^m b_i(u) b_j(v) \mathbf{p}_{ij}.$$

Fixando-se $u = u_0$, tem-se todos os fatores $b_i(u_0)$ fixos, o que determina uma curva parametrizada em v . A curva $\mathbf{p}_{u_0}(v) = \mathbf{p}(u_0, v)$ é uma curva de Bézier denominada *curva isoparamétrica*.

As propriedades das superfícies de Bézier são semelhantes às das curvas de Bézier. Além de derivadas parciais contínuas, a superfície gerada está contida no poliedro determinado pela envoltória convexa dos pontos de controle, e interpola os pontos $\mathbf{p}_{00} = \mathbf{p}(0, 0)$, $\mathbf{p}_{03} = \mathbf{p}(0, 3)$, $\mathbf{p}_{30} = \mathbf{p}(3, 0)$ e $\mathbf{p}_{33} = \mathbf{p}(3, 3)$. É possível interpretar outras condições envolvidas na geração das superfícies de Bézier como uma aproximação das derivadas parciais em seus vértices. Considerando o vértice $(0, 0)$:

$$\begin{aligned}\mathbf{p}(0, 0) &= \mathbf{p}_{00}, \\ \frac{\partial \mathbf{p}}{\partial u}(0, 0) &= 3(\mathbf{p}_{10} - \mathbf{p}_{00}), \\ \frac{\partial \mathbf{p}}{\partial v}(0, 0) &= 3(\mathbf{p}_{01} - \mathbf{p}_{00}), \\ \frac{\partial^2 \mathbf{p}}{\partial u \partial v}(0, 0) &= 9(\mathbf{p}_{00} - \mathbf{p}_{01} + \mathbf{p}_{10} - \mathbf{p}_{11}).\end{aligned}$$

As três primeiras condições são extensões dos resultados para a curva de Bézier; a quarta condição pode ser vista como uma medida da tendência de curvatura da superfície na vizinhança do ponto $(0, 0)$.

4.1.2 Superfícies B-Splines

Pode-se estender a aplicação das funções de base das curvas B-Splines para definir superfícies parametrizadas:

$$\mathbf{p}(u, v) = \sum_{i=0}^n \sum_{j=0}^m B_{i,j,d}(u, v) \mathbf{p}_{ij},$$

onde a *blending function* $B_{i,j,d}(u, v) = B_{i,d}(u)B_{j,d}(v)$ é definida a partir das funções de base das curvas B-Spline (Watt, 1993).

Além da matriz com os pontos de controle \mathbf{p}_{ij} , as superfícies B-Spline utilizam dois vetores de nós (*knots*) u_k e v_k , contendo sequências não decrescentes de valores para os parâmetros u e v (utilizadas nas funções de base para as curvas B-Splines).

Analogamente às curvas B-Spline, as sequências u_k e v_k determinam uma subdivisão de toda superfície gerada (*surface patches*), com sua continuidade C^2 sendo decorrente do caso unidimensional. Tal como as curvas B-Spline, os valores dos nós podem ser irregularmente espaçados, originando superfícies não uniformes. Deve-se evitar pontos de controle duplicados, que criam descontinuidade.

Um exemplo simples de superfície B-Spline, determinada por 16 pontos de controle, é ilustrado na Figura 4.2.

Nesse caso, a superfície gerada ocupa a região próxima aos quatro pontos de controle centrais. Devido a herança das propriedades da envoltória convexa e continuidade C^2 , a superfície B-Spline é considerada mais suave que superfícies construídas a partir das curvas de Bézier e Hermite.

4.1.3 Superfícies NURBS

As superfícies NURBS apresentam características semelhantes às superfícies B-Splines.

Assim como as curvas NURBS, é possível definir superfícies a partir de uma malha de pontos de controle $\mathbf{p}_{i,j}$ com pesos $w_{i,j}$ associados, e utilizando coordenadas homogêneas. Uma superfície NURBS definida por curvas polinomiais

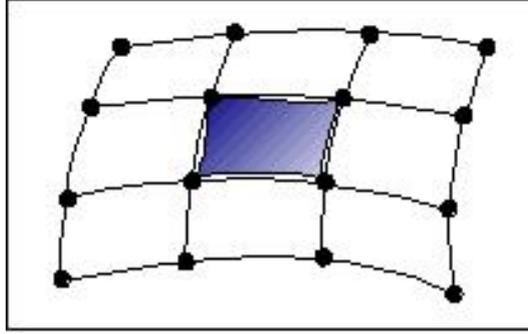


Figura 4.2: Trecho de superfície B-Spline

em u e v (de graus p e q , respectivamente) e funções de base ($B_{i,p}(u)$ e $B_{j,q}(v)$) (WEISSTEIN, 2000), têm a forma geral:

$$\mathbf{p}(u, v) = \frac{\sum_{i=0}^n \sum_{j=0}^m B_{i,p}(u) B_{j,q}(v) w_{i,j} \mathbf{p}_{i,j}}{\sum_{i=0}^n \sum_{j=0}^m B_{i,p}(u) B_{j,q}(v) w_{i,j}}$$

Essa equação pode ser reescrita como:

$$\mathbf{p}(u, v) = \sum_{i=0}^n \sum_{j=0}^m \mathbf{R}_{i,j}(u, v) \mathbf{p}_{i,j} \quad 0 \leq u, v \leq 1$$

onde

$$\mathbf{R}_{i,j}(u, v) = \frac{B_{i,p}(u) B_{j,q}(v) w_{i,j}}{\sum_{i=0}^n \sum_{j=0}^m B_{i,p}(u) B_{j,q}(v) w_{i,j}}.$$

Utilizando coordenadas homogêneas, e definido-se $\mathbf{p}_{i,j}^w = w_{i,j} \mathbf{p}_{i,j}$, obtêm-se

$$\mathbf{p}(u, v) = \sum_{i=0}^n \sum_{j=0}^m B_{i,p}(u) B_{j,q}(v) \mathbf{p}_{i,j}^w$$

sendo essa a fórmula mais utilizada nos algoritmos computacionais.

