Universidade de São Paulo
Instituto de Matemática e Estatística

Introdução à Computação Gráfica

Marcel P. Jackowski
mjack@ime.usp.br

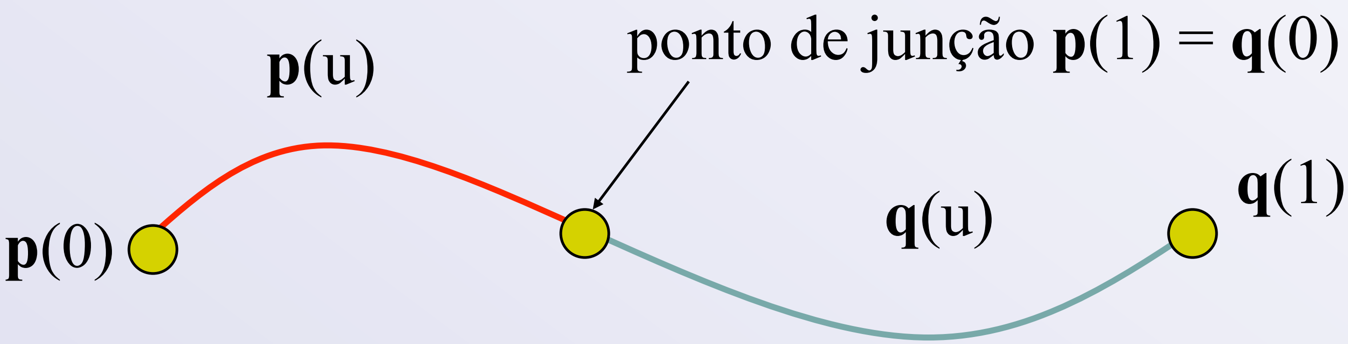
Aula #15

Objetivos

- Curvas paramétricas
  - Motivação: animação da câmera
  - Continuidade paramétrica (C^k)
  - Continuidade geométrica (G^k)
- Curvas polinomiais
- Curvas de Bézier
  - Propriedades

Segmentos de curva

- Uma curva paramétrica pode ser escrita como:
  $\mathbf{p}(u)=[x(u), y(u), z(u)]^T$, $1 <= u <= 0$
- Normalmente desenhamos uma curva que possui suporte global
- Em CG e CAD, é mais viável desenhar pequenos segmentos de curva que são interconectados



Curvas polinomiais paramétricas

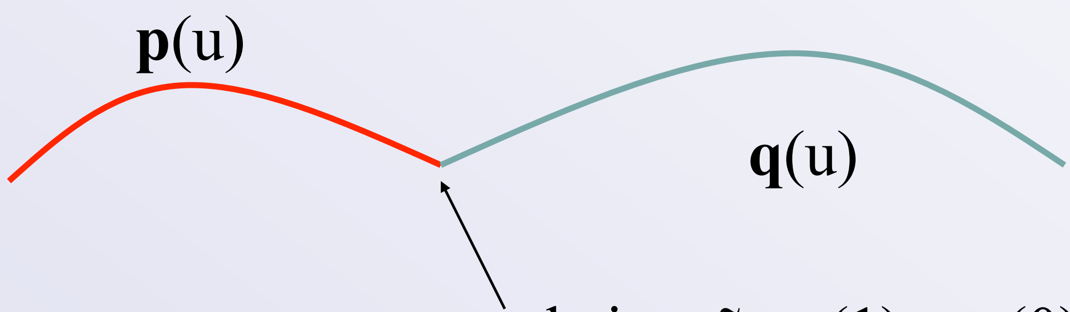
$$x(u) = \sum_{i=0}^{N} c_{xi} u^i \quad y(u) = \sum_{j=0}^{M} c_{yj} u^j \quad z(u) = \sum_{k=0}^{L} c_{zk} u^k$$

- Se N=M=K, precisamos determinar 3(N+1) coeficientes
- Cada uma das curvas para x, y e z são independentes e podem ser definidas de maneira idêntica

- Usaremos a forma $\mathrm{p}(u) = \sum_{k=0}^{L} c_k u^k$ onde p pode ser x, y, z

Porquê polinômios ?

- Fáceis de calcular
- São contínuas e diferenciáveis em todo o seu domínio
  - Devemos nos preocupar com a continuidade nos pontos de junção



ponto de junção $\mathbf{p}(1) = \mathbf{q}(0)$
mas $\mathbf{p}'(1) \mathrel{!}= \mathbf{q}'(0)$

Curvas polinomiais cúbicas

- Quando N=M=L=3, resulta em facilidade de avaliação e flexibilidade no design

$$p(u) = \sum_{k=0}^{3} c_k u^k$$

- Quatro coeficientes são necessários para definir x, y and z
- Achar quatro condições independentes para vários valores de u que resultarão em 4 equações com 4 variáveis para cada x, y and z
  - Tais condições são uma mistura de requisitos de <u>continuidade nos pontos de junção</u> e <u>condições de representação dos dados</u>

Superfícies paramétricas cúbicas

$$\mathbf{p}(\mathrm{u,v})=[\mathrm{x(u,v),\ y(u,v),\ z(u,v)}]^T$$

onde

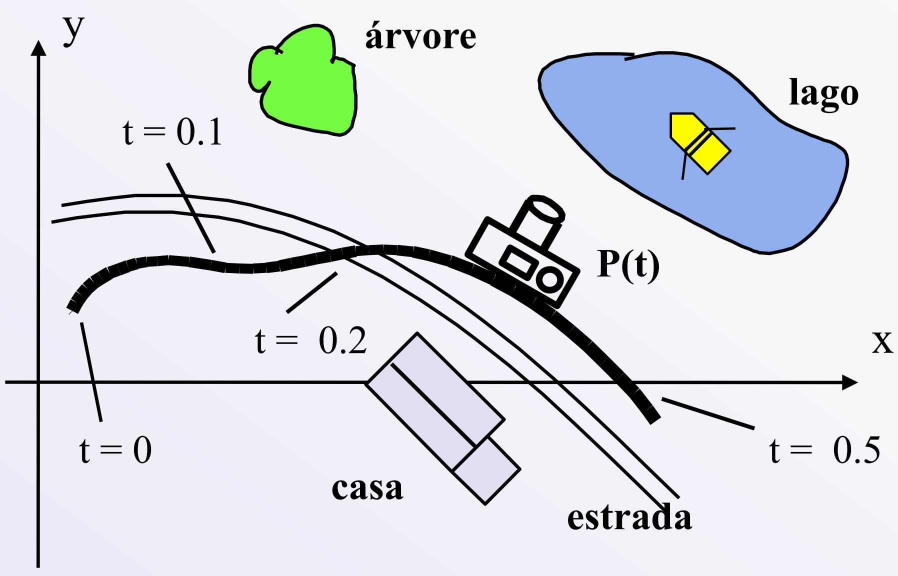
$$\mathrm{p}(u,v)=\sum_{i=0}^{3}\sum_{j=0}^{3} c_{ij}u^i v^j$$

