

Representação de Curvas e Superfícies

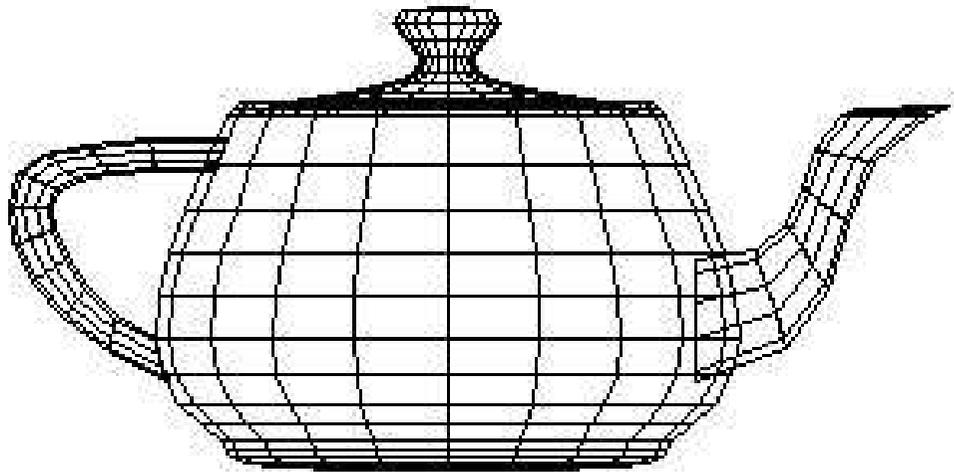
Sistemas Gráficos/
Computação Gráfica e Interfaces

Representação de Curvas e Superfícies

Representação de superfícies: permitem descrever objectos através das suas faces. As três representações mais comuns são:

- Malha poligonal
- Superfícies paramétricas bicúbicas
- Superfícies quadráticas

Representação paramétricas de curvas: importantes na computação gráfica 2D e pelo facto das superfícies paramétricas serem uma generalização destas curvas.



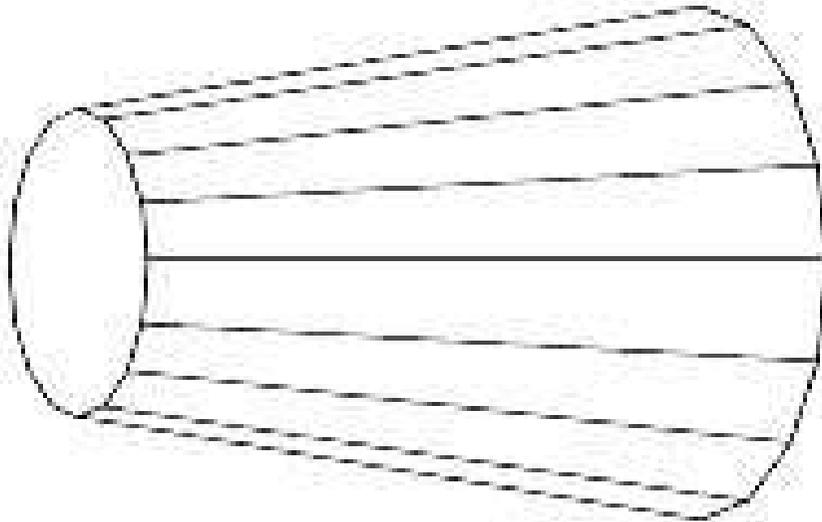
“Tea-pot” modelado por superfícies curvas suaves (bicúbicas).

Modelo de referência na Computação Gráfica, nomeadamente para teste de novas técnicas de realismo de textura e superfície.

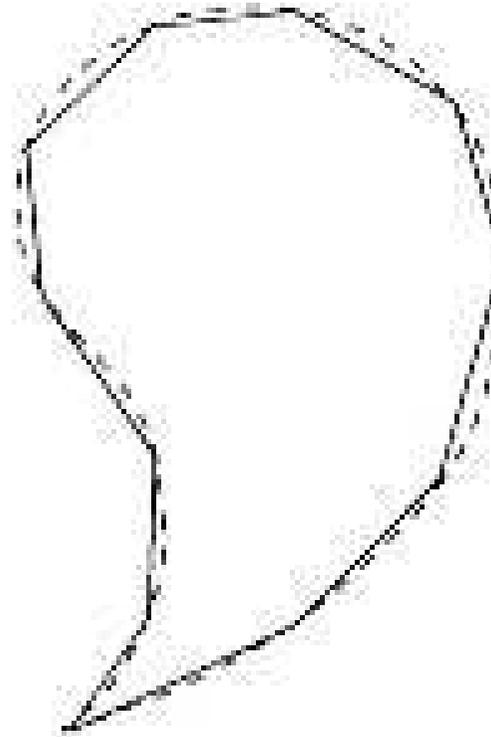
Criado por Martin Newell (1975)

Malha Poligonal

Malha Poligonal: é uma colecção de arestas, vértices e polígonos interligados de modo que cada aresta é apenas partilhado no máximo por dois polígonos.



Objecto 3D representado por malha de polígonos.



Curva \leftrightarrow linha poligonal

Secção de um objecto curvo.

O erro de aproximação pode ser reduzido aumentando o número de polígonos.

Malha Poligonal

Características da malha poligonal:

- Uma aresta liga 2 vértices.
- Um polígono é definido por uma sequência fechada de arestas.
- Uma aresta pode ser partilhada por 2 polígonos adjacentes.
- Um vértice é partilhado pelo menos por 2 arestas.
- Todas as arestas fazem parte de algum polígono.

A estrutura de dados para **representar a malha poligonal** pode ter várias configurações, que são avaliadas pelo **espaço de memória** e **tempo de processamento** necessário para obter resposta, por exemplo, a:

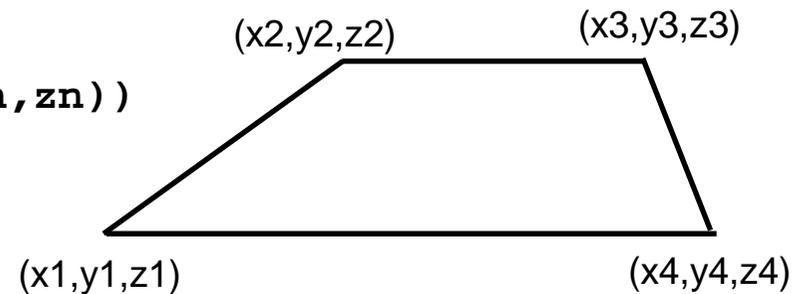
- Obter todas as arestas que se unem num dado vértice.
- Determinar os polígonos que partilham uma aresta ou um vértice.
- Determinar os vértices ligados a uma aresta.
- Determinar as arestas de um polígono.
- Representar graficamente a malha.
- Identificar erros na representação, como falta de uma aresta, vértice ou polígono.

Malha Poligonal

1. Representação Explícita: cada polígono é representado por uma lista de coordenadas dos vértices que o constituem.

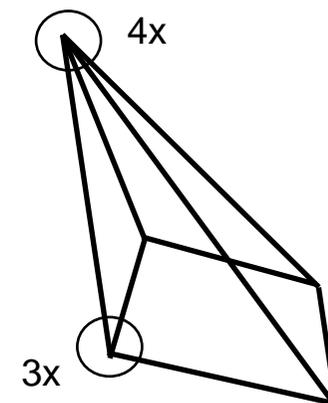
Uma aresta é definida por dois vértices consecutivos e entre o último e primeiro da lista.

