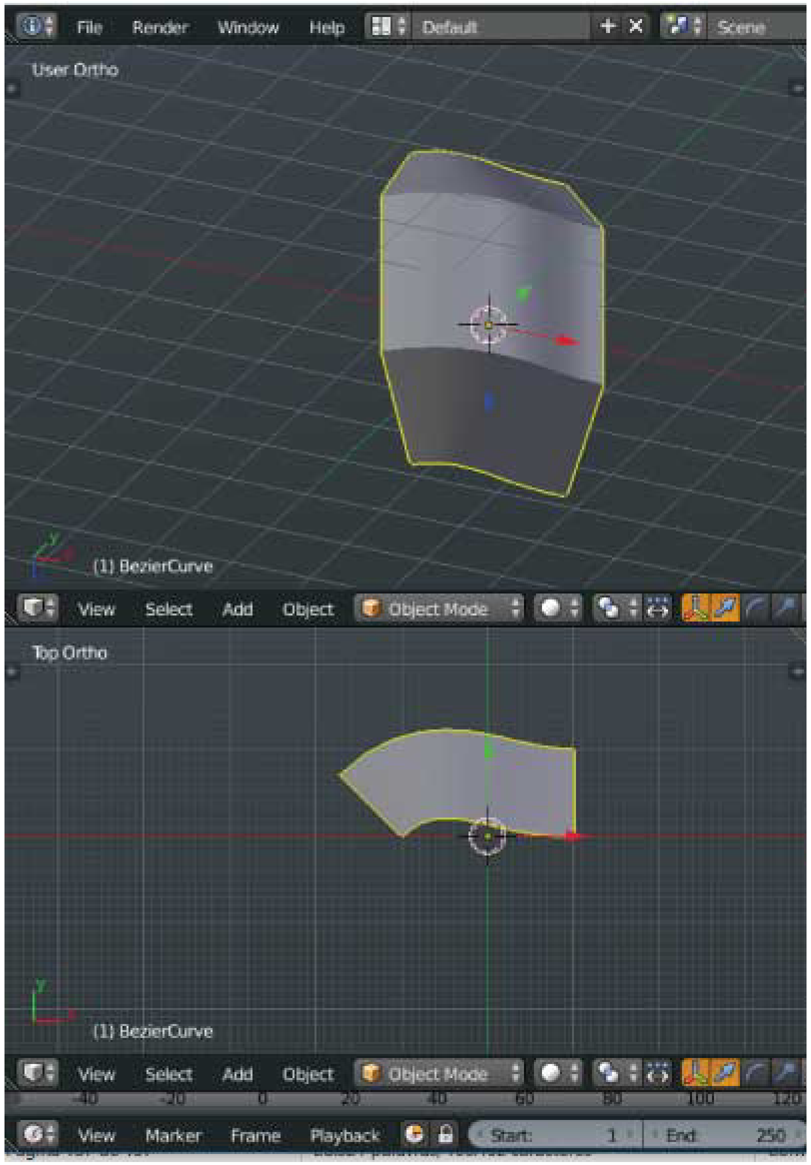
http://computacaografica.ic.uff.br/conteudocap3.html



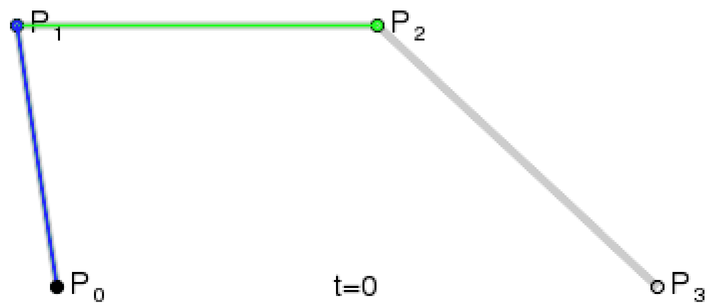
aula 17

IC/UFF – 2019 - 2

Curvas BEZIER

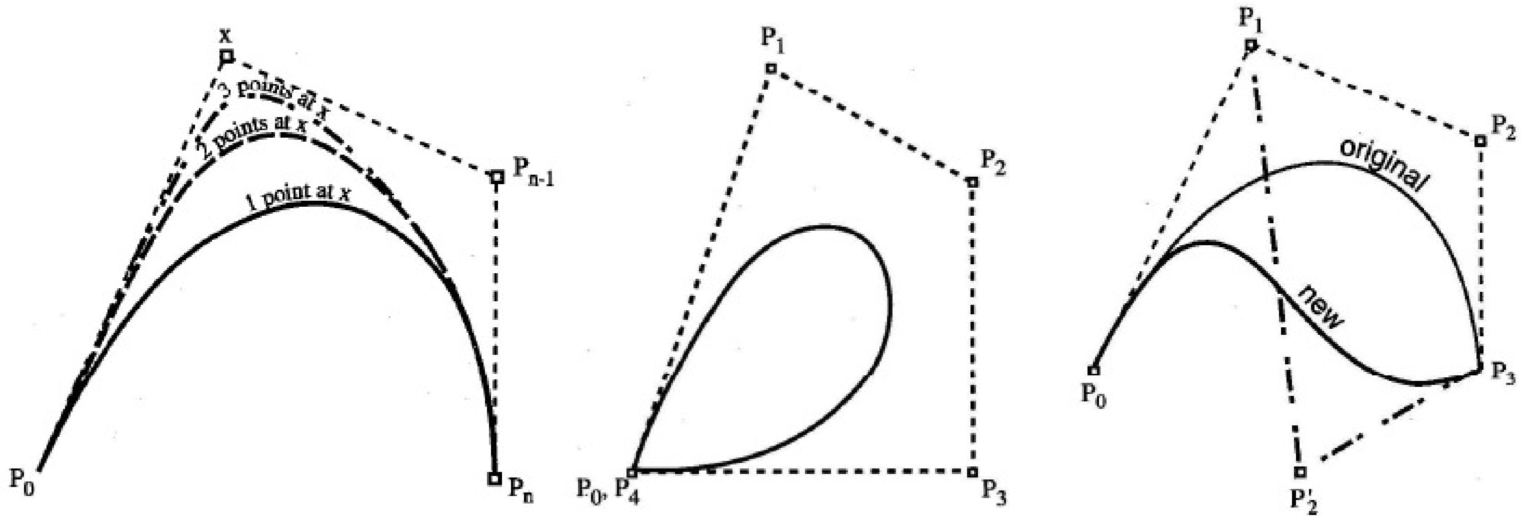
https://pt.wikipedia.org/wiki/Curva_de_Bezier

Curvas de Bezier – parametro, pontos de controle, forma livre.



Curva de Bezier

- Desenvolvidas por Pierre Bezier para descrever o desenho de curvas e superfícies de forma livre
- Polígono definidor
- Primeiro e último ponto
- Vetor tangente

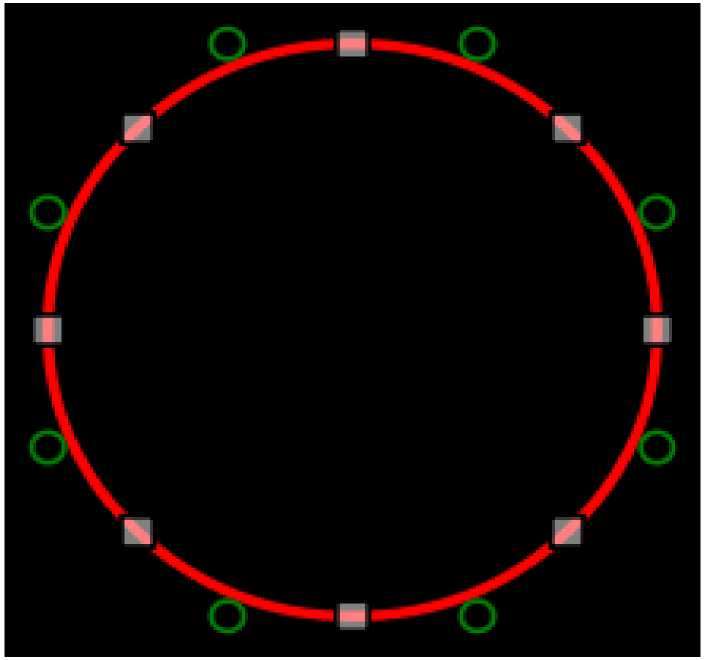


pontos de controle = P_i

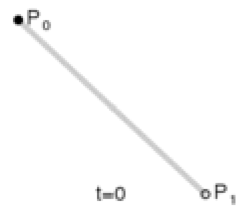
Onde se usa: Qualquer representação de curvas

Até onde você nem imaginar!
Por exemplo:
Os contornos dos caracteres em fontes TrueType são feitas de segmentos de retas e curvas Bézier quadrática.

O circulo ao lado é formado por 8 segmentos. Os quadrados são os pontos de controle da extremidade e os anéis os de controle do interior.

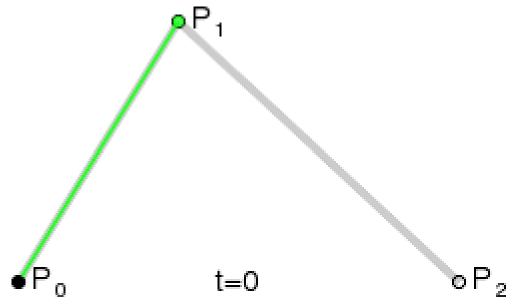


Um segmento linear pode ser definido por Bezier:



pontos de controle = P_0 , P_1

Um segmento de curva quadrática de Bézier é definido por 2 pontos extremos e 1 de controle.



pontos de controle, i =0,1,2 , P_i

Curva polinomial desenvolvida em 1962 por Pierre Bézier.