e p representa x, y or z

Precisamos de 48 coeficientes (3 conjuntos independentes de 16 coeficientes) para determinar um retalho ("patch") da superfície.

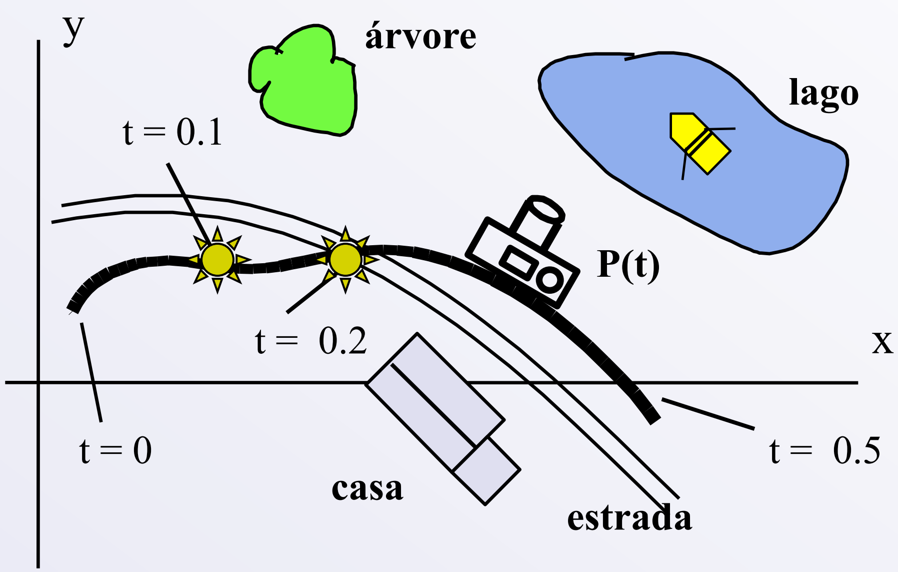
Animação

- A trajetória da câmera em uma cena deve ser especificada a cada instante de tempo.
- Ela está localizada no ponto *P*(*t*) no tempo *t*.



Animação (ii)

- Escolhemos uma função *P*(*t*) de forma que a câmera se mova conforme a trajetória desejada;
- Esta câmera, por exemplo, pode tirar fotos da cena em tempos $t = 0.1$, $t = 0.2$, etc.
- A direção de visualização deve também ser especificada a cada instante.



Animação (iii)

- A câmera deve se deslocar na trajetória $P(t)$ de forma suave, sem movimentos bruscos.
  - Isto impõe uma restrição na velocidade $\mathbf{P}'(t)$.
- Outros objetos também poderão se mover na cena: o carro, o barco, pessoas saindo da casa, etc.
- O movimento destes objetos pode ser descrito através de funções paramétricas $F(t)$, $G(t)$, etc, apropriadas.

Suavidade de Movimento

- A **velocidade v**(t) é um vetor que descreve a velocidade e direção de um objeto se movendo ao longo da trajetória $P(t)$.
- É caracterizada pela primeira derivada da trajetória $P(t)$:

$$\bar{v} = \frac{dP(t)}{dt} = \left(\frac{dx}{dt}, \frac{dy}{dt} \right)$$

Suavidade de Movimento (ii)

- A **reta tangencial** à curva $P(t)$ em $t = t_0$ (em forma paramétrica) é $L(u)$.
- Ela passa através de $P(t_0)$ quando $u = 0$ e move-se na direção $\mathbf{v}(t_0)$:
  - $L(u) = P(t_0) + \mathbf{v}(t_0)\, u$

Suavidade de Movimento (iii)

- A direção **normal** à curva pode também ser determinada em cada ponto.
- Ela é definida como a direção perpedicular à linha tangencial em cada ponto.
- Se a linha tangencial possui direção $\mathbf{v}(t_0)$ no tempo t_0, a direção normal em t_0 é múltipla do vetor
  - $\mathbf{n}(t_0) = \mathbf{v}^{\perp}(t_0) = (-dy/dt, dx/dt)|_{t = t0}$

Continuidade Paramétrica (C^k)

- Dizemos que uma curva $P(t)$ tem continuidade paramétrica de ordem k no intervalo de t E $[a, b]$ se todas as derivadas da curva, até grau k, existem e são contínuas no intervalo $[a, b]$.
- Dizemos, então que, P(t) tem suavidade k no intervalo t E $[a, b]$.
- Para evitar movimentos abruptos em animações, utilizaremos curvas com suavidade 1.