$P = ((x_1, y_1, z_1), (x_2, y_2, z_2), \dots, (x_n, y_n, z_n))$



Avaliação da estrutura de dados:

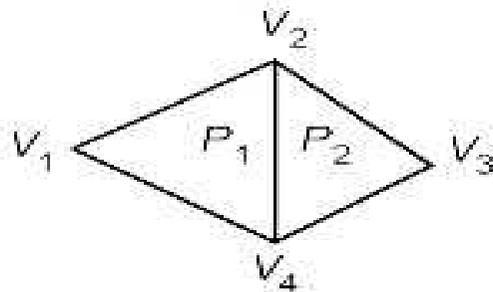
- Consumo de memória (vértices repetidos).
- Não há uma representação explícita das arestas e vértices partilhados.
- Na representação gráfica a mesma aresta é “clipped” e desenhada mais do que uma vez.
- Ao arrastar um vértice é necessário conhecer todas as arestas que partilham aquele vértice.



Malha Poligonal

2. Representação por Apontadores para Lista de Vértices: cada polígono é representado por uma lista de índices (ou apontadores) para uma lista de vértices.

Lista de Vértices $V = ((x_1, y_1, z_1), (x_2, y_2, z_2), \dots, (x_n, y_n, z_n))$



$V = (V_1, V_2, V_3, V_4) = (x_1, y_1, z_1), (x_2, y_2, z_2), \dots, (x_4, y_4, z_4)$

$P_1 = (1, 2, 4)$

$P_2 = (4, 2, 3)$

Vantagens:

- Cada vértice da malha poligonal é guardado uma única vez na memória.
- A coordenada de um vértice é facilmente alterada.

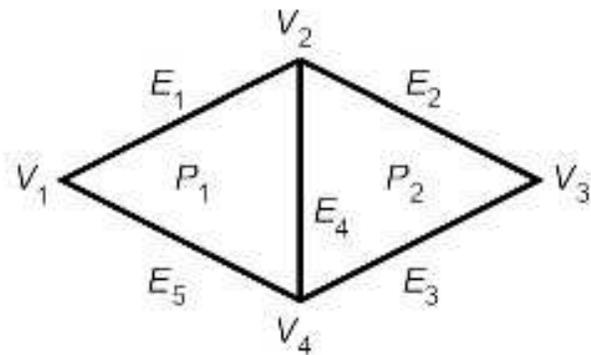
Desvantagens:

- Difícil obter os polígonos que partilham uma dada aresta.
- As arestas continuam a ser “*clipped*” e desenhada mais do que uma vez.

Malha Poligonal

3. Representação por Apontadores para Lista de Arestas: cada polígono é representado por uma lista de apontadores para uma lista de arestas, na qual cada aresta aparece uma única vez. Por sua vez, cada aresta aponta para os dois vértices que a definem e guarda também quais os polígonos a que pertence.

Um polígono é representado por $P=(E_1,E_2,\dots,E_n)$ e uma aresta como $E=(V_1,V_2,P_1,P_2)$. Se a aresta pertence apenas a um polígono então **P2** é *null*.



$$V = (V_1, V_2, V_3, V_4) = ((x_1, y_1, z_1), \dots, (x_4, y_4, z_4))$$

$$E_1 = (V_1, V_2, P_1,)$$

$$E_2 = (V_2, V_3, P_2,)$$

$$E_3 = (V_3, V_4, P_2,)$$

$$E_4 = (V_4, V_2, P_1, P_2)$$

$$E_5 = (V_4, V_1, P_1,)$$

$$P_1 = (E_1, E_4, E_5)$$

$$P_2 = (E_2, E_3, E_4)$$

Curvas Cúbicas

Motivação: Representar curvas suaves do mundo real.

- A representação por **malha poligonal** é uma aproximação de primeira ordem:
 - A curva é aproximada por uma sequência de segmentos lineares.
 - Grande quantidade de dados (vértices) para obter a curva com precisão.
 - Difícil manipulação para mudar a forma da curva, i.e. necessário posicionar vários pontos com precisão.
- **Geralmente utilizam-se polinómios de grau 3 (Curvas Cúbicas), sendo a curva completa formada por um conjunto de curvas cúbicas.**
 - **grau < 3** oferecem pequena flexibilidade no controlo da forma das curvas e não permitem uma interpolação entre dois pontos com a definição da derivada nos pontos extremos. Um polinómio de grau 2 é especificado por 3 pontos que definem o plano onde a curva toma lugar.
 - **grau > 3** podem introduzir oscilações indesejáveis e exigir maior cálculo computacional.

Curvas Cúbicas

A representação das curvas é feita na forma PARAMÉTRICA:

$$x = f_x(t), y = f_y(t)$$

$$\text{ex: } x=3t^3 + t^2 \quad y=2t^3+t$$

A forma Explícita:

$$y=f(x) \quad \text{ex: } y=x^3+2x^2$$

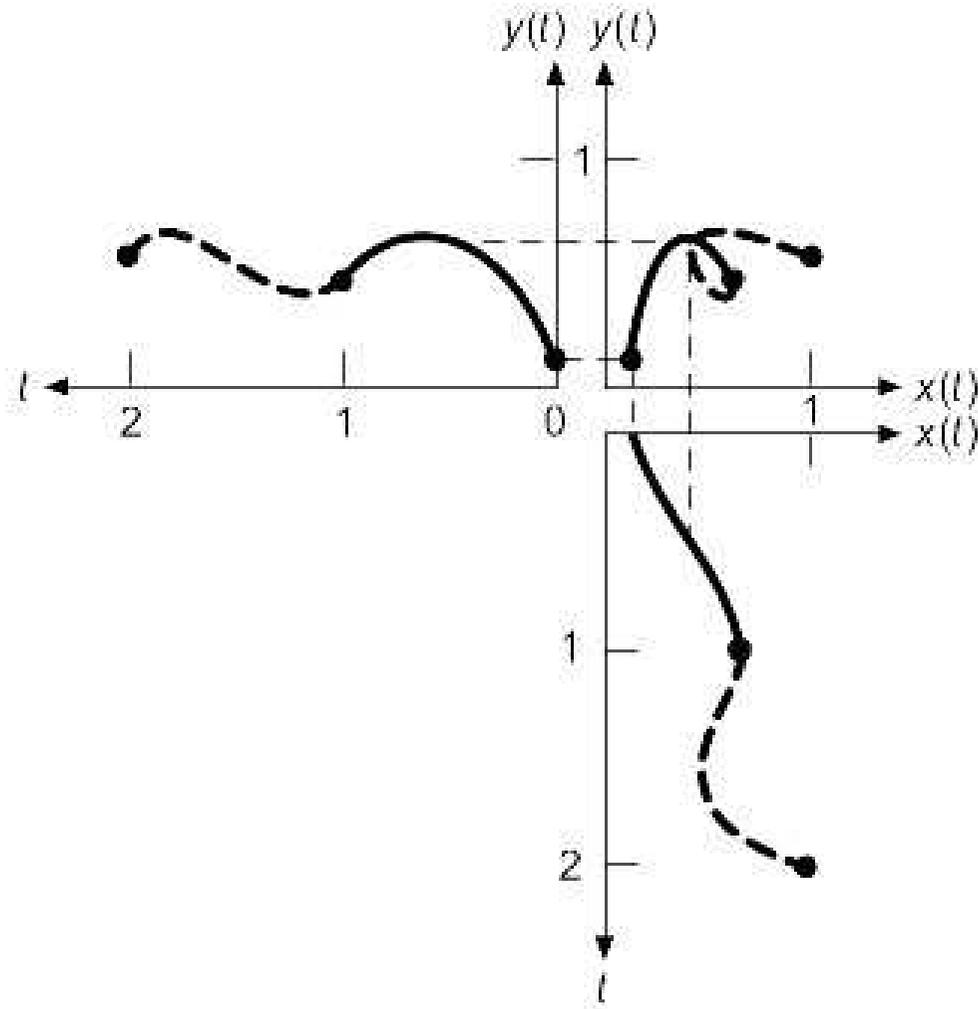
1. Não podemos ter vários valores de **y** para o mesmo **x**
2. Não podemos descrever curvas com tangentes verticais