Utilizada no projeto de automóveis (Renault).

Baseada no algoritmo de De Casteljau em 1957.

Curva de aproximação.

Controle global.

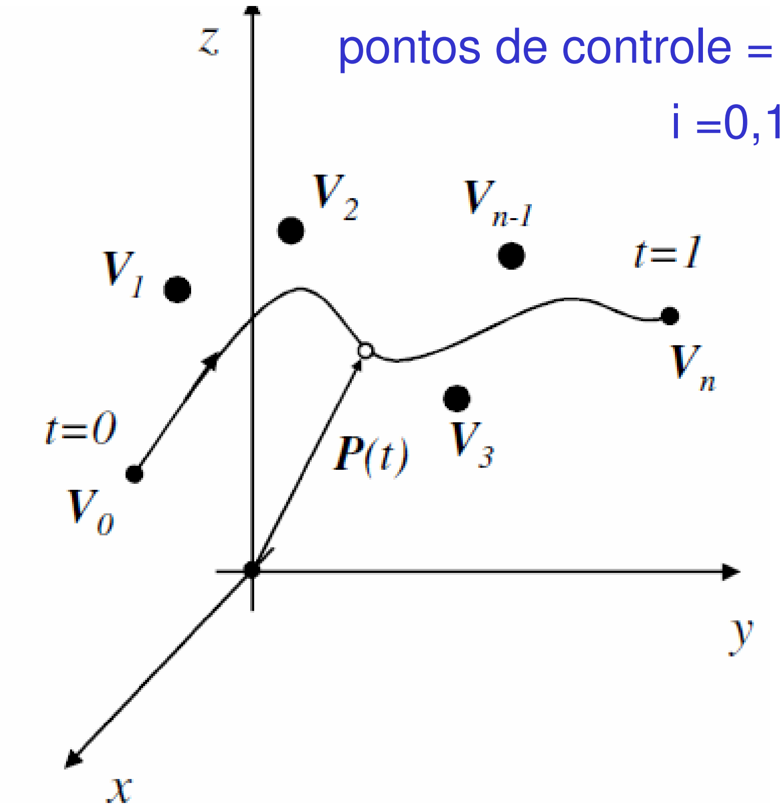
P. de Casteljau, 1959 (Citroën)
P. de Bézier, 1962 (Renault) - UNISURF
Forest 1970: Polinômios de Bernstein

Forma geral pode ter **n+1** pontos de controle, vamos chamar esses agora de V_i e $P(t)$ os pontos da curva:

$$\vec{P}(t) = \sum_{i=0}^{n} B_{i,n}(t)\,\vec{V}_i$$

pontos de controle = *Vi*

i =0,1,2 ,



onde:

$$B_{i,n}(t) = \binom{n}{i} (1-t)^{n-i}\, t^{i}$$

pol. Bernstein

$$\binom{n}{i} = \frac{n!}{i!(n-i)!}$$

coef. binomial

Fatorial de um numero = ***n ! = n (n-1) 1***

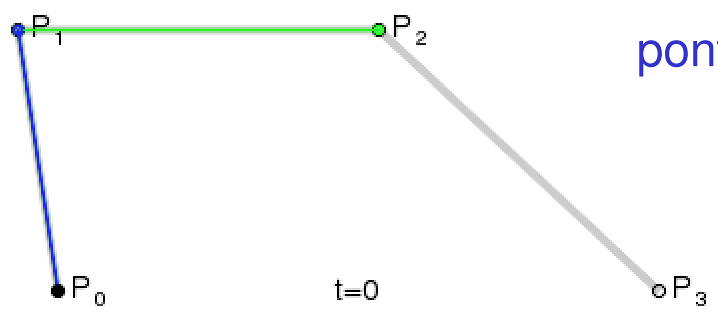
Como vimos na aula anterior

- As cúbicas são especialmente úteis

(porque mesmo ????)

(esqueceu?)

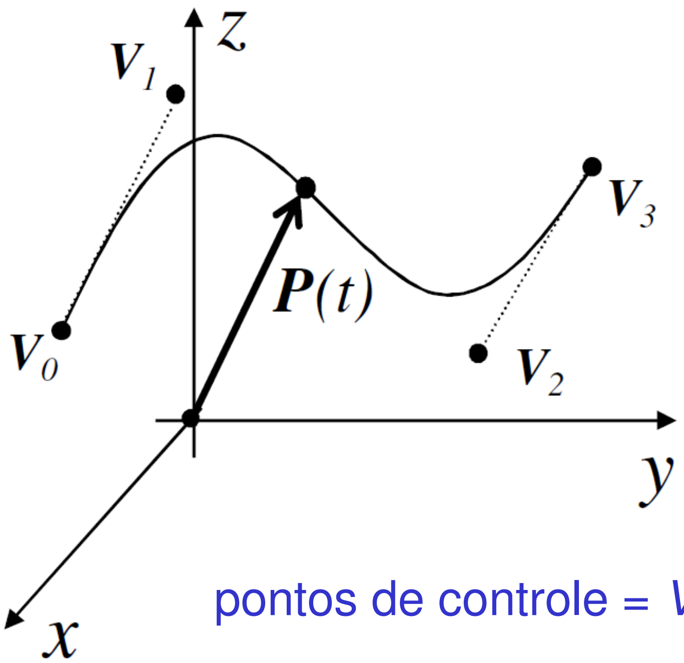
(Vai lá, na aula passada, ver....)



pontos de controle = *Pi*

Bezier cúbica:

$$\vec{P}(t) = \sum_{i=0}^{3} B_{i,3}(t)\,\vec{V}_i$$



pontos de controle = *Vi*

$$B_{0,3}(t) = \binom{3}{0}(1-t)^{3-0}\,t^0 = (1-t)^3$$

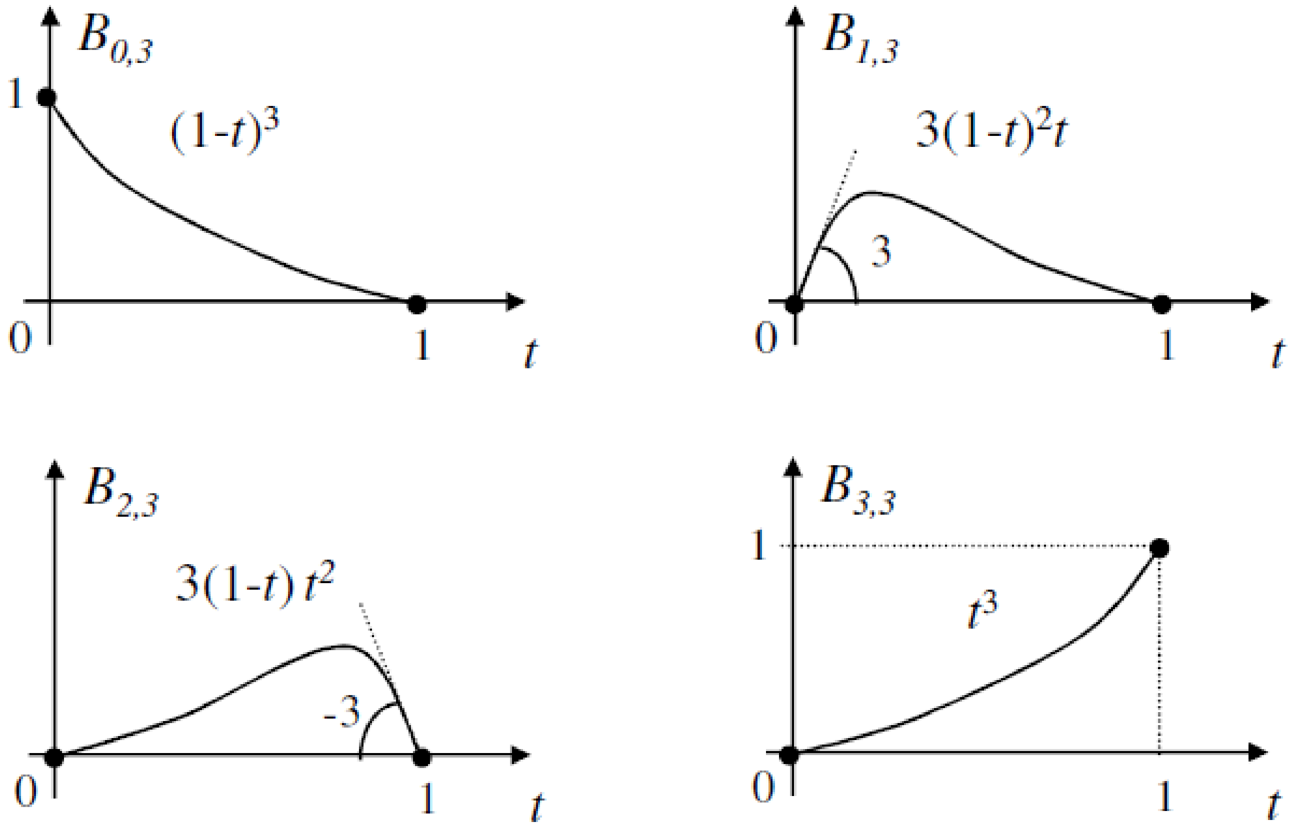
$$B_{1,3}(t) = \binom{3}{1}(1-t)^{3-1}\,t^1 = 3(1-t)^2 t$$

$$B_{2,3}(t) = \binom{3}{2}(1-t)^{3-2}\,t^2 = 3(1-t)t^2$$

$$B_{3,3}(t) = \binom{3}{3}(1-t)^{3-3}\,t^3 = t^3$$

$$\vec{P}(t) = (1-t)^3\,\vec{V}_0 + 3(1-t)^2 t\,\vec{V}_1 + 3(1-t)t^2\,\vec{V}_2 + t^3\,\vec{V}_3$$