As Funções Básicas Racionais apresentam as mesmas propriedades das Funções Básicas B-Spline. Conseqüentemente, as propriedades das superfícies NURBS são as mesmas das superfícies B-Spline. É importante observar que as superfícies NURBS generalizam as superfícies de Bézier e superfícies B-Spline.

Capítulo 5

Conclusão

Este trabalho apresentou os tipos de curvas e superfícies suas principais características quanto a deformação e suavidade. Vimos que para a computação gráfica, o entendimento das curvas é essencial para a definição de superfícies deformáveis, principalmente quanto a sua suavidade e continuidade.

As superfícies NURBS apresentam uma melhor suavidade em relação às demais, além de poder generalizar as superfícies de Bézier e B-Splines. sendo ela a mais utilizada nos algoritmos computacionais.

Outra importante característica das NURBS trata da visualização em perspectiva. Determinadas operações como rotação, escala ou translação, podem ser obtidas aplicando-se transformações apenas em pontos específicos de uma forma geométrica. Por exemplo, pode-se rotacionar apenas os vértices de um cubo, e as novas coordenadas determinarão o novo cubo rotacionado. Tal processo é muito mais rápido do que rotacionar cada ponto do cubo previamente gerado.

Embora o mesmo processo se aplique a curvas e superfícies, a visualização em perspectiva de para as B-Splines não segue o mesmo princípio: a aplicação da transformação apenas nos pontos de controle não produz o mesmo resultado. Segundo Angel (Angel, 2004), as NURBS não apresentam esta limitação.

Com base no estudo realizado, é possível definir superfícies suaves e que mantenham suas propriedades de continuidade quando submetidas a uma deformação. de posse desse conhecimento, é possível desenvolver uma ferramenta capaz de simular tais resultados, de forma interativa e que possa demonstrar a qualidade e a importância da deformação de superfícies flexíveis em diversos tipos de simulação tridimensional.

Referências Bibliográficas

- [ANGEL 2004] ANGEL, Edward. **Interactive Computer Graphics**. 4nd ed. Boston Massachusetts: Addison Wesley, 2006. 784 p.
- [BOUGKNIGHT 1970] BOUGKNIGHT, W. J. **A Procedure for Generation of Three-dimensional Half-tone Computer Graphics Presentations**. Communications of the ACM, 1970.
- [BERG 1998] BERG, Mark T. de; KREVELD, Marc J. Van; OVERMARS, Mark; CHEONG, Otfried S. **A Procedure for Generation of Three-dimensional Half-tone Computer Graphics Presentations**. Communications of the ACM, 1970.
- [CELNIKER 1991] CELNIKER, George; GOSSARD, Dave. **Deformable Curve and Surface Finite-Elements for Free Form Shape Design**. Computer Graphics, Vol. 25, N. 4, Julho 1991.
- [EBERT 1998] EBERT, D. et al. **Texturing & modeling**. 2nd ed. San Diego: AP Professional, 1998, 415p.
- [ETZMUF 1998] ETZMUF, Olaf, et al. **A Cloth Modelling System for**

- Animated Characters.** In Proceedings of Graphiktag 2001.
- [FOLEY 1990] FOLEY, J. D.; DAM, M. V. **Fundamentals of Interactive Computer Graphics.** 1nd ed. Boston Massachusetts: Addison Wesley, 1984. 348 p.
- [GLASSNER, 1995] GLASSNER, GLASSNER, A. S. **Principles of digital image synthesis.** San Francisco: Morgan Kaufmann, USA, 1995.
- [MCLNERNY 1996] NG, H. N. and GRIMSDALE, R. L. **Computer Graphics Techniques for Modeling Cloth.** IEEE Comput. Graph. Appl. 16, 5. Sep. 1996.
- [MONTAGNAT 2001] MONTAGNAT, J., DELINGETTE, H, AYACHE N. **A Review of Deformable Surfaces: Topology, Geometry and Deformation.** Image and Vision Computing, vol. 19, no. 14, pp. 1023–1040, 2001.
- [MCLNERNY 1996] MCLNERNY, T. and TERZOPOULOS, D. **Deformable models in medical image analysis: a survey.** Medical Image Analysis, 1(2):91.108, 1996.
- [PERSIANO 1996] PERSIANO, R. M. **Bases da Modelagem Geométrica.** Trabalho Apresentado na 10^a Escola de Computação, Instituto de Computação da UNICAMP, 141 p., 1996.

- [TERZOPOULOS 1988] TERZOPOULOS, D. and FLEISCHER, K. **Deformable Models**. Visual Computer, 4:306.331, 1988.
- [VOLINO 2000] VOLINO and THALMANN, N. M. **Virtual Clothing, Theory and Practice**. Berlin:Springer, 2000.
- [WATT 1993] WATT, Allan **3D Computer Graphics**. 2nd ed. Reading, Massachusetts: Addison Wesley, 1993. 500 p.
- [WEISSTEIN 2007] WEISSTEIN, Eric W. "NURBS Surface."From MathWorld—A Wolfram Web Resource. <http://mathworld.wolfram.com/NURBSSurface.html>. Acessado em outubro 2007.