A forma Implícita:

$$f(x,y)=0 \quad \text{ex: } x^2+y^2-r^2 = 0$$

1. Necessita de restrições para poder modelar apenas uma parte da curva
2. Difícil juntar duas curvas de forma suave

Curvas Cúbicas Paramétricas



A figura mostra uma curva formada por duas curvas cúbicas paramétricas em 2D.

Curvas Cúbicas Paramétricas

Forma geral de representação da curva:

$$x(t) = a_x t^3 + b_x t^2 + c_x t + d_x$$

$$y(t) = a_y t^3 + b_y t^2 + c_y t + d_y$$

$$z(t) = a_z t^3 + b_z t^2 + c_z t + d_z \quad 0 \leq t \leq 1$$

Sendo: $T = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix}$

$$C = \begin{bmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \\ d_x & d_y & d_z \end{bmatrix}$$

$$Q(t) = \begin{bmatrix} x(t) & y(t) & z(t) \end{bmatrix} = T.C$$

Curvas Cúbicas Paramétricas

A representação anterior é usada para representar uma única curva. **Como juntar os vários segmentos de curva ?**

Pretendemos a junção num ponto \rightarrow continuidade geométrica e,

Que tenham o mesmo declive na junção \rightarrow suavidade (continuidade da derivada).

A garantia de continuidade e suavidade na junção é garantida fazendo coincidir as derivadas (tangentes) das curvas no ponto de junção. Para isso calcula-se:

$$\frac{\partial Q(t)}{\partial t} = \left(\frac{\partial x(t)}{\partial t} \quad \frac{\partial y(t)}{\partial t} \quad \frac{\partial z(t)}{\partial t} \right) = \frac{\partial(CT)}{\partial t} = C \frac{\partial T}{\partial t}$$

Com:
$$\frac{\partial T}{\partial t} = [3t^2 \quad 2t \quad 1 \quad 0]$$

Curvas Cúbicas Paramétricas

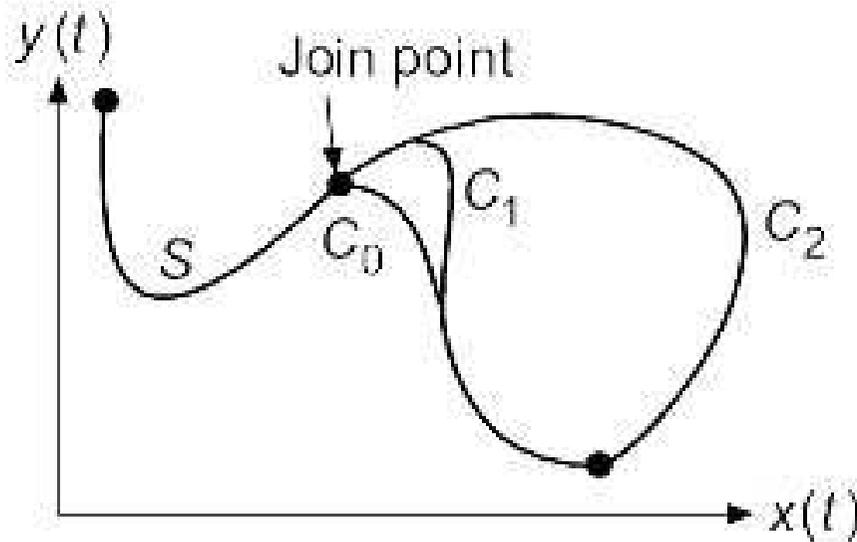
Tipos de Continuidade:

- G^0 – continuidade geométrica zero → as curvas juntam-se num ponto.
- G^1 – continuidade geométrica um → a direcção dos vectores tangentes é igual.
- C^1 – continuidade paramétrica 1 → as tangentes no ponto de junção têm a mesma direcção e amplitude (primeira derivada igual).
- C^n – continuidade paramétrica n → as curvas têm no ponto de junção todas as derivadas iguais até à ordem n.

Curvas Cúbicas Paramétricas

Se considerarmos t como **tempo**, a continuidade C^1 significa que a velocidade de um objecto que se desloque ao longo da curva se mantém contínua

A continuidade C^2 implicaria que a aceleração seria também contínua.



No ponto de junção da curva **S** com as curvas **C₀**, **C₁** e **C₂** temos:

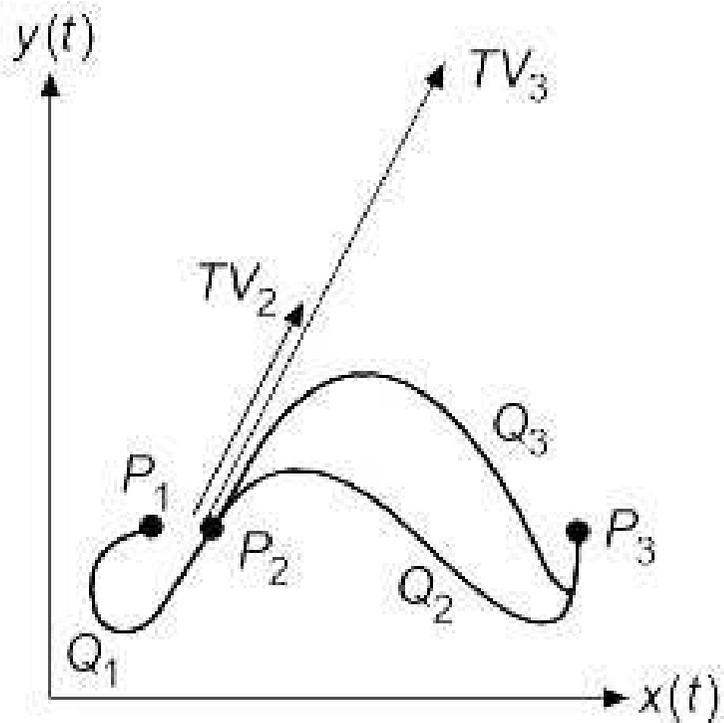
Continuidade G^0 entre **S** e **C₀**

Continuidade C^1 entre **S** e **C₁**

Continuidade C^2 entre **S** e **C₂**

Curvas Cúbicas Paramétricas

A continuidade paramétrica é mais restritiva que a continuidade geométrica:



Por exemplo: C^1 implica G^1

No ponto de junção P_2 temos:

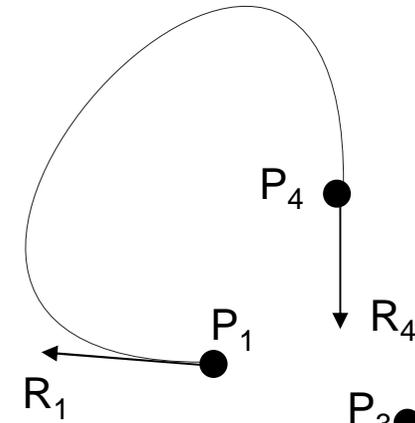
Q_2 e Q_3 são G^1 com Q_1

Só Q_2 é C^1 com Q_1 ($TV_1=TV_2$)