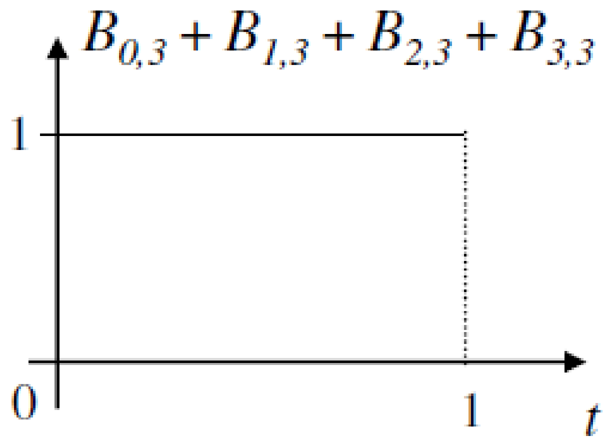
Polinômios cúbicos de **Bernstein**



A soma dos

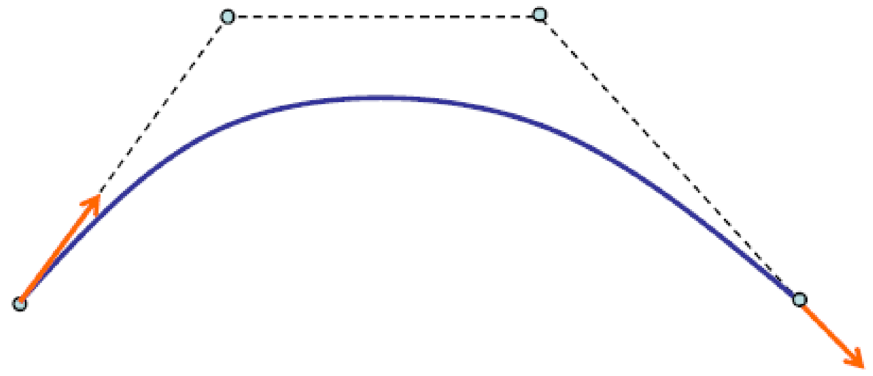
Polinômios Cúbicos de Bernstein

resulta:



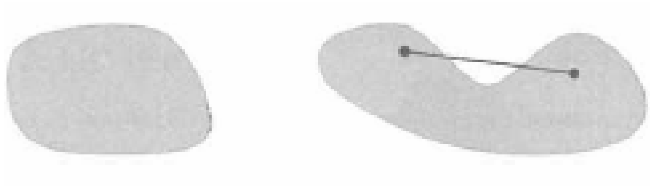
Propriedades de Curva de Bézier

- Continuidade infinita (todas as derivadas são contínuas)
- O grau da curva (do polinômio) é dado pelo número de pontos do polígono de controle menos 1
- A curva de Bézier está contida no fecho convexo do polígono de controle
- A curva interpola o primeiro e último ponto do polígono de controle



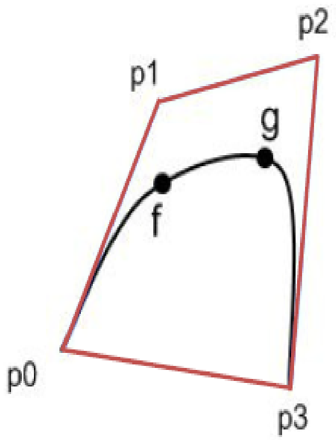
Fecho convexo?

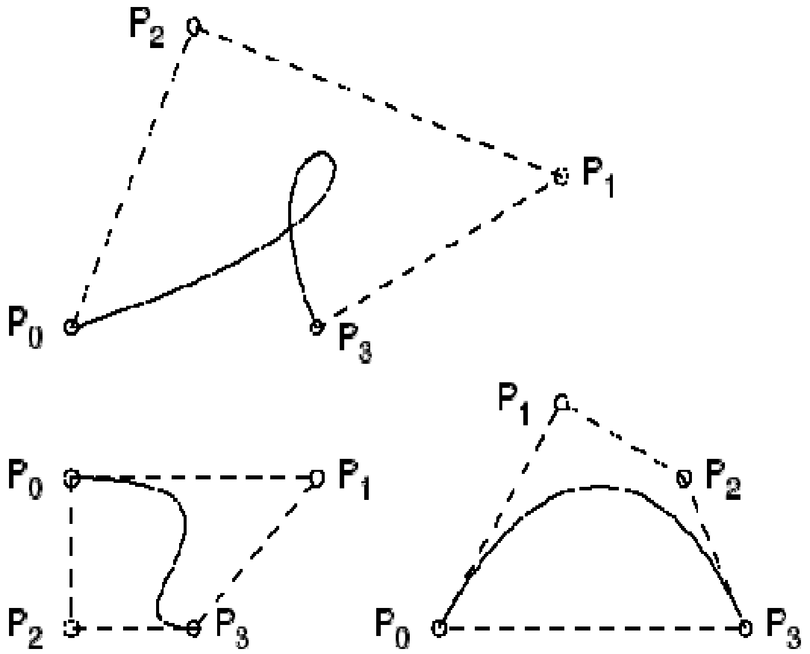
- Como se chamou isso na aula passada???



Propriedade: *Convex Hull*

– Uma curva de Bézier está completamente dentro do maior polígono convexo, formado pelos pontos de controle.



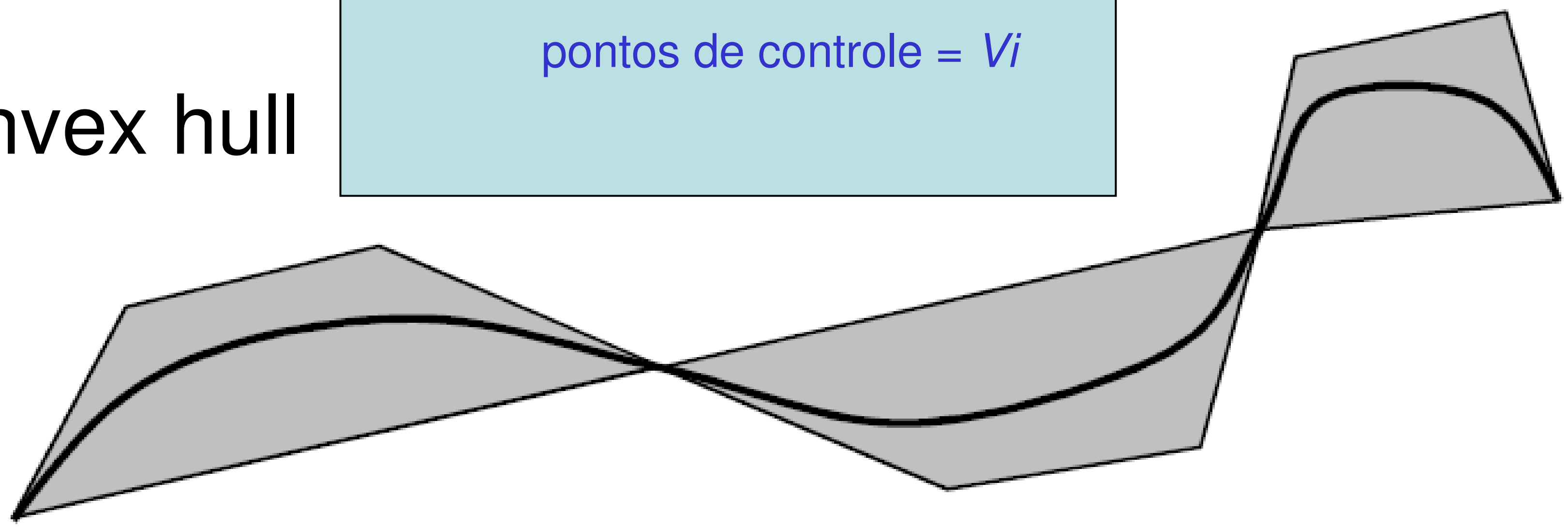


pontos de controle = *Pi*

E para muitas curvas para formar uma única como fica o Fecho convexo?

pontos de controle = *Vi*

Convex hull

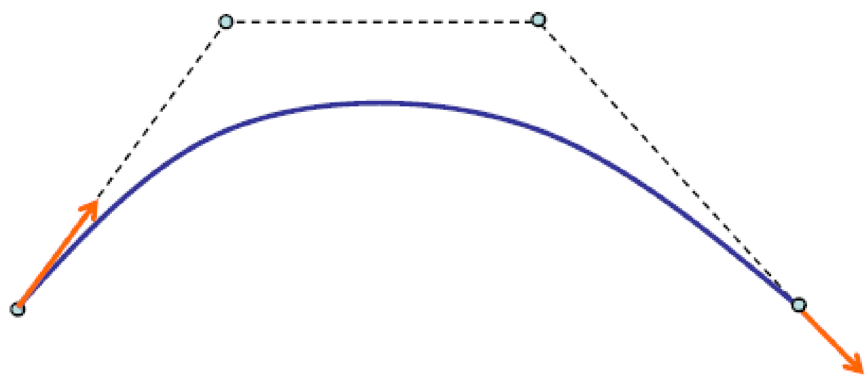


$$\vec{P}(t) = \sum_{i=0}^{n} \alpha_i \vec{V}_i \quad com \quad \sum_{i=0}^{n} \alpha_i = 1$$