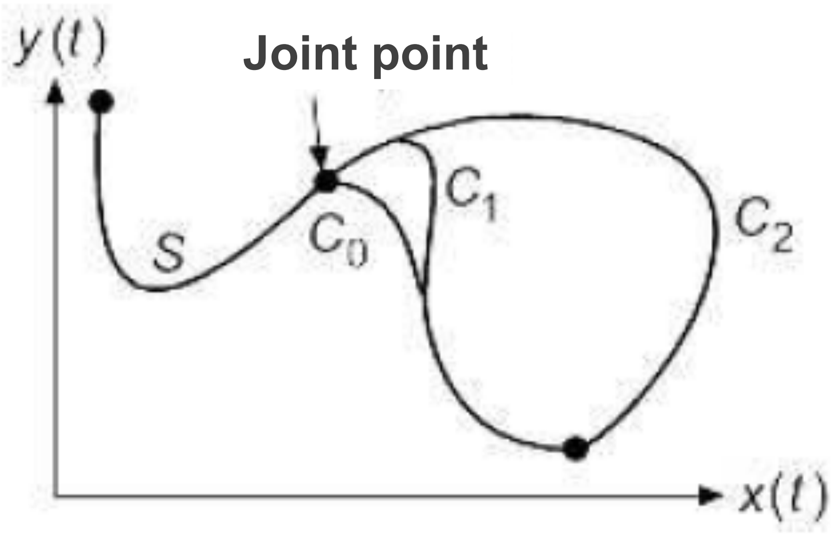
Suavidade

- Uma curva é dita ser **suave-0** em um intervalo se ela for contínua (C^0) naquele intervalo.
- Uma curva é dita **suave-1** se sua primeira derivada existir e for contínua (C^1) em um dado intervalo.
- Uma curva é dita **suave-2** se sua primeira e segunda derivada existir e forem contínuas (C^2) no intervalo.
- Analogamente, uma curva é chamada **suave-3** se sua primeira, segunda e terceira derivadas existirem e forem contínuas (C^3).
  - Uma curva suave-3 deve também ser suave-2, mas uma curva suave-2 pode não ser suave-3.

Continuidade Geométrica (G^k)

- A continuidade G^0 é igual a C^0: $P(t)$ é contínuo em t no intervalo [a, b].
- Continuidade G^1 em [a, b] implica que P'(t-dt) = k P'(t+dt) para alguma constante k e para todo c no intervalo [a,b].
- Continuidade G^2 em [a, b] implica que P'(t-dt) = k P'(t+dt) e P''(t+dt) = m P''(t-dt) para as constantes k e m e para todo c no intervalo [a, b].

Continuidade: Exemplo 1

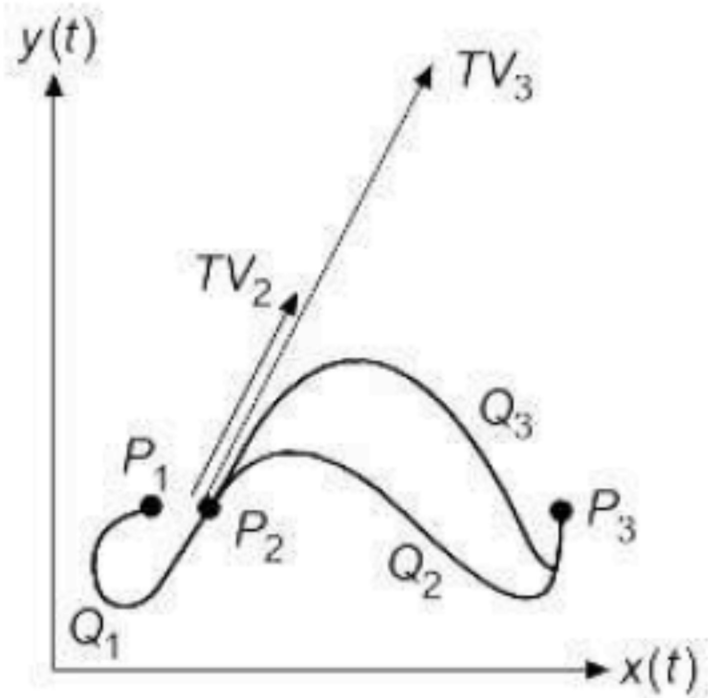


No ponto de junção da curva ***S*** com as curvas ***C_0***, ***C_1*** e ***C_2*** temos:

Continuidade G^0 entre ***S*** e ***C_0***
Continuidade C^1 entre ***S*** e ***C_1***
Continuidade C^2 entre ***S*** e ***C_2***

Continuidade: Exemplo 2

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 polinomiais

- Um polinômio de grau k é representado pela função

  $P(t) = \mathbf{a_k} t^k + \mathbf{a_{k-1}} t^{k-1} + \ldots + \mathbf{a_1} t + \mathbf{a_0}$,

  onde $\{\mathbf{a_k}, \mathbf{a_{k-1}}, \ldots, \mathbf{a_0}\}$ são os coeficientes.
- O **grau** de um polinômio: maior potência de *t* (k).
- A **ordem** de um polinômio: o número de seus coeficientes (k+1).

Polinômios de 1º. grau

- $P(t) = a_0 + a_1 t$, uma curva linear (linha reta).
  - A representação P(t) contém (em 2D) 2 equações, uma para x(t) e uma para y(t).
  - Em 3D, temos três equações, uma para cada x(t), y(t), z(t).
  - P(t) passa pelo ponto a_0 quando $t = 0$, e pelo ponto a_1 quando $t = 1$.

Polinômios de 2º. grau

- $x(t) = \mathbf{a}t^2 + \mathbf{b}t + \mathbf{c}$, $y(t) = \mathbf{d}t^2 + \mathbf{e}t + \mathbf{f}$
  - Para qualquer escolha de ***a***, ***b***, ***c***, ***d***, ***e***, e ***f***, esta curva representa uma parábola.
  - Não conseguiremos gerar uma elipse ou hipérbole utilizando esta forma.
- Podemos usar a função implícita
  - $F(x, y) = \mathbf{A}x^2 + \mathbf{2B}xy + \mathbf{C}y^2 + \mathbf{D}x + \mathbf{E}y + \mathbf{F} = 0$ para gerar cônicas.