Curvas Cúbicas Paramétricas – Tipos de Curvas

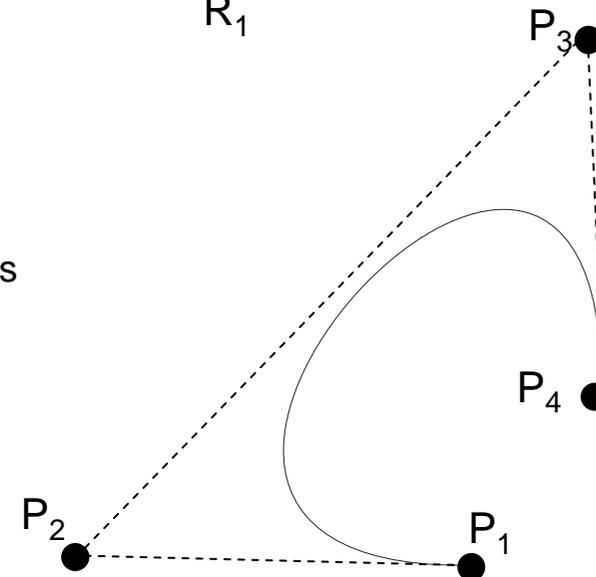
1. Curvas de Hermite

- Continuidade G^1 nos pontos de junção
- Vector geométrico:
 - 2 pontos extremos e
 - Os vectores tangentes nesses pontos



2. Curvas de Bézier

- Continuidade G^1 nos pontos de junção
- Vector geométrico:
 - 2 pontos extremos e
 - 2 pontos que controlam os vectores tangentes nesses extremos



3. Curvas Splines

- Família de curvas muito alargada
- Maior controlo da continuidade nos pontos de junção (Continuidade C^1 e C^2)

Notação comum

$$Q(t) = [x(t) \quad y(t) \quad z(t)] = T.C$$

$$Q(t) = T.M.G$$

$$\begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix}$$

$$\begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix}$$

$$\begin{bmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{bmatrix}$$

Matriz T

Matriz de Base

Matriz Geométrica

Matriz de Base: Caracteriza o tipo de curva

Matriz Geométrica: Condiciona geometricamente uma dada curva e contém valores relacionados com a geometria da curva.

Notação comum

$$Q(t) = T.M.G$$

$$Q(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \cdot \begin{bmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{bmatrix}$$

$$Q(t) = (t^3 m_{11} + t^2 m_{21} + t m_{31} + m_{41}).G_1 + (t^3 m_{12} + t^2 m_{22} + t m_{32} + m_{42}).G_2 + \\ (t^3 m_{13} + t^2 m_{23} + t m_{33} + m_{43}).G_3 + (t^3 m_{14} + t^2 m_{24} + t m_{34} + m_{44}).G_4$$

Conclusão 1: Q(t) é uma soma pesada dos elementos do vector geométrico

Conclusão 2: Os pesos são polinomiais cúbicas em t → **FUNÇÕES DE MISTURA**

(Blending functions)

$$Q(t) = T.C = T.M.G = B.G$$

Curvas de Hermite

$$Q(t) = T \cdot M_H \cdot G_H = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} M_H \cdot G_H = B_H \cdot G_H$$

$$Q'(t) = \begin{bmatrix} 3t^2 & 2t & 1 & 0 \end{bmatrix} M_H \cdot G_H$$

Vector geométrico:

$$G_H = \begin{bmatrix} P_1 \\ P_4 \\ R_1 \\ R_4 \end{bmatrix}$$

$$\left\{ \begin{array}{l} Q(0) = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} M_H \cdot G_H = P_1 \\ Q(1) = \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} M_H \cdot G_H = P_4 \\ Q'(0) = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} M_H \cdot G_H = R_1 \\ Q'(1) = \begin{bmatrix} 3 & 2 & 1 & 0 \end{bmatrix} M_H \cdot G_H = R_4 \end{array} \right.$$

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \cdot M_H \cdot G_H = \begin{bmatrix} P_1 \\ P_2 \\ R_1 \\ R_4 \end{bmatrix} = G_H \quad \Rightarrow \quad M_H = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix}^{-1} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Curvas de Hermite

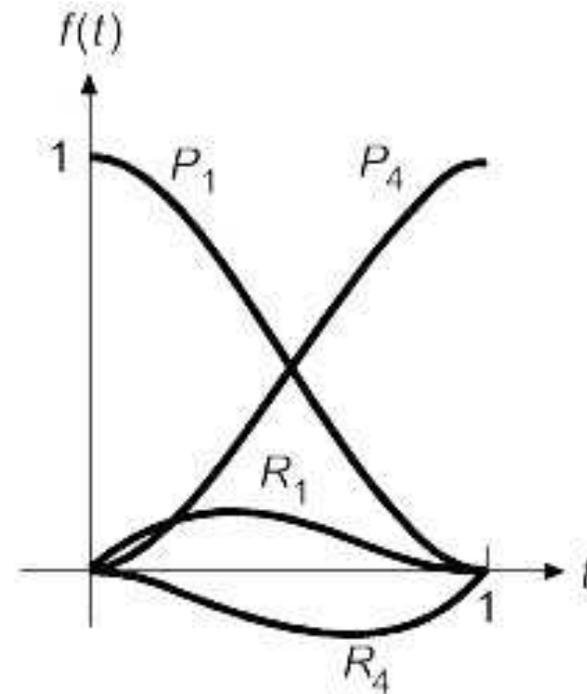
Funções de Mistura (Blending functions)

$$Q(t) = T \cdot M_H \cdot G_H = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} M_H \cdot G_H = B_H \cdot G_H$$