Cont.

pontos de controle = *Pi*

- As tangentes à curva em $\mathbf{p}_0$ e $\mathbf{p}_n$ têm a direção dos segmentos de reta $\mathbf{p}_0\mathbf{p}_1$ e $\mathbf{p}_{n-1}\mathbf{p}_n$, respectivamente
  - Para cúbicas, as derivadas são $3(\mathbf{p}_1 - \mathbf{p}_0)$ e $3(\mathbf{p}_2 - \mathbf{p}_3)$
- Transformar os pontos de controle (transf. afim) e desenhar a curva é equivalente a desenhar a curva transformada

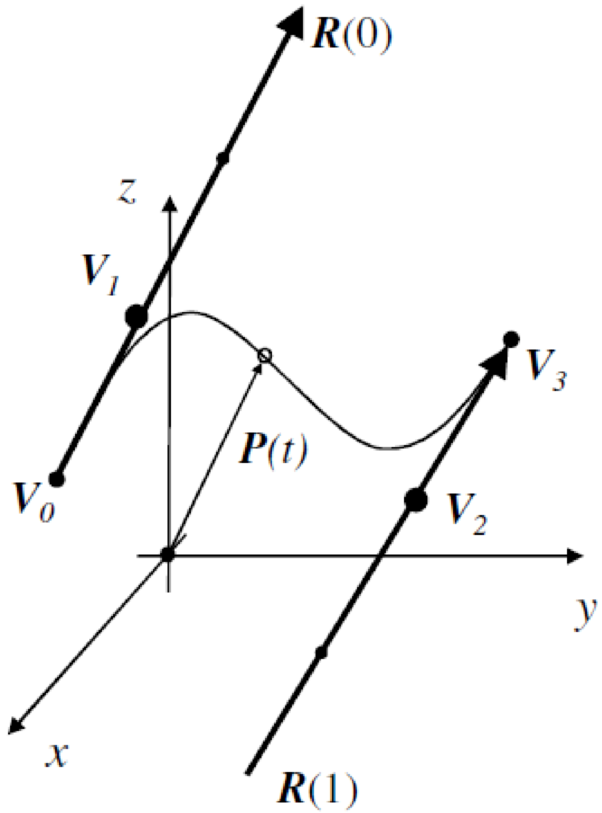


Demonstrando essas propriedades para uma Bezier cúbica:

$$\vec{P}(t) = (1-t)^3\,\vec{V}_0 + 3(1-t)^2 t\,\vec{V}_1 + 3(1-t)t^2\,\vec{V}_2 + t^3\,\vec{V}_3$$

$$\frac{d}{dt}\vec{P}(t) = -3(1-t)^2\,\vec{V}_0 + \left[-6(1-t)t + 3(1-t)^2\right]\vec{V}_1 + \left[-3t^2 + 6(1-t)t\right]\vec{V}_2 + t^3\,\vec{V}_3$$

pontos de controle = *Vi*



$$\vec{P}(0) = \vec{V}_0$$

$$\vec{P}(1) = \vec{V}_3$$

$$\frac{d}{dt}\vec{P}(0) = -3\,\vec{V}_0 + 3\,\vec{V}_1$$

$$\frac{d}{dt}\vec{P}(1) = -3\,\vec{V}_2 + 3\,\vec{V}_3$$

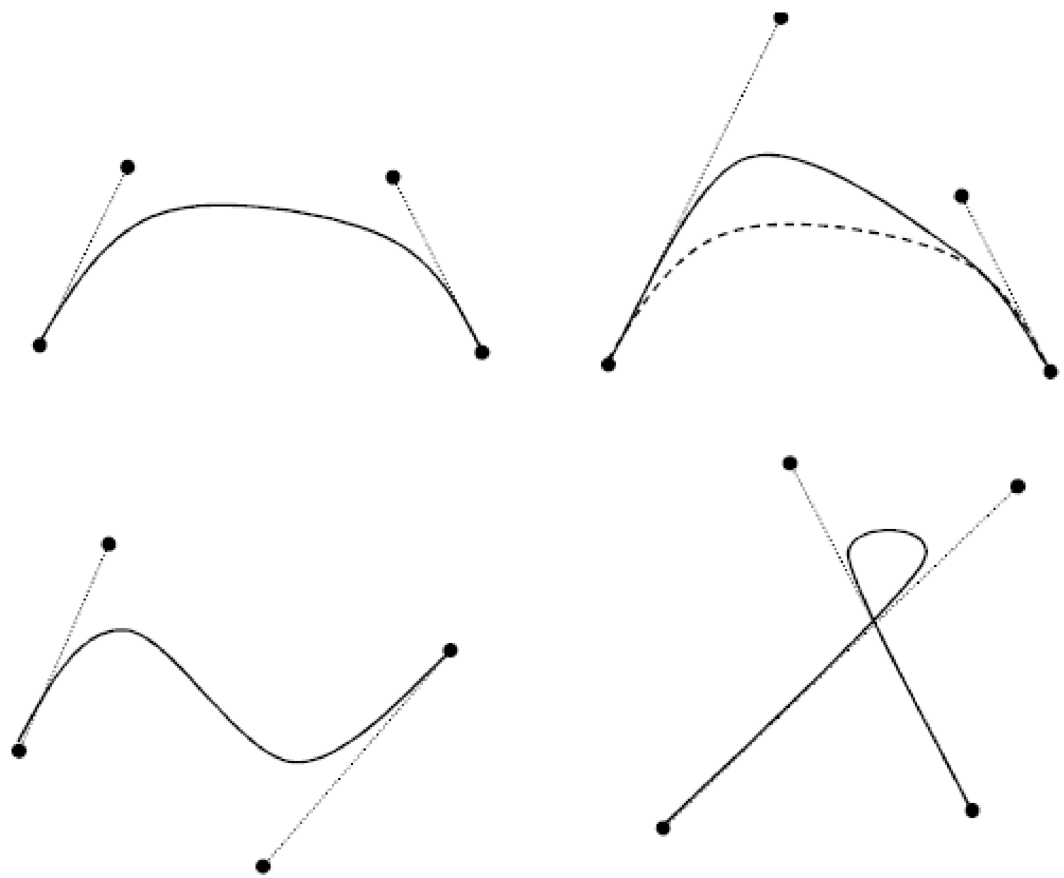
Se fosse pedido para reparametrizar de forma especial

- por exemplo com mais pontos onde a derivada da curva fosse maior, ou ela tivesse maior curvatura ?

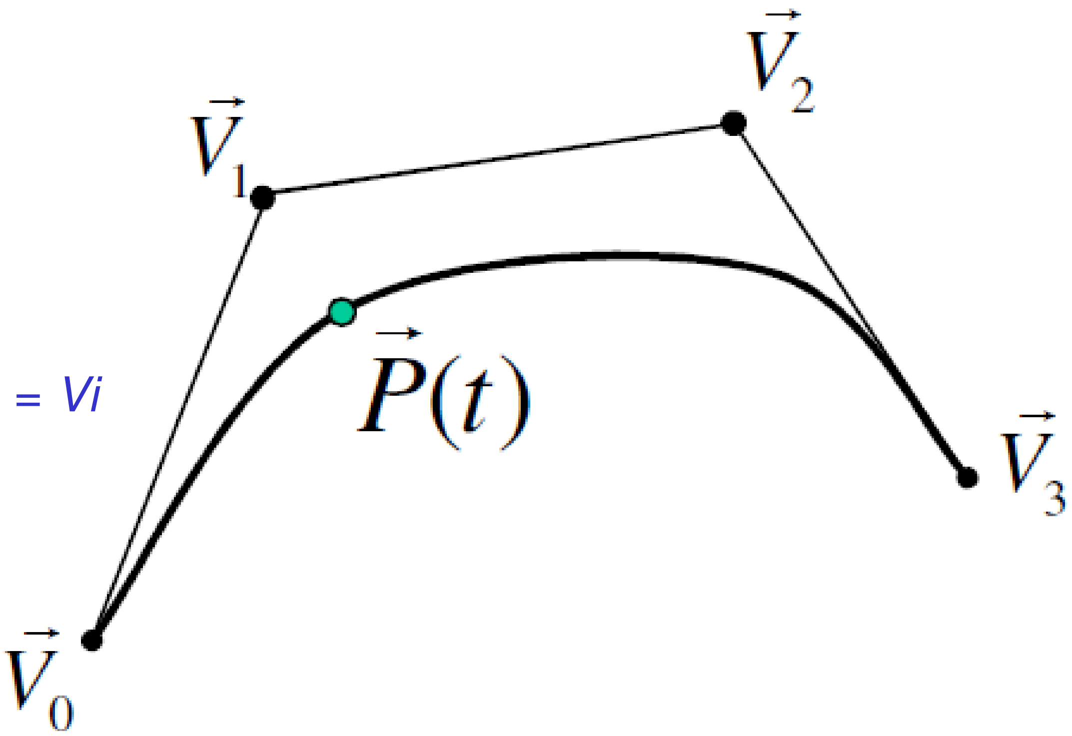
- Com as expressões do slide anterior isso poderia ser feito!

- (simples não??)

A ordem e posição dos pontos controla a curva!