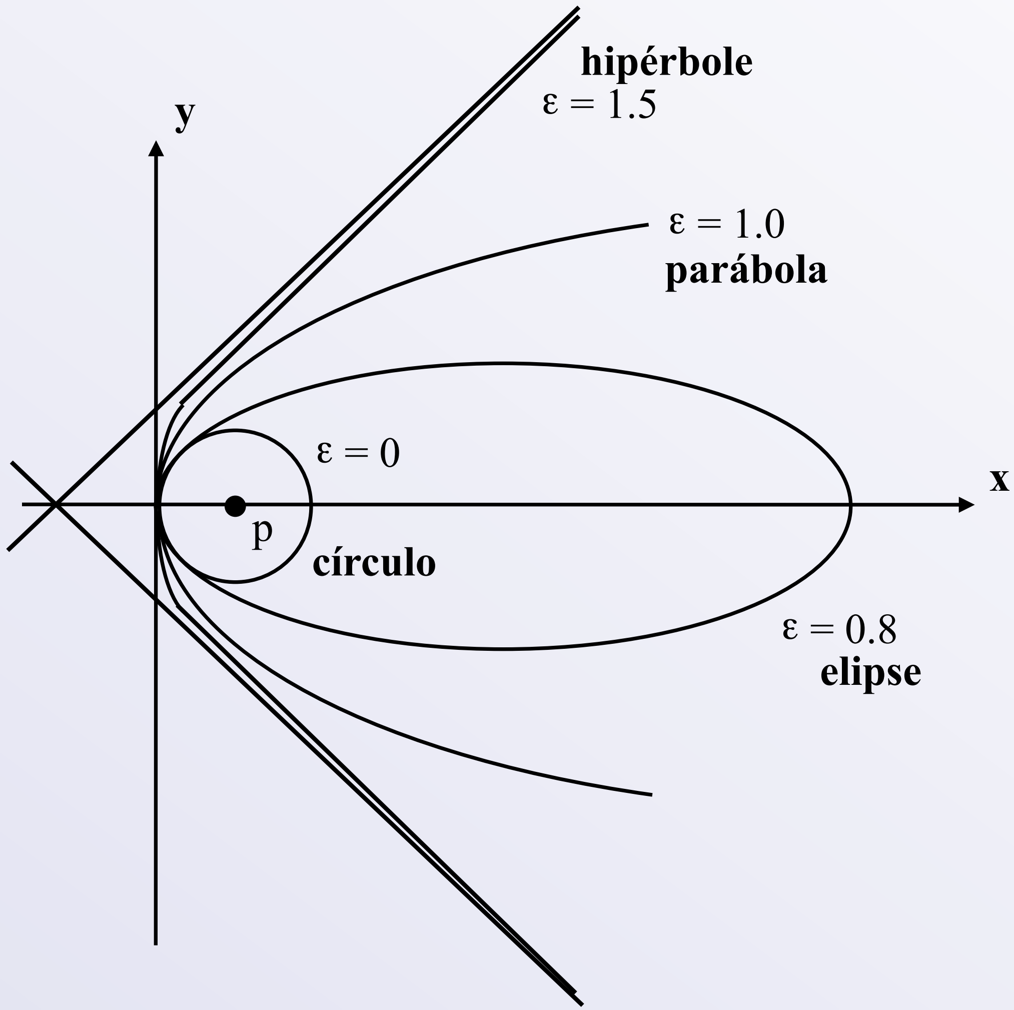
Polinômios de 2º. grau

- A cônica gerada depende do valor do discriminante, $AC - B^2$.
  - Se $AC - B^2 > 0$, geramos uma elipse.
  - Se $AC - B^2 = 0$, geramos uma parábola.
  - Se $AC - B^2 < 0$, geramos uma hipérbole.
- Exemplos de funções implícitas:
  - Elipse: $x^2 + xy + y^2 - 1$: ($AC\text{-}B^2 = 0.75$)
  - Parábola: $x^2 + 2xy + y^2 + 3x - 6y - 7$: ($AC\text{-}B^2 = 0$)
  - Hipérbole: $x^2 + 4xy + 2y^2 -4x +y -3$: ($AC\text{-}B^2 = -2$)

Polinômios de 2º. grau

- Um caso especial da forma quadrática, é a equação do vértice comum:
  - $y^2 = 2px - (1-\boldsymbol{\varepsilon}^2)x^2$, mostra como as diferentes cônicas são relacionadas.
  - Esta curva passa pelo ponto (0,0) e tem tamanho proporcional a constante *p*. A cônica que ela descreve depende do valor da **excentricidade ε**.
  - A excentricidade mede o desvio da curva em relação à um círculo perfeito (excentricidade = 0).

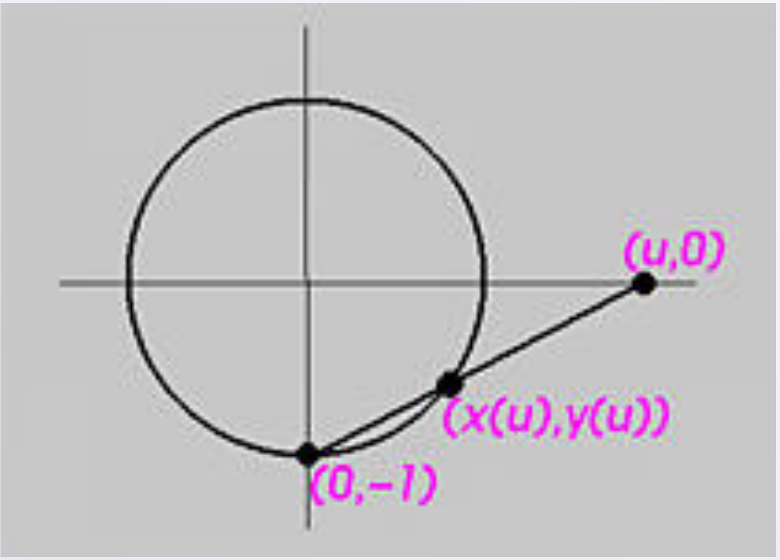
Polinômios de 2º. grau



Círculo

- Equação do círculo
  - $x^2 + y^2 = 1$
- Equação da reta entre (0,-1) e (u,0)
  - $x = uy + u$
  - Substituindo na equação do círculo, teremos duas raízes: -1 e y = $(1 - u^2) / (1 + u^2)$

$$x = \frac{2u}{1+u^2}$$
$$y = \frac{1-u^2}{1+u^2}$$



Curvas polinomiais de 3º. grau

- Curvas polinomiais de 3º. grau e maior não são de fácil conversão para a forma paramétrica.
- Polinômios cúbicos, no entanto, são úteis no desenho de curvas e superfícies, mas utilizaremos uma coleção de pontos de controle e um algoritmo para gerar pontos na curva.
- O designer poderá editar a posição dos pontos de controle e visualizar a nova curva.
- Esta é uma abordagem visual, deixando o usuário acompanhar interativamente o progresso do design da curva.