$$M_H = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

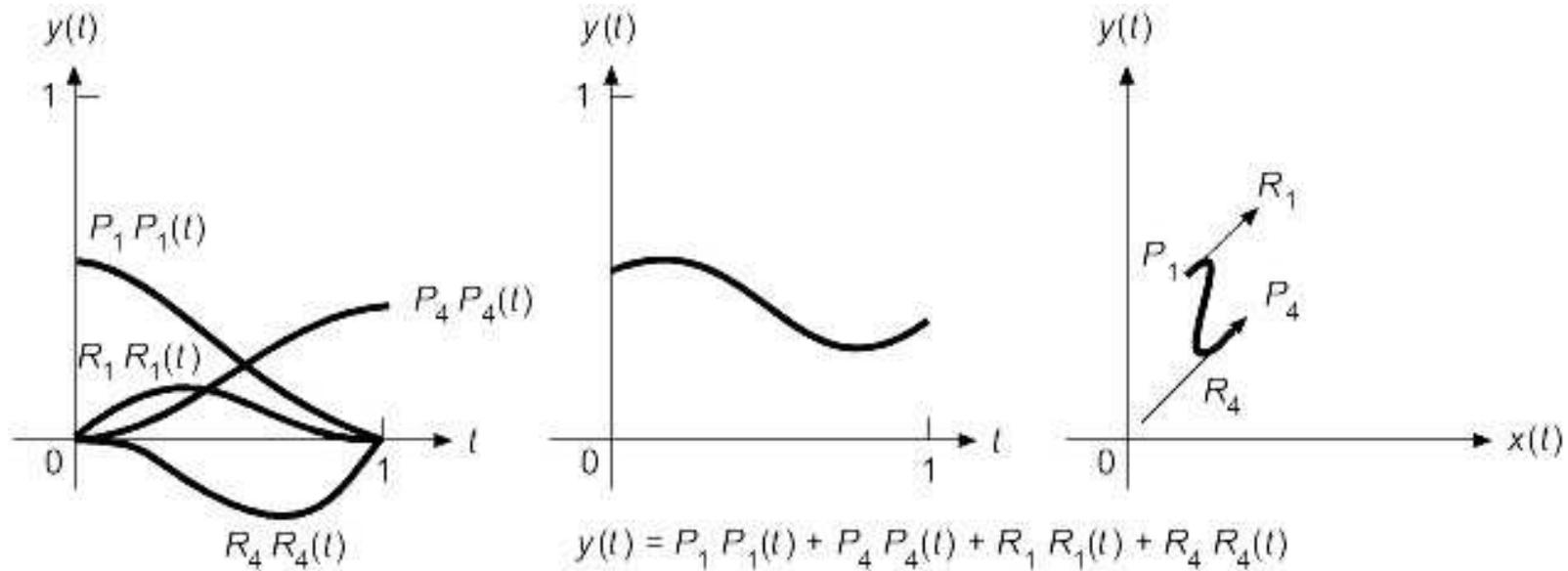
$$G_H = \begin{bmatrix} P_1 \\ P_4 \\ R_1 \\ R_4 \end{bmatrix}$$

$$Q(t) = (2t^3 - 3t^2 + 1)P_1 + (-2t^3 + 3t^2)P_4 + (t^3 - 2t^2 + t)R_1 + (t^3 - t^2)R_4$$



Funções de Mistura das curvas de Hermite, referenciadas pelo elemento do vector geométrico que as multiplica, respectivamente.

Curvas de Hermite - Exemplo



Esquerda: Funções de mistura pesadas pelo factor correspondente do vector geométrico.

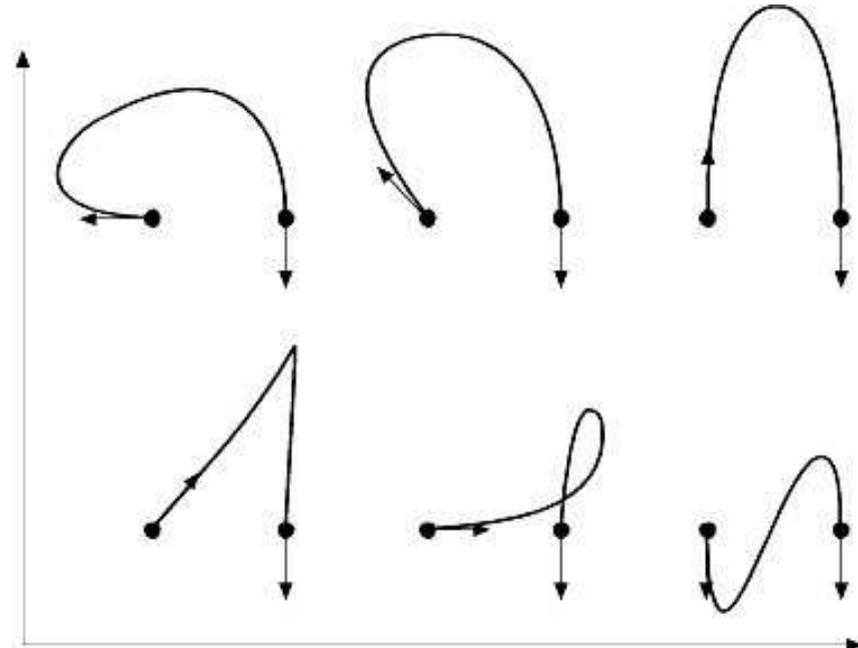
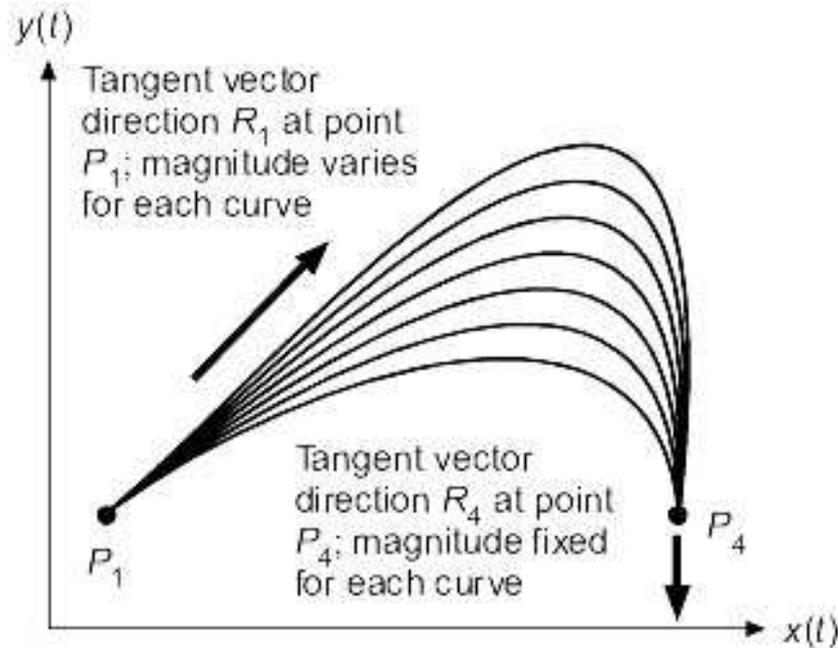
Centro: $y(t) =$ soma das quatro funções da esquerda

Direita: Curva de Hermite

Curvas de Hermite - Exemplos

- P_1 e P_4 fixos
- R_4 fixo
- R_1 varia em amplitude

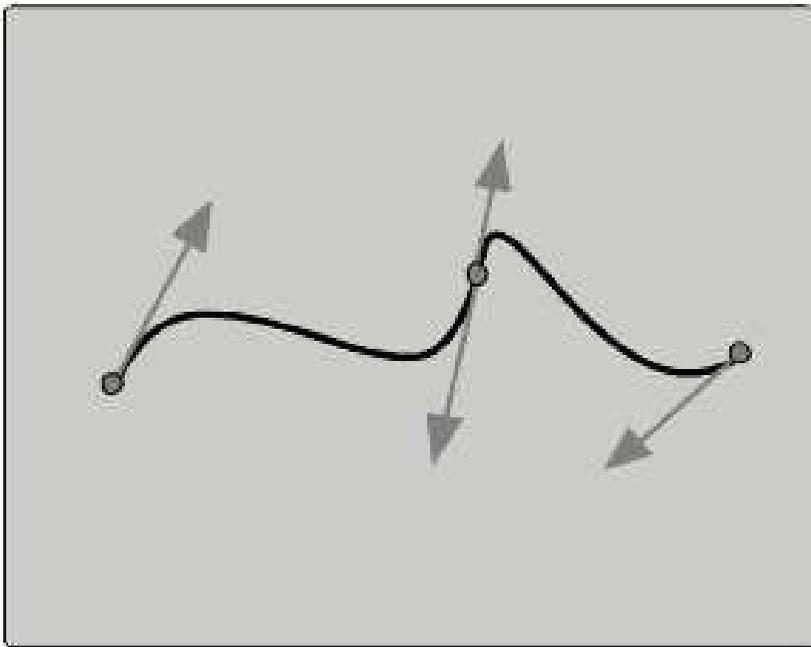
- P_1 e P_4 fixos
- R_4 fixo
- R_1 varia em direcção