Representação matricial :



pontos de controle = *Vi*

$$\vec{P}(t) = (1-t)^3\,\vec{V}_0 + 3(1-t)^2 t\,\vec{V}_1 + 3(1-t)t^2\,\vec{V}_2 + t^3\,\vec{V}_3$$

$$\vec{P}(t) = \left\langle t^3 \quad t^2 \quad t \quad 1 \right\rangle \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & -3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} V_{0x} & V_{0y} & V_{0z} \\ V_{1x} & V_{1y} & V_{1z} \\ V_{2x} & V_{2y} & V_{2z} \\ V_{3x} & V_{3y} & V_{3z} \end{bmatrix}$$

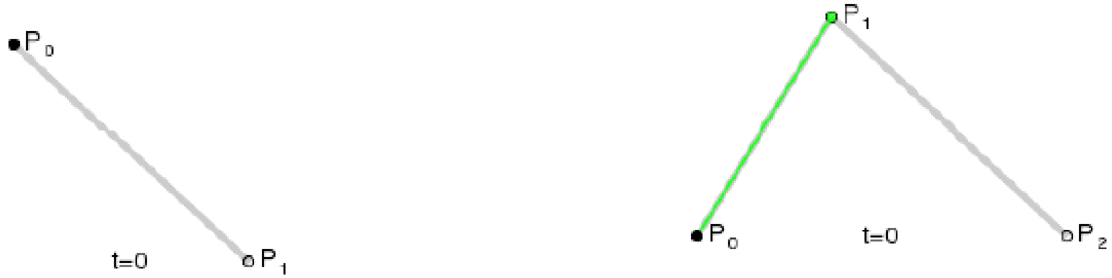
Matriz de Geometria (**G**) e Matriz Base (**M**)

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

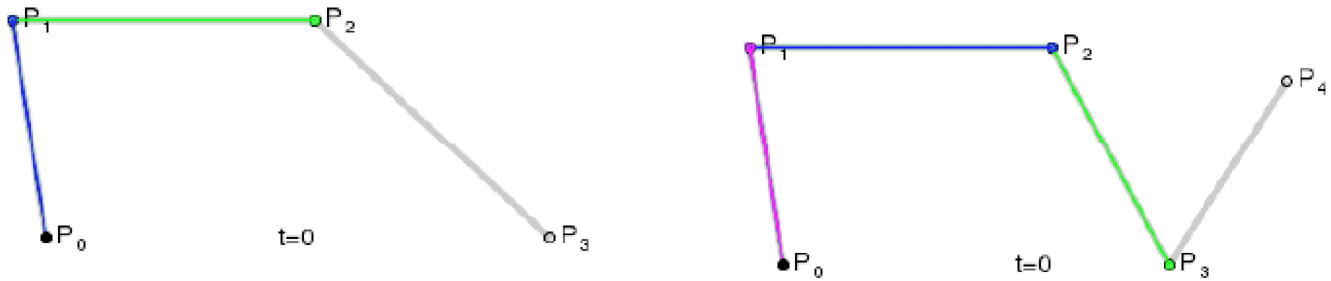
$$Q(t) = TMG$$

pontos de controle = *Gi*

Outras formas de Bezier



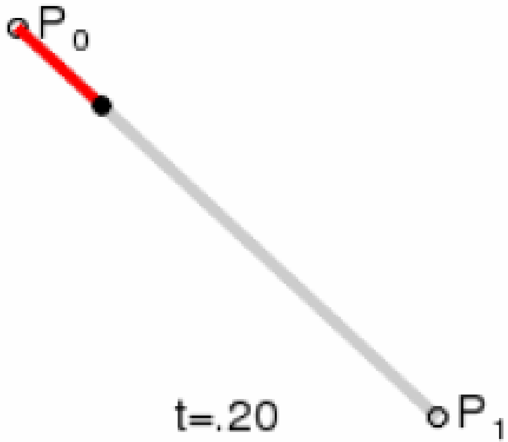
pontos de controle = *Pi*



Curvas de Bézier: 1. linear; 2. quadrática; 3. cúbica; 4. quártica.

Outras formas de Bezier

$$P(t) = (1-t) \,.\, P_0 + t \,.\, P_1$$

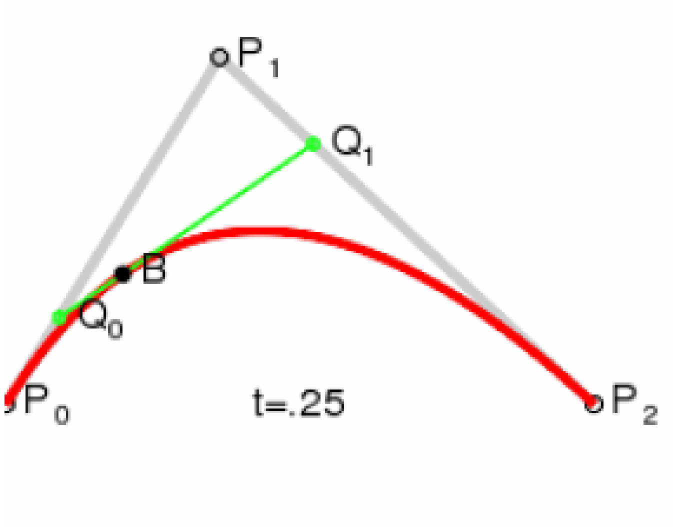


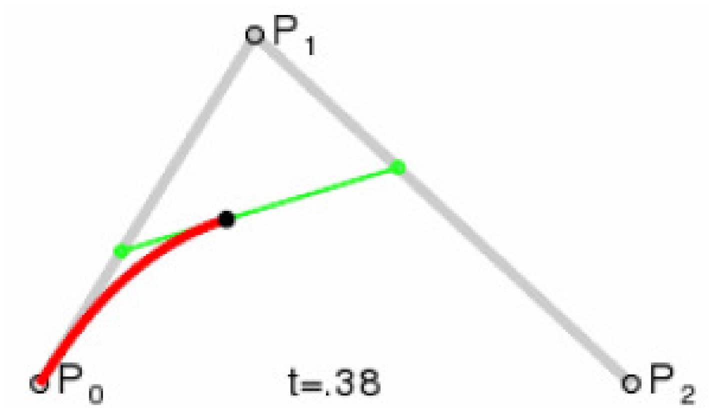
pontos de controle = *Pi*

Outras formas de Bezier

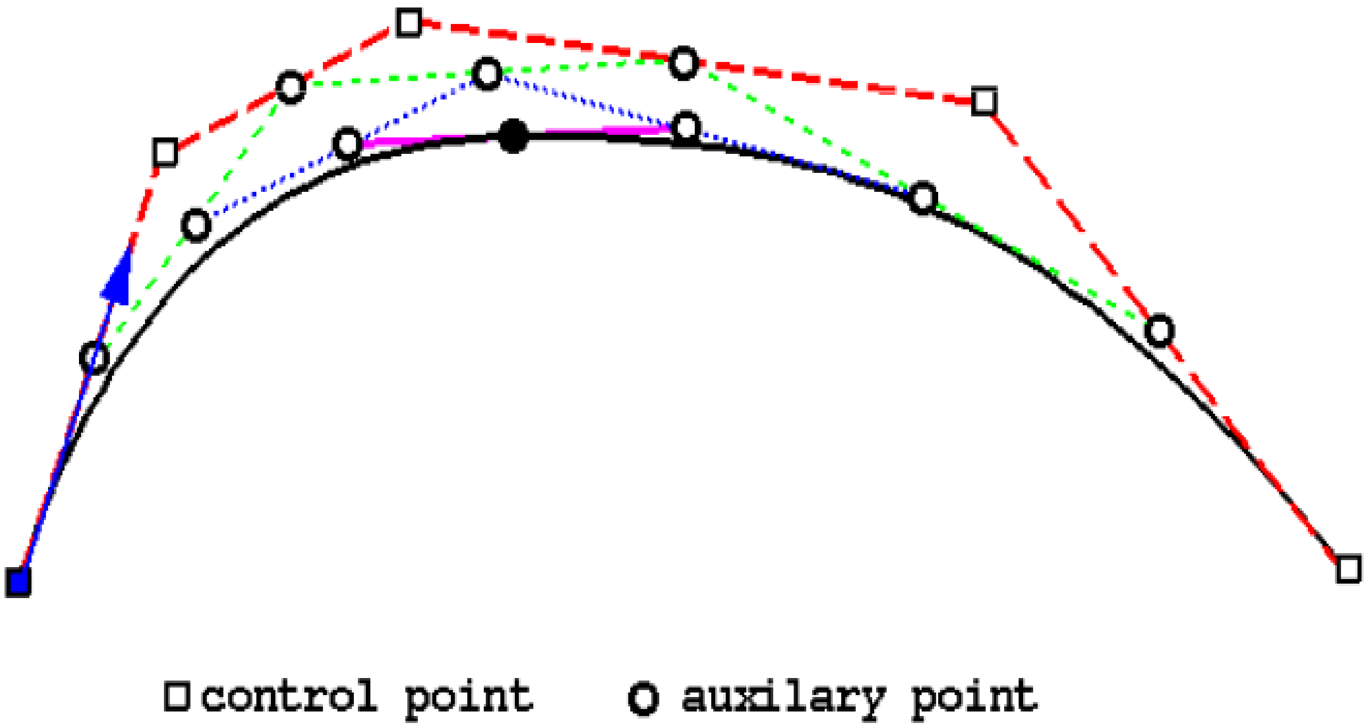
$$P(t) = (1-t)^2\,\mathbf{P}_0 + 2\,t\,(1-t)\,\mathbf{P}_1 + t^2\,\mathbf{P}_2$$