Formas Paramétricas Racionais

- x e y são definidos como a razão entre dois polinômios (quadráticos neste exemplo).

$$P(t) = \frac{P_0(1-t)^2 + 2wP_1t(1-t) + P_2t^2}{(1-t)^2 + 2wt(1-t) + t^2}$$

- P_0, P_1, e P_2 são quaisquer 3 pontos em um plano, chamados de pontos de controle.
- A variável w representa o parâmetro peso.

Formas Paramétricas Racionais

- A equação P(t) constiui-se na verdade de 2 equações: uma para cada x(t) e y(t):

$$x(t) = \frac{x_0(1-t)^2 + 2wx_1t(1-t) + x_2t^2}{(1-t)^2 + 2wt(1-t) + t^2}$$

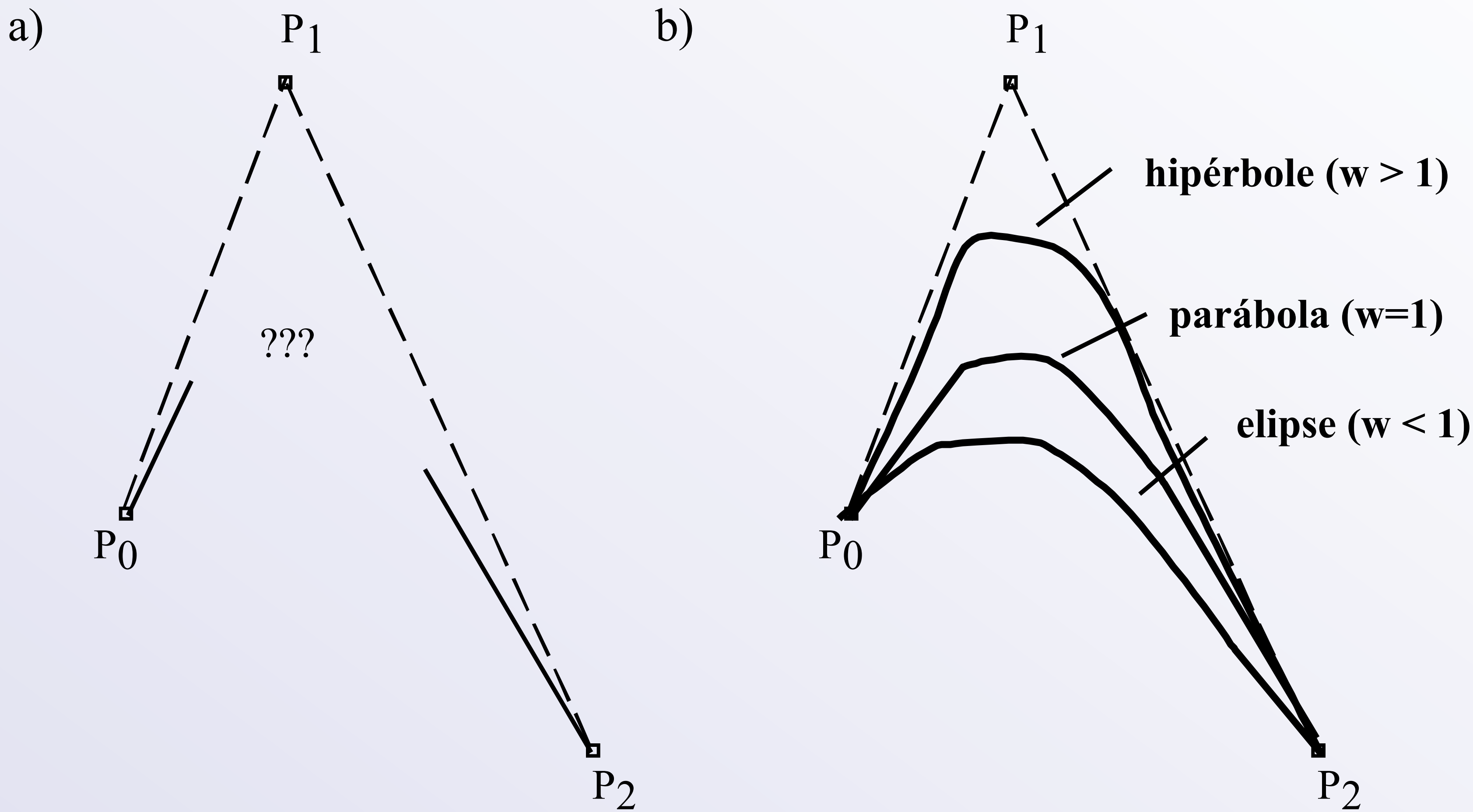
$$y(t) = \frac{y_0(1-t)^2 + 2wy_1t(1-t) + y_2t^2}{(1-t)^2 + 2wt(1-t) + t^2}$$

- *P*(*t*) é uma combinação linear dos pontos de controle.

Formas Paramétricas Racionais

- Quando t=0, a função resume-se simplesmente em (x_0, y_0); então esta curva passa, ou **interpola**, o ponto P_0.
- Quando t=1, ela passa por P_2. Para t entre 0 e 1, $P(t)$ dependerá de todos os três pontos de uma forma mais complicada.
- A figura no próximo slide mostra as curvas resultantes da variação do parâmetro w.