Curvas de Hermite

Exemplo de Desenho Interactivo

- Os pontos extremos podem ser reposicionados
- Os vectores tangentes podem ser alterados puxando as setas
- Os vectores tangentes são forçados a serem colineares (continuidade G^1) e R_4 é visualizado em sentido contrário (maior visibilidade)
- É comum dispor de comandos para forçar continuidade C^0 , C^1 ou G^1



Continuidade na junção:

$$\begin{bmatrix} P_1 \\ P_4 \\ R_1 \\ R_4 \end{bmatrix} \rightarrow \begin{bmatrix} P_4 \\ P_7 \\ K.R_4 \\ R_7 \end{bmatrix}$$

- $K > 0 \rightarrow G^1$
- $K = 1 \rightarrow C^1$

Curvas de Hermite

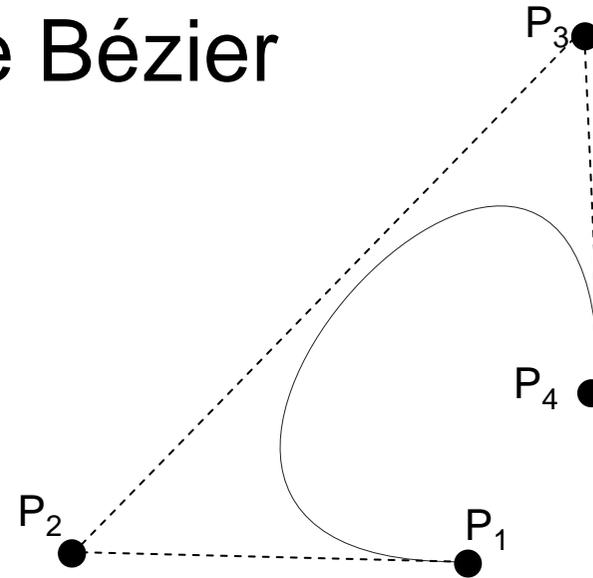
3. Seja a sucessão $C1, C2, C3, C4$ de curvas de Hermite representadas pelos vectores geométricos juntos. Complete estes com os valores em falta, de forma a obter continuidade do tipo C^1 em todos os pontos de junção e justifique os casos em que isso não seja possível, de acordo com os dados fornecidos.

$$C1 = \begin{bmatrix} 0,0 \\ 3,3 \\ 0,2 \\ ?,? \end{bmatrix}; \quad C2 = \begin{bmatrix} ?,? \\ ?,? \\ 2,0 \\ 0,2 \end{bmatrix}; \quad C3 = \begin{bmatrix} 6,6 \\ 3,6 \\ 0,1 \\ 0,-1 \end{bmatrix}; \quad C4 = \begin{bmatrix} 3,3 \\ 6,3 \\ ?,? \\ 2,0 \end{bmatrix}$$

Curvas de Bézier

Vector Geométrico:

$$G_B = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix}$$



Para uma mesma curva,
demonstra-se que,
comparando com G_H :

$$R_1 = Q'(0) = 3.(P_2 - P_1)$$

$$R_4 = Q'(1) = 3.(P_4 - P_3)$$



$$G_H = \begin{bmatrix} P_1 \\ P_4 \\ R_1 \\ R_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix}$$

$G_H = M_{HB} \cdot G_B$

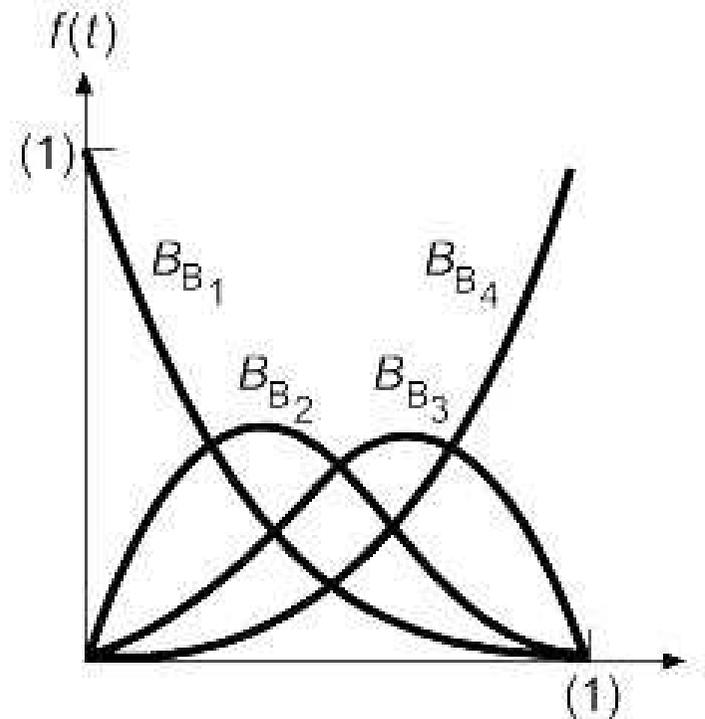
Curvas de Bézier

$$Q(t) = T \cdot M_H \cdot G_H = T \cdot M_H \cdot (M_{HB} \cdot G_B) = T \cdot \underbrace{(M_H \cdot M_{HB})}_{M_B} \cdot G_B$$

- A mesma curva em representação Bézier: $Q(t) = T \cdot M_B \cdot G_B$

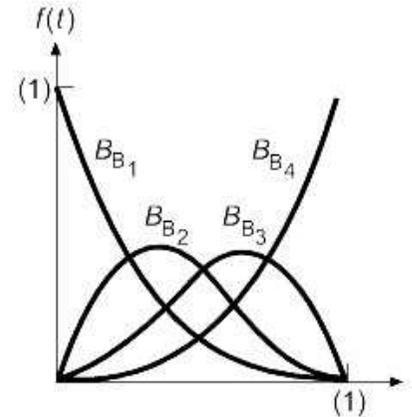
$$M_B = M_H \cdot M_{HB} = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$Q(t) = \begin{aligned} &(1-t)^3 P_1 + \\ &3t(1-t)^2 P_2 + \\ &3t^2(1-t) P_3 + \\ &t^3 P_4 \end{aligned}$$



Curvas de Bézier

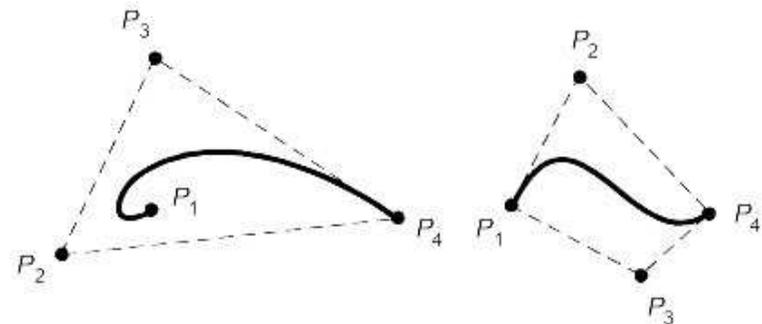
$$Q(t) = (1-t)^3 P_1 + 3t(1-t)^2 P_2 + 3t^2(1-t) P_3 + t^3 P_4$$



Observações sobre as funções de Mistura:

- Para $t=0$ $Q(t)=P_1$, para $t=1$ $Q(t)=P_4$ \rightarrow A curva passa em P_1 e P_4
- A soma em qualquer ponto é 1.
- Verifica-se que $Q(t)$ é uma **média pesada dos 4 pontos de controlo**, logo a **curva está contida no interior do polígono** convexo (2D) ou poliedro convexo (3D) definido por esses pontos, designado de “convex hull”.