pontos de controle = *Pi*





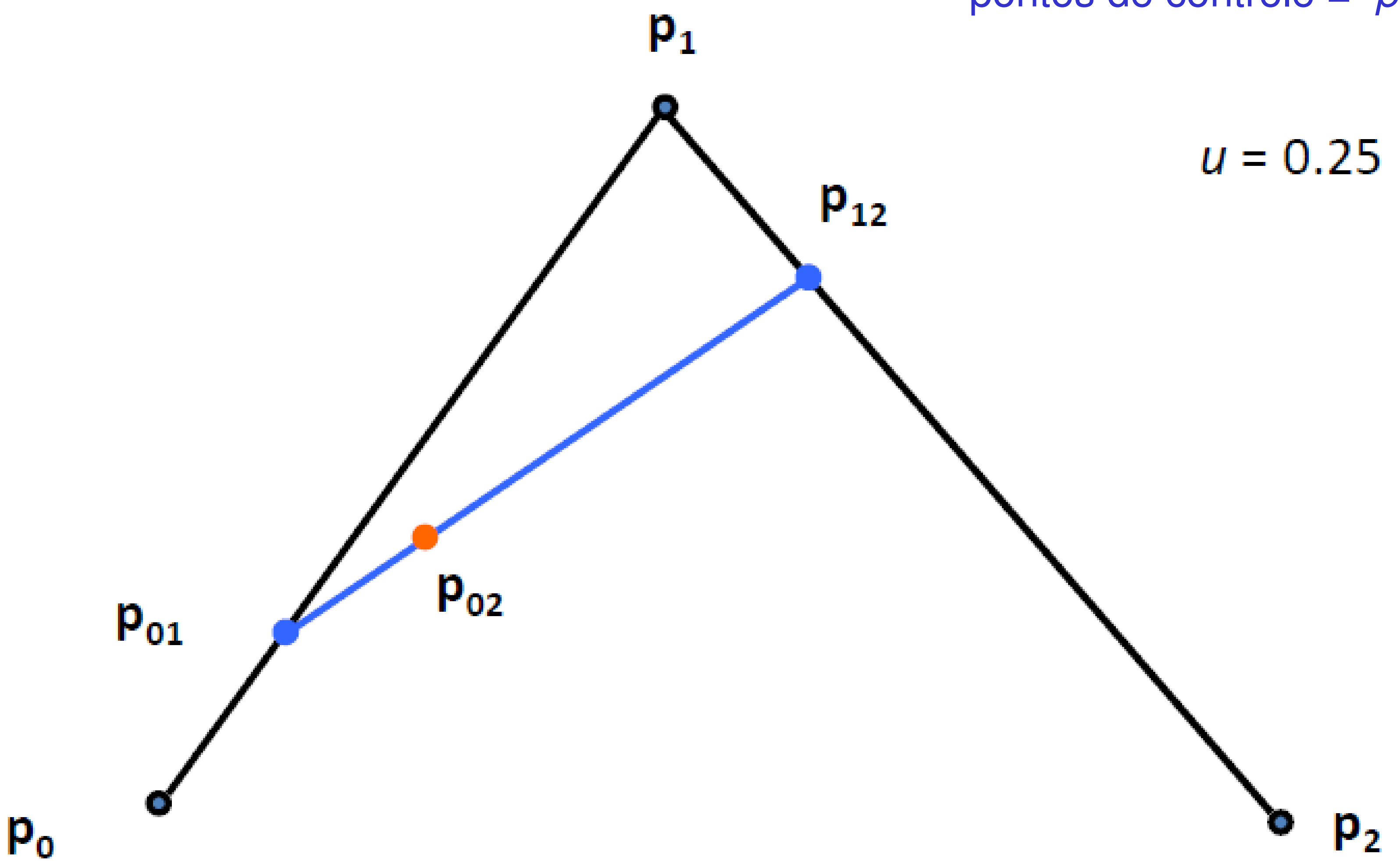
De Casteljau: algoritmo geométrico para construção de curvas Bézier.