Formas Paramétricas Racionais



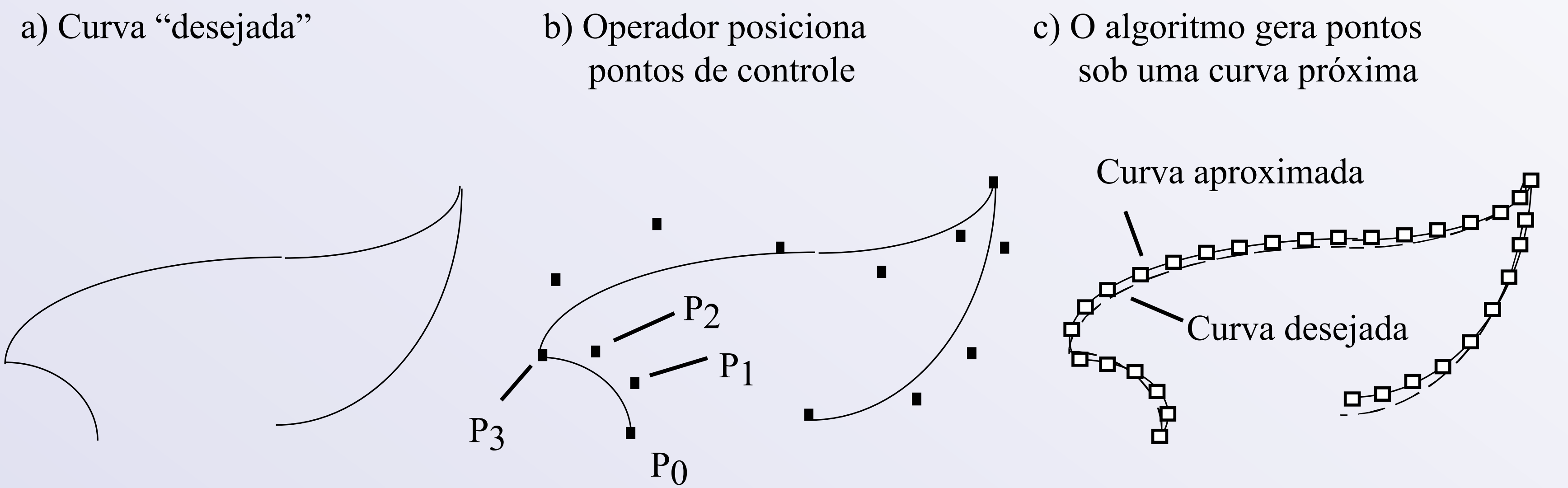
As formas paramétricas racionais permitem gerar
cônicas exatas, sem a utilização de termos trigonométricos

Design Interativo de Curvas

- Permitir que o designer especifique um pequeno número de pontos de controle e desenhe uma ampla variedade de formas.
- O objetivo é representar esta curva de forma que ela possa ser reproduzida facilmente, incluindo variações na sua forma e tamanho;
- Assim, estas curvas poderão ser enviadas para máquinas a fim de executar cortes, criar moldes, etc.
- Não existe uma fórmula simples para criar este tipo de representação.

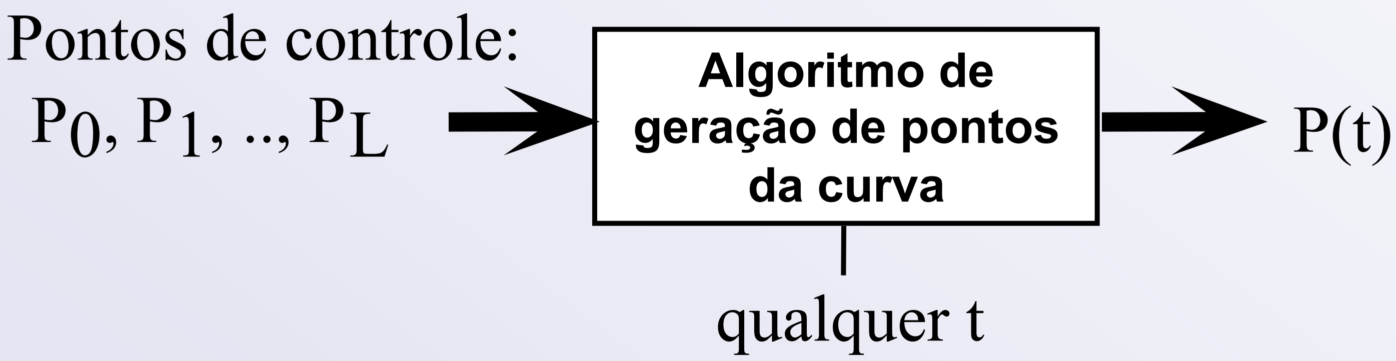
Design Interativo de Curvas (ii)

- Para desenhar, o designer move o ponteiro ao longo de uma curva ideal, clicando em um conjunto de **pontos de controle** P_0, P_1,... próximas à curva ideal.



Design Interativo de Curvas (iii)

- The papel do algoritmo é produzir um ponto *P*(*t*) para qualquer valor de *t*. Os dados de entrada são o conjunto de pontos de controle, que determinam a forma da curva *P*(*t*).



Design Interativo de Curvas (iv)