Que vantagem podemos extrair daqui ?



Curvas de Bézier

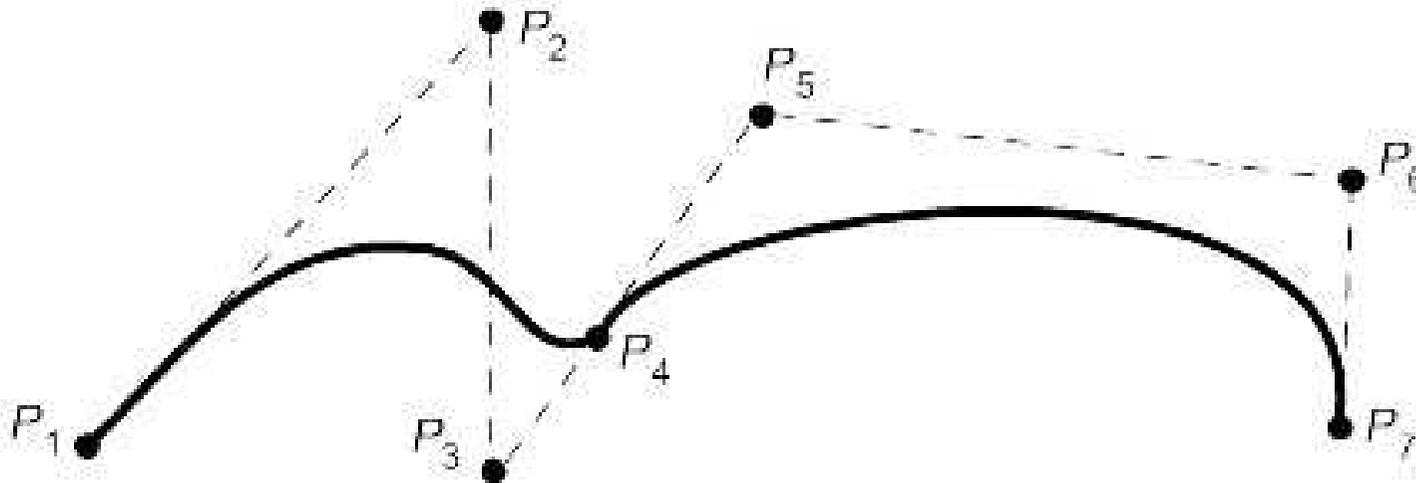
Junção de curvas de Bézier

Continuidade G^1 :

$$P_4 - P_3 = K(P_5 - P_4) \quad \text{com } K > 0 \quad \text{i.e. } P_3, P_4 \text{ e } P_5 \text{ devem ser colineares}$$

Continuidade C^1 :

$$P_4 - P_3 = K(P_5 - P_4) \quad \text{restringindo } K = 1$$



Desenho de Curvas Cúbicas

Dois algoritmos:

1. Avaliação de $x(t)$, $y(t)$ e $z(t)$ para valores incrementais de t entre 0 e 1.
2. Subdivisão da curva: Algoritmo de Casteljau

1. Avaliação de $x(t)$, $y(t)$ e $z(t)$

Regra de Horner permite reduzir o número de operações de 11 multiplicações e 10 adições para 9 e 10, respectivamente.

$$f(t) = at^3 + bt^2 + ct + d = ((at + b).t + c).t + d$$

2. Algoritmo de Casteljau

Efectua a subdivisão recursiva da curva, parando apenas quando a curva em questão é suficientemente “plana” para poder ser aproximada por um segmento de recta.

Algoritmo eficiente: requer apenas 6 *shifts* e 6 adições em cada divisão.

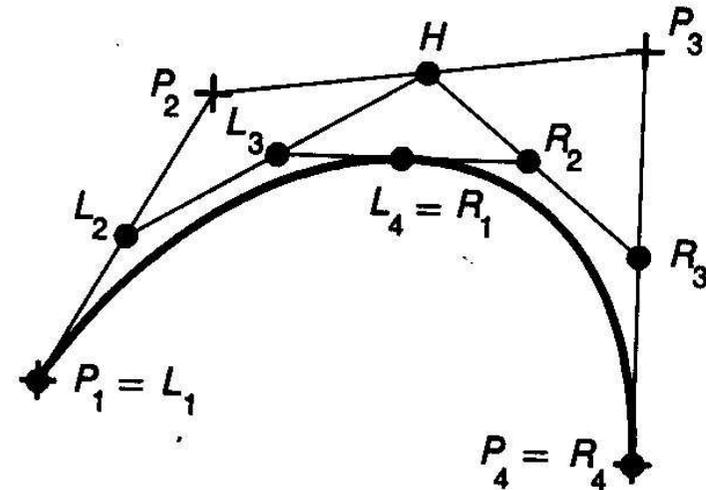
Desenho de Curvas Cúbicas - Algoritmo de Casteljau

Critérios possíveis de paragem:

- A curva em questão é suficientemente “plana” para poder ser aproximada por um segmento de recta.
- Os 4 pontos de controlo estão no mesmo pixel.

$$L_2 = (P_1 + P_2)/2, \quad H = (P_2 + P_3)/2, \quad L_3 = (L_2 + H)/2, \quad R_3 = (P_3 + P_4)/2$$

$$R_2 = (H + R_3)/2, \quad L_4 = R_1 = (L_3 + R_2)/2$$



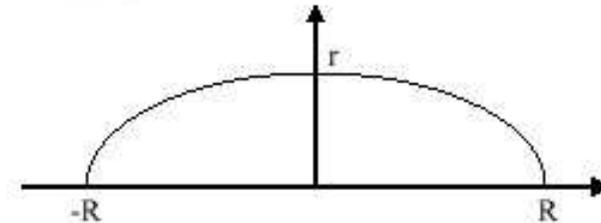
Desenho de Curvas Cúbicas

Algoritmo de Calteljau

```
void DrawCurveRecSub(curve,  $\epsilon$ )
{
    if (Straight(curve,  $\epsilon$ ))
        DrawLine(curve);
    else {
        SubdivideCurve(curve, leftCurve, rightCurve);
        DrawCurveRecSub(leftCurve,  $\epsilon$ );
        DrawCurveRecSub(rightCurve,  $\epsilon$ );
    }
}
```

Exercício

6. Determine as posições dos quatro pontos de controlo de uma curva de Bézier equivalente à elipse da figura junta:
- a)- Analiticamente.
 - b)- Usando métodos baseados no algoritmo de Casteljou.
-



Superfícies Cúbicas

As superfícies cúbicas são uma generalização das curvas cúbicas. A equação da superfície é obtida a partir da equação da curva:

$$Q(t) = T . M . G, \quad \text{sendo } G \text{ constante.}$$

Mudar para a variável s : $Q(s) = S . M . G$

Fazendo variar os pontos do vector Geométrico em 3D ao longo de um percurso parametrizado por t obtém-se:

$$Q(s, t) = S.M.G(t) = S.M. \begin{bmatrix} G_1(t) \\ G_2(t) \\ G_3(t) \\ G_4(t) \end{bmatrix}$$

A matriz geométrica é composta por 16 pontos.