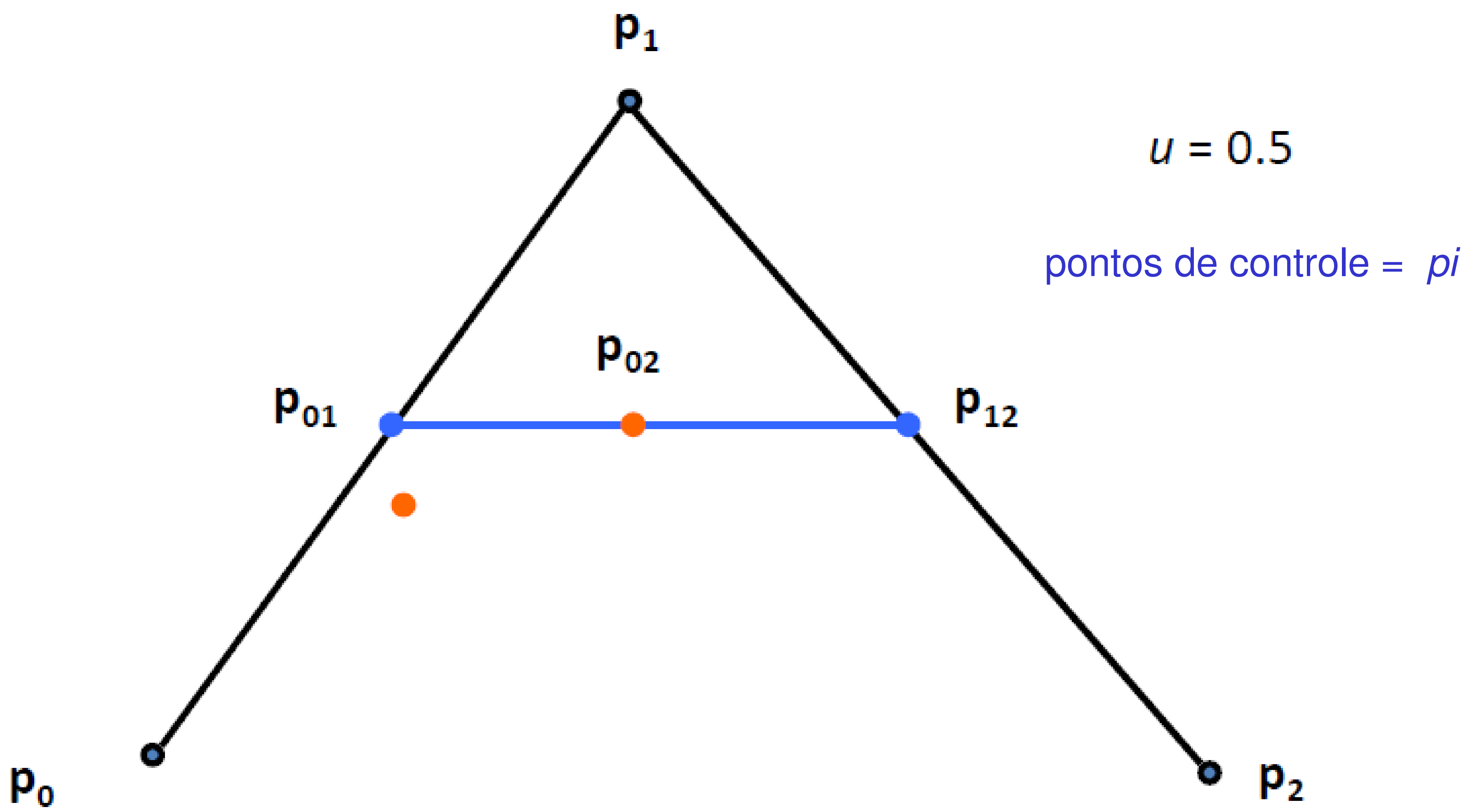


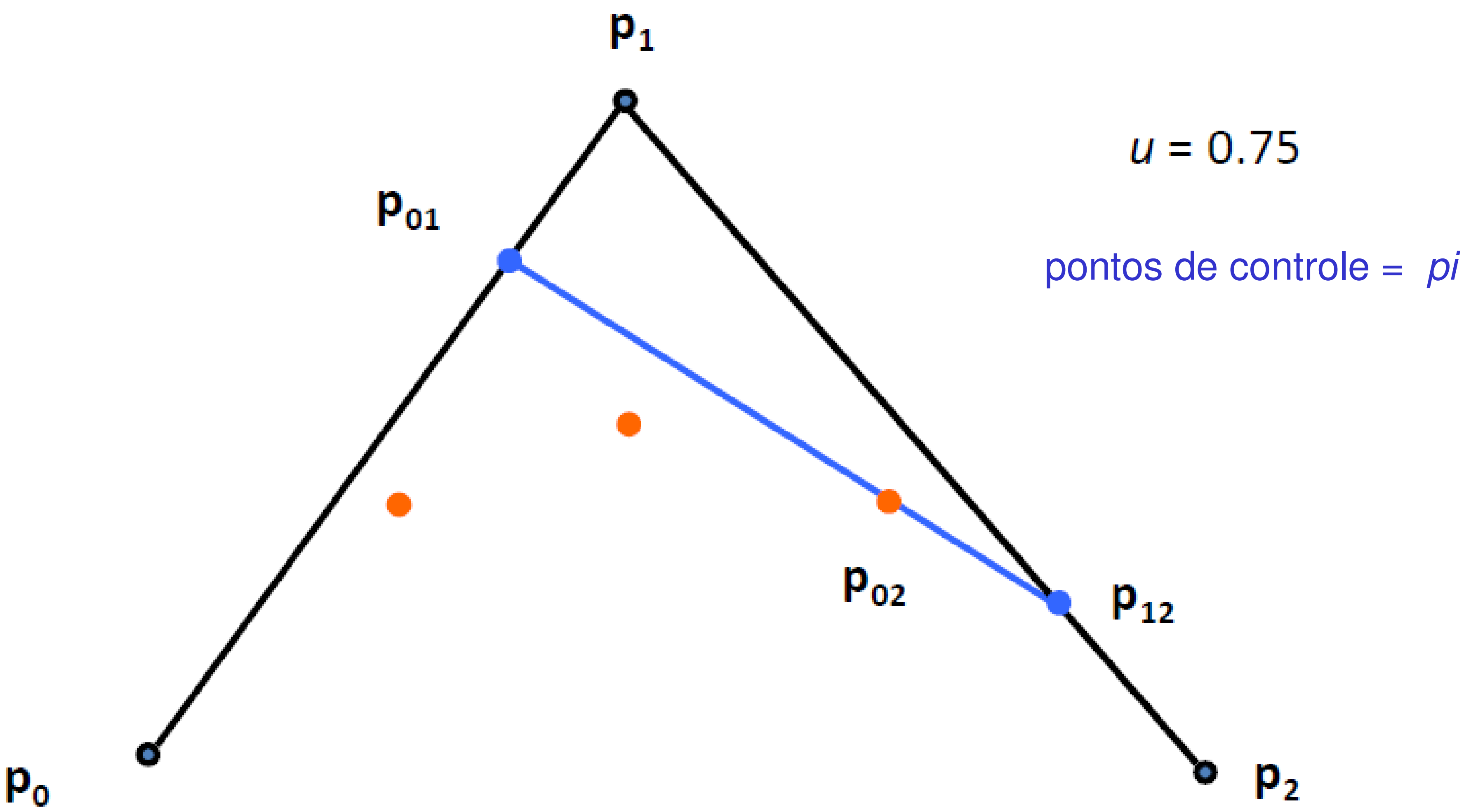
Algoritmo geométrico

pontos de controle = *pi*

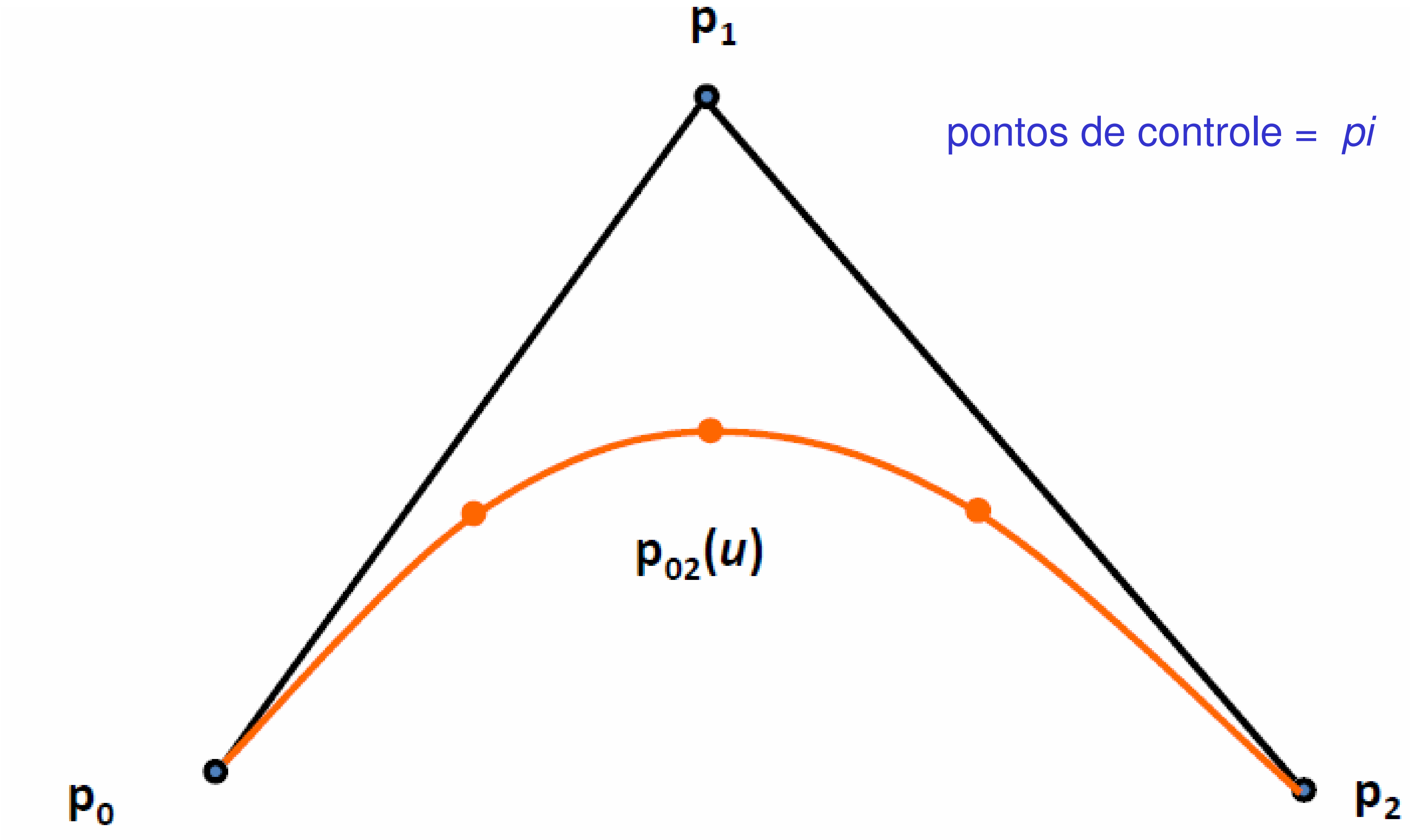
u = 0.25




p_1
$u = 0.5$
pontos de controle = *pi*
p_{02}
p_{01}
p_{12}
p_0
p_2



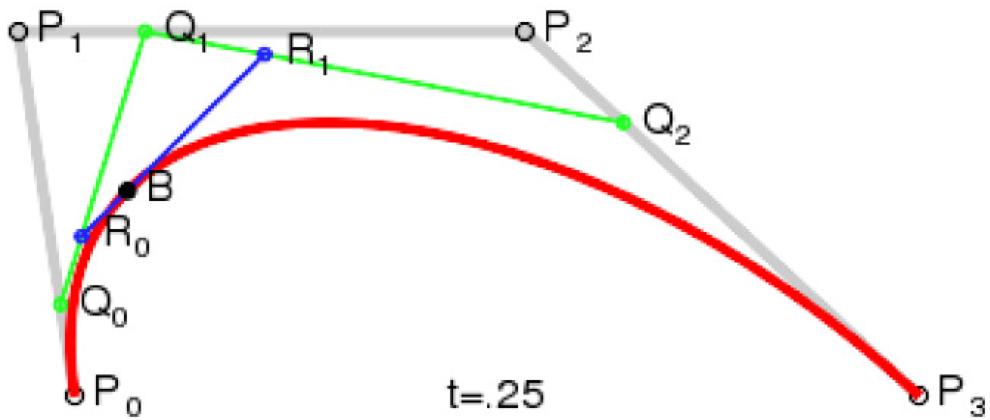
p_1
$u = 0.75$
p_{01}
pontos de controle = *pi*
p_{02}
p_{12}
p_0
p_2



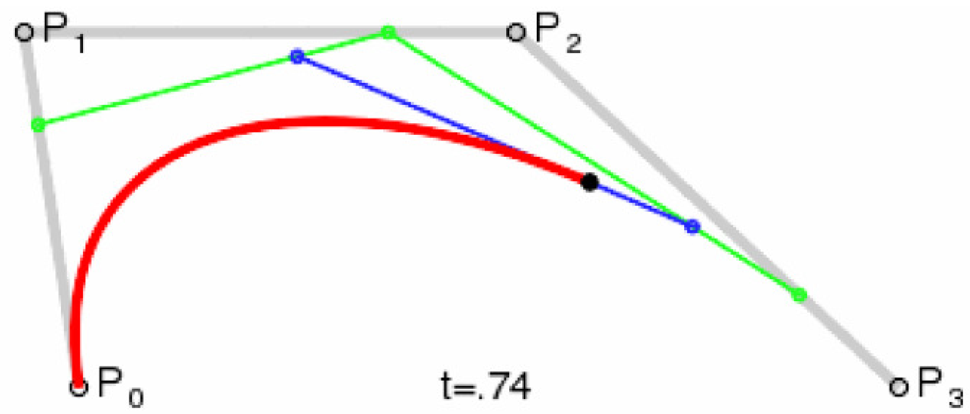
p1
pontos de controle = pi
p02(u)
p0
p2

Outras formas de Bezier

$$P(t) = (1 - t)^3 \mathbf{P}_0 + 3\,t\,(1 - t)^2 \mathbf{P}_1 + 3\,t^2\,(1 - t)\,\mathbf{P}_2 + t^3\,\mathbf{P}_3$$



pontos de controle = *Pi*

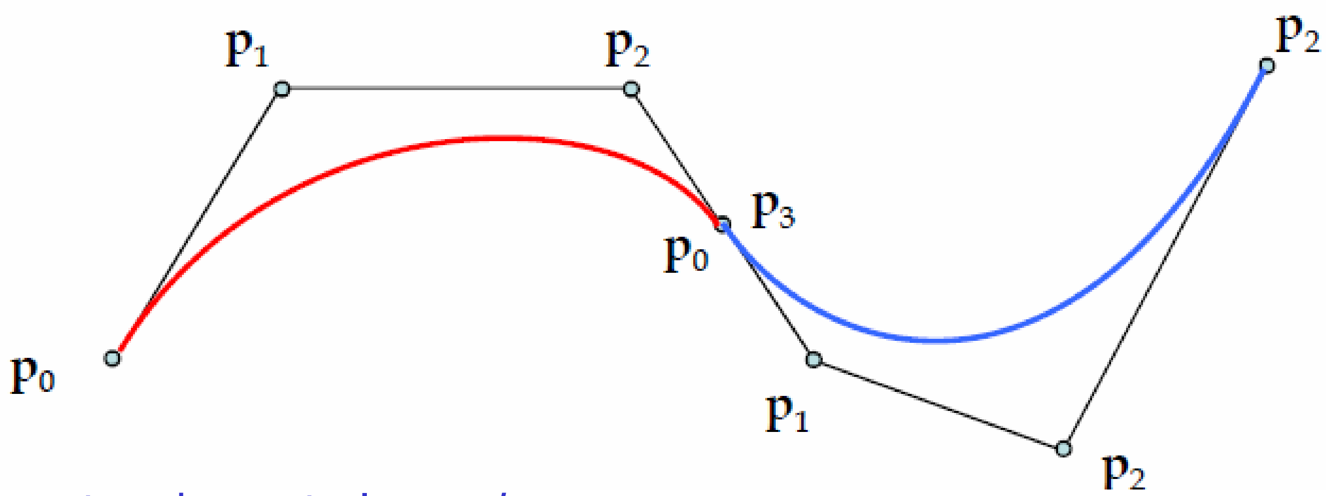


Cont.