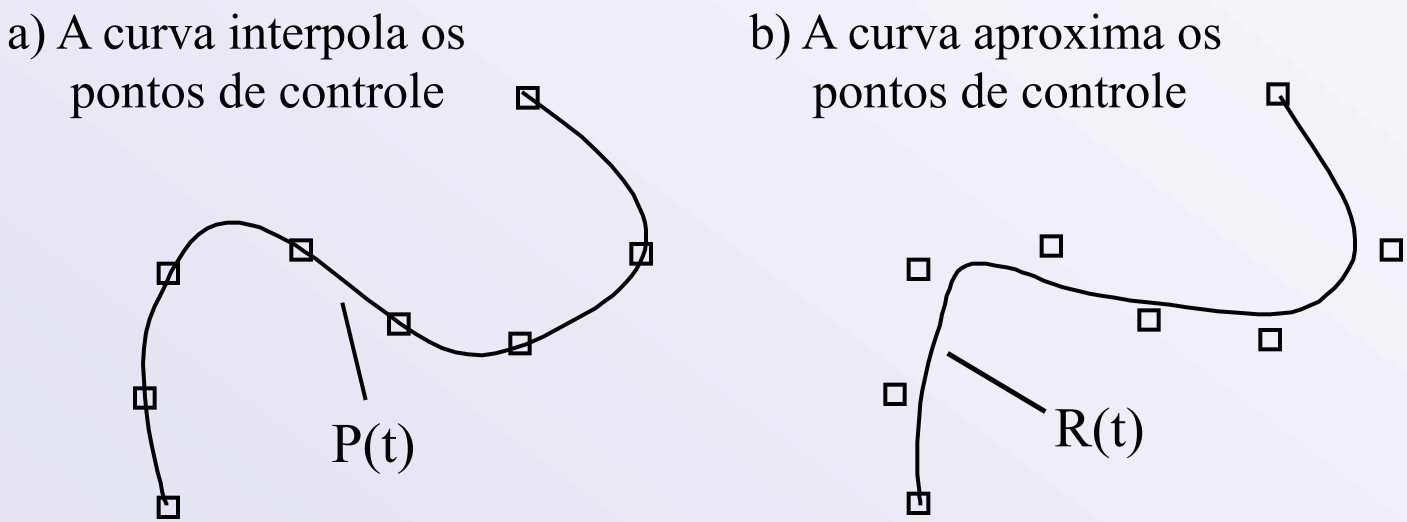
- Normalmente implementado como uma função
  - ```
    Point2D PontoCurva(double t,
                       Point2D *pts_controle);
    ```
- Que retorna um ponto para qualquer valor de *t* dentro de um certo intervalo.
- Para desenhar a curva, o usuário escolhe uma seqüência de valores de *t*, e chama a função PontoCurva para cada um deles;
- Finalmente, conecta-se os pontos gerados através de segmentos de reta (polilinha).
    ```

Design Interativo de Curvas (v)

- Passos no design interativo de curvas:
  - 1. Seleciona os pontos de controle iniciais;
  - 2. Usa algoritmo para gerar curva;
  - 3. Se a curva resultante for satisfatória, pare.
  - 4. Caso contrário, mova alguns pontos de controle;
  - 5. Volte ao passo 2;

Interpolação e Aproximação

- *Interpolação*: a curva passa <u>nos</u> pontos de controle.
- *Aproximação*: a curva passa <u>próxima</u> aos pontos de controle.



Curvas de Bézier

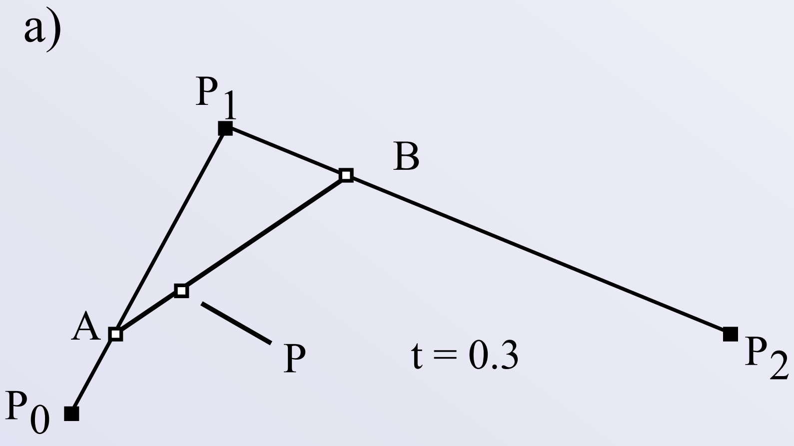
- As curvas de Bézier (curvas aproximativas) foram inventadas para auxiliar no design de automóveis. O algoritmo “de Casteljau” é usado para desenhá-las.
- O algoritmo de De Casteljau baseia-se em uma seqüência de passos de transformação geométrica de fácil implementação.
- Através desta transformação, é possível deduzir uma série de propriedades das que curvas que ela gera.

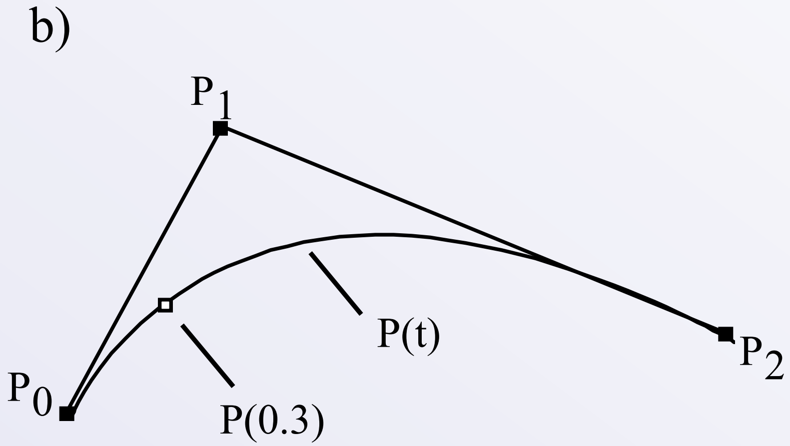
Curvas de Bézier (ii)

- Transformação de 3 pontos para obtenção de uma parábola:
  - Escolha três pontos: P_0, P_1, and P_2.
  - Escolha um valor de t entre 0 and 1, ex. $t = 0.3$.
  - Localize o ponto A que está a uma fração t ao longo da linha de P_0 to P_1. Analogamente, localize B a mesma fração t entre os pontos P_1 e P_2.
  - Os novos pontos serão:
    - $A(t) = (1-t)P_0 + tP_1$, $B(t) = (1-t)P_1 + tP_2$

Curvas de Bézier (iii)

- Agora repita a interpolação linear nestes pontos (usando o mesmo valor de *t*): Ache o ponto, *P*(*t*), que está na fração *t* do caminho entre A e B: P(t) = (1-t)A + tB.