Superfície de Hermite

Para a coordenada x:

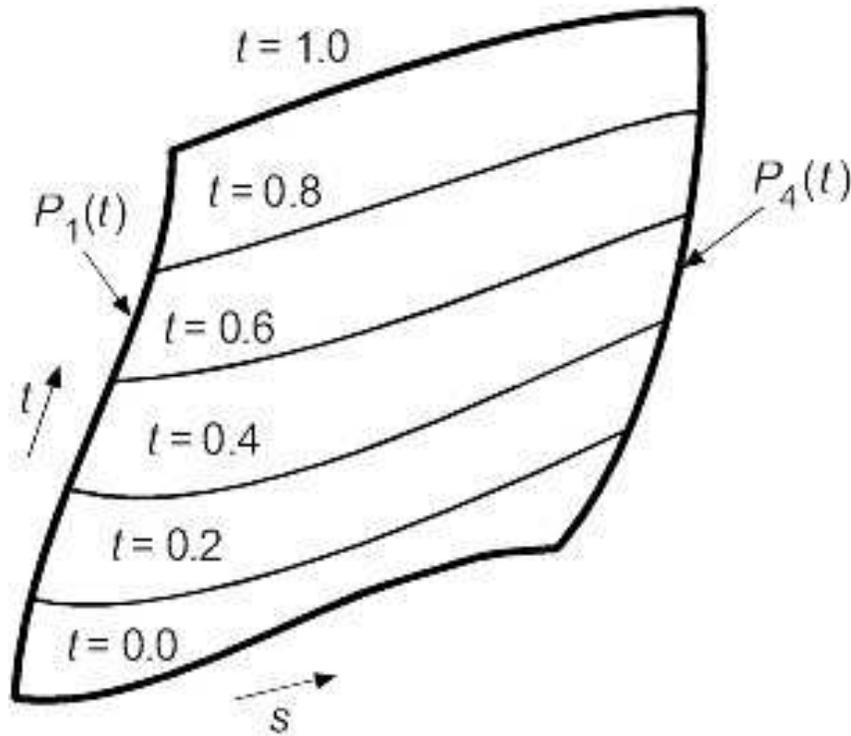
$$x(s,t) = S.M_H.G_{Hx}(t) = S.M_H \cdot \begin{bmatrix} P_1(t) \\ P_4(t) \\ R_1(t) \\ R_4(t) \end{bmatrix}_x$$

$$P_{1x}(t) = T.M_H \cdot \begin{bmatrix} g_{11} \\ g_{12} \\ g_{13} \\ g_{14} \end{bmatrix}_x \quad P_{4x}(t) = T.M_H \cdot \begin{bmatrix} g_{21} \\ g_{22} \\ g_{23} \\ g_{24} \end{bmatrix}_x \quad R_{1x}(t) = T.M_H \cdot \begin{bmatrix} g_{31} \\ g_{32} \\ g_{33} \\ g_{34} \end{bmatrix}_x \quad R_{4x}(t) = T.M_H \cdot \begin{bmatrix} g_{41} \\ g_{42} \\ g_{43} \\ g_{44} \end{bmatrix}_x$$

$$\begin{bmatrix} P_1(t) \\ P_4(t) \\ R_1(t) \\ R_4(t) \end{bmatrix}_x = \underbrace{\begin{bmatrix} g_{11} & g_{12} & g_{13} & g_{14} \\ g_{21} & g_{22} & g_{23} & g_{24} \\ g_{31} & g_{32} & g_{33} & g_{34} \\ g_{41} & g_{42} & g_{43} & g_{44} \end{bmatrix}}_{G_{Hx}} \cdot M_H^T \cdot T^T = G_{Hx} \cdot M_H^T \cdot T^T$$

Conclui-se que: $x(s,t) = S.M_H.G_{Hx}.M_H^T.T^T$

Superfície de Hermite



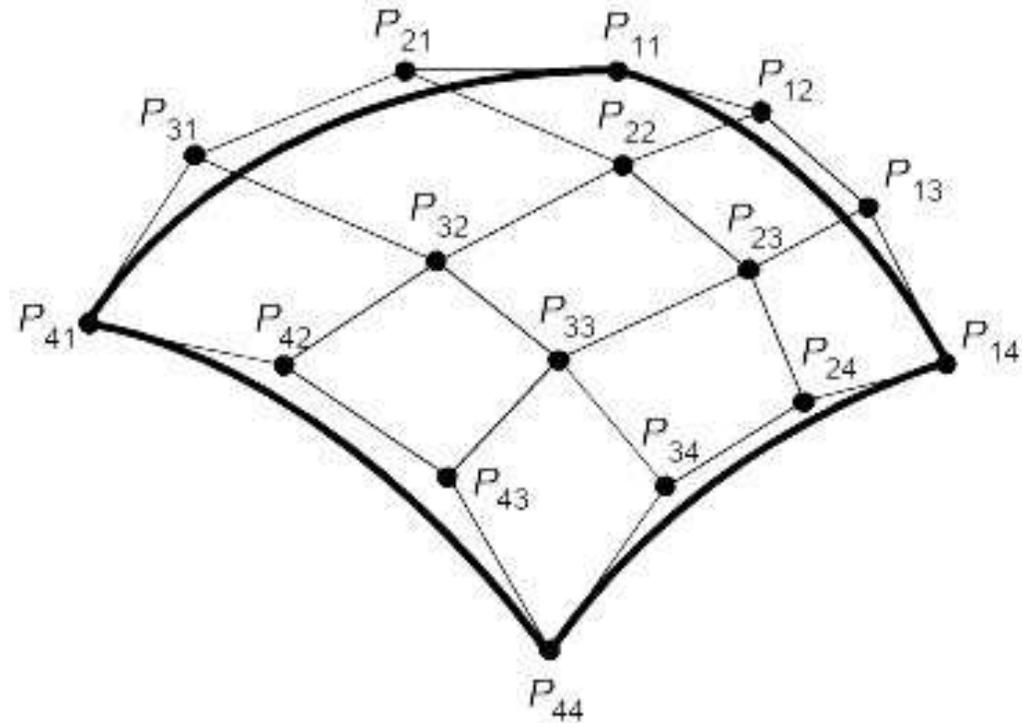
Superfície de Bézier

As equações para a superfície de Bézier podem ser obtidas da mesma forma que as de Hermite, resultando:

$$x(s, t) = S.M_B.G_{Bx}.M_B^T.T^T$$

$$y(s, t) = S.M_B.G_{By}.M_B^T.T^T$$

$$z(s, t) = S.M_B.G_{Bz}.M_B^T.T^T$$



A matriz geométrica tem 16 pontos de controlo.

Superfície de Bézier

Continuidade C^0 e G^0 é obtida fazendo coincidir os quatro pontos de controlo de fronteira: P_{14} , P_{24} , P_{34} , P_{44}

Para obter G^1 devem ser colineares:

P_{13} , P_{14} e P_{15}

P_{23} , P_{24} e P_{25}

P_{33} , P_{34} e P_{35}

P_{43} , P_{44} e P_{45}

e

$$(P_{14} - P_{13}) / (P_{15} - P_{14}) = K$$

$$(P_{24} - P_{23}) / (P_{25} - P_{24}) = K$$

$$(P_{34} - P_{33}) / (P_{35} - P_{34}) = K$$

$$(P_{44} - P_{43}) / (P_{45} - P_{44}) = K$$