- Curvas Bézier com *k* pontos de controle são de grau *k* – 1
- Curvas de grau alto são difíceis de desenhar
  - Complexas
  - Sujeitas a erros de precisão
- Normalmente, queremos que pontos de controle tenham efeito *local*
  - Em curvas Bézier, todos os pontos de controle têm efeito *global*
- Solução:
  - Emendar curvas polinomiais de grau baixo
  - Relaxar condições de continuidade

Emendando Curvas Bézier

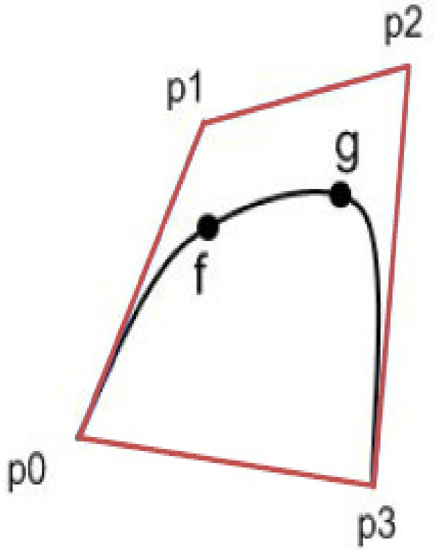
- Continuidade C^0: Último ponto da primeira = primeiro ponto da segunda
- Continuidade C^1: C^0 e segmento $\mathbf{p}_2\mathbf{p}_3$ da primeira com mesma direção e comprimento que o segmento $\mathbf{p}_0\mathbf{p}_1$ da segunda
- Continuidade C^2: C^1 e + restrições sobre pontos $\mathbf{p}_1$ da primeira e $\mathbf{p}_2$ da segunda

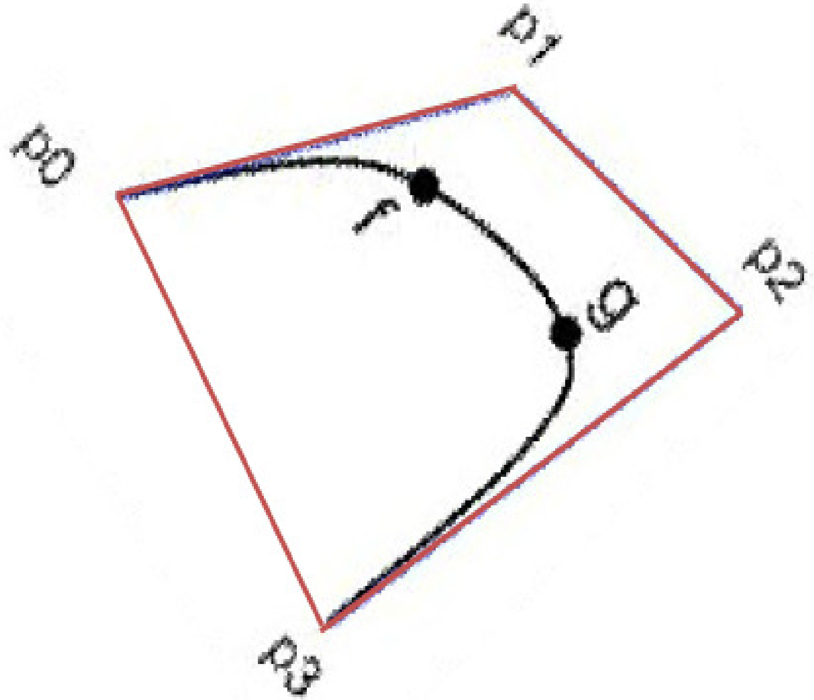


pontos de controle = *pi*

Transformações

- Executar as transformações (S,R,T) na curva é equivalente a realizar as transformações nos pontos de controle.

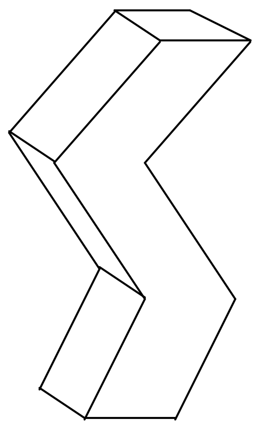




pontos de controle = *pi*

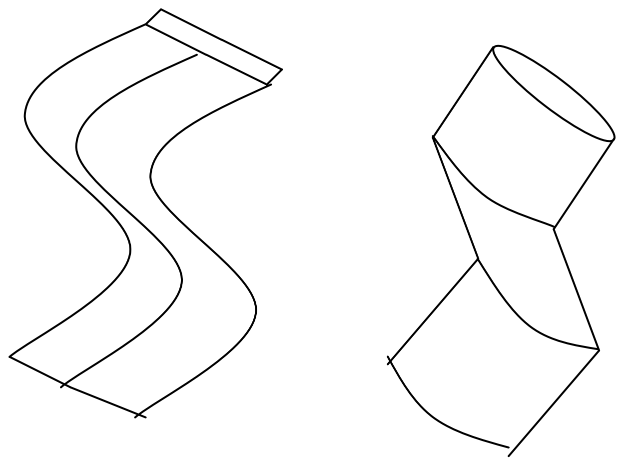
Trabalho 2 – parte 1

- Inclua a opção do objeto usado no trabalho 1 ser opaco, ou seja inclua em sua implementação um dos possiveis métodos apresentados de eliminação de partes invisíveis. (Use o mesmo programa do trabalho 1, e faça isso quando a animação dele acabar, ele ficará parado, daqui para a frente no trabalho 2)



Trabalho 2 – parte 2

- Inclua a opção do seu objeto ser formado por pelo menos uma curva e não mais só linhas. Você pode fazer isso como achar mais adequado. Abaixo alguns exemplos.



Socorro!!

Como faço isso???

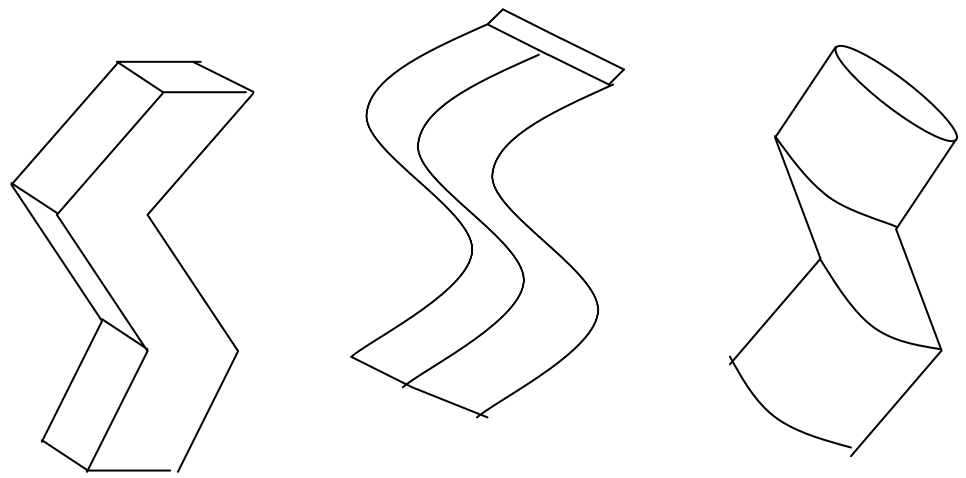
Implementado e equação abaixo em loop, para t= 0, 1, 0,01:

$$\vec{P}(t) = (1-t)^3\,\vec{V}_0 + 3(1-t)^2 t\,\vec{V}_1 + 3(1-t)t^2\,\vec{V}_2 + t^3\,\vec{V}_3$$

Você faz isso para cada curva que resolver fazer. No final armazena os pontos de controle. Os pontos iniciais , finais , e os intermediários em coordenadas (3D) no espaço do objeto são os pontos de controle. Com os pontos de controle a curva de Bezier pode ser desenhada!! (e até poderia ser animada como vez com as retas)

Trabalho 2 – parte 3

- Escolha uma das opções do seu objeto estático (em linhas ou com curvas) e faça o shading mais adequado a ele , considerando que ele tem a cor em RGB: 70, 60,140.



O assunto da P2

- É até a aula de hoje
- E , por mim :
  - Ela pode mesmo ser na próxima quinta (14/07) como havíamos marcado desde o inicio do semestre!
  - E por vocês?