Curvas de Bézier (iv)

- Se este processo for repetido para *todo t* entre 0 e 1, a curva *P*(*t*) será gerada.
- A forma paramétrica resultante para tal curva será $P(t) = (1-t)^2P_0 + 2t(1-t)P_1 + t^2P_2$

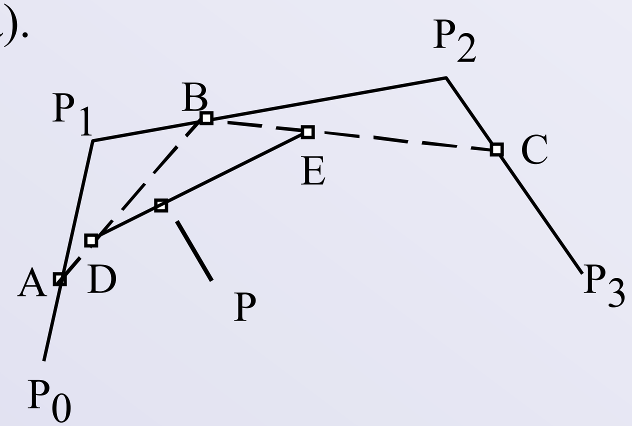
Curvas de Bézier (v)

- A forma paramétrica para $P(t)$ é quadrática em t, então concluimos que tal curva é uma parábola.
- Ela continuará sendo uma parábola mesmo quando t variar entre $-\infty$ to ∞.
- Ela passará em P_0 quando $t = 0$ e em P_2 quando $t = 1$ (porquê ?)
- Assim obtemos um processo bem-definido que gera uma curva parabólica suave baseada em três pontos de controle.

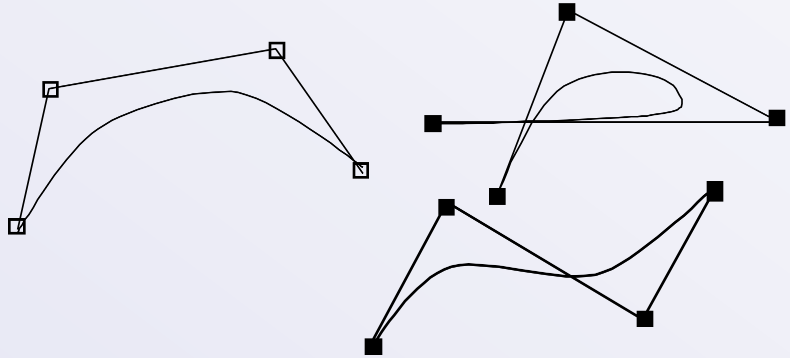
Curvas de Bézier (vi)

- A forma mais comum das curvas Bezier utilizam 4 pontos de controle.
- Para um dado valor de *t,* o ponto *A* é posicionado a uma fração *t* do caminho entre P_0 e P_1, e similarmente para *B* e *C*.
- Então *D* é colocado a uma fração *t* do caminho entre *A* to *B*, e similarmente para o ponto *E*.
- Finalmente, o ponto desejado *P* está localizado a uma fração *t* do caminho entre *D* e *E*.

a).



b).



Curvas de Bézier (vii)

- Se este processo for efetuado para todo t entre 0 e 1, a curva P(t) começará em P_0, será atraída na direção de P_1 e P_2, e terminará em P_3.
- Esta será a curva Bézier resultante de quatro pontos de controle.

Curvas de Bézier (viii)

- A curva Bézier baseada em quatro pontos de controle possui a forma paramétrica $P(t) = P_0(1-t)^3 + P_1 3(1-t)^2 t + P_2 3(1-t)t^2 + P_3 t^3$.
- Cada ponto de controle P_i é pesado por um polinômio cúbico, e os termos são somados.
- Estes termos são chamados de **polinômios de Bernstein.**

Polinômios de Bernstein

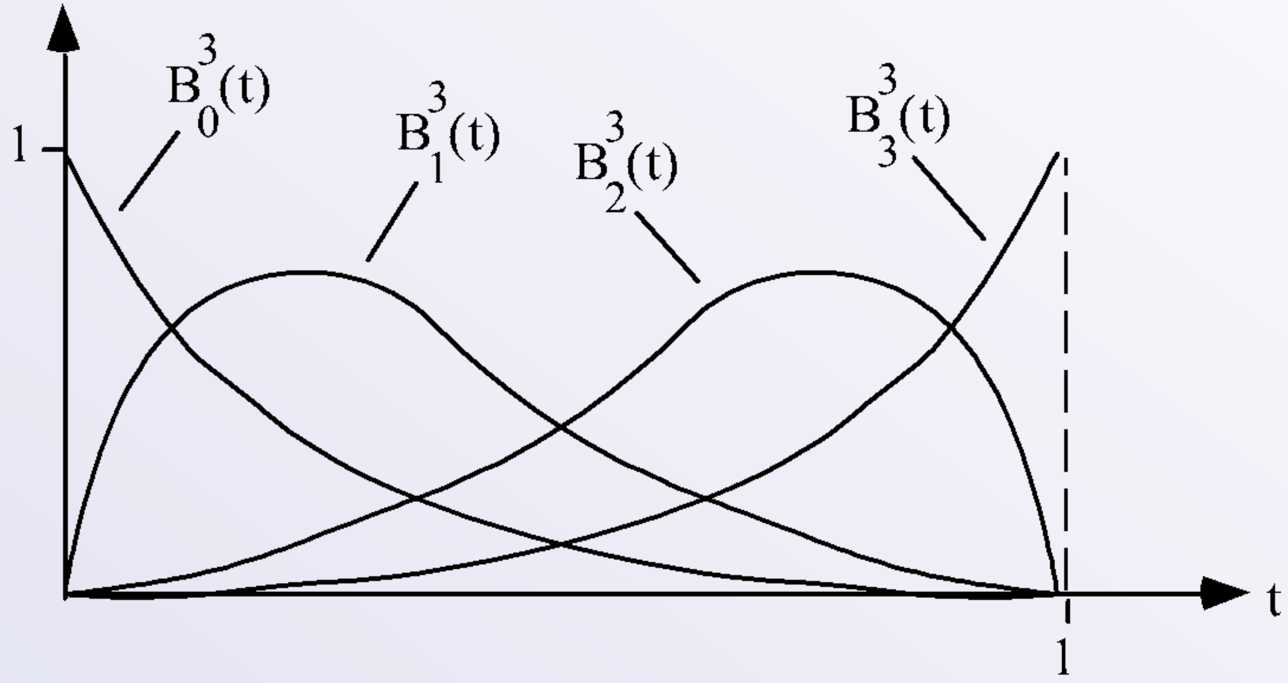
- Os polinômios de Bernstein são:

$$B_0^3 = (1-t)^3 \quad B_1^3 = 3(1-t)^2 t \quad B_2^3 = 3(1-t)t^2 \quad B_3^3 = t^3$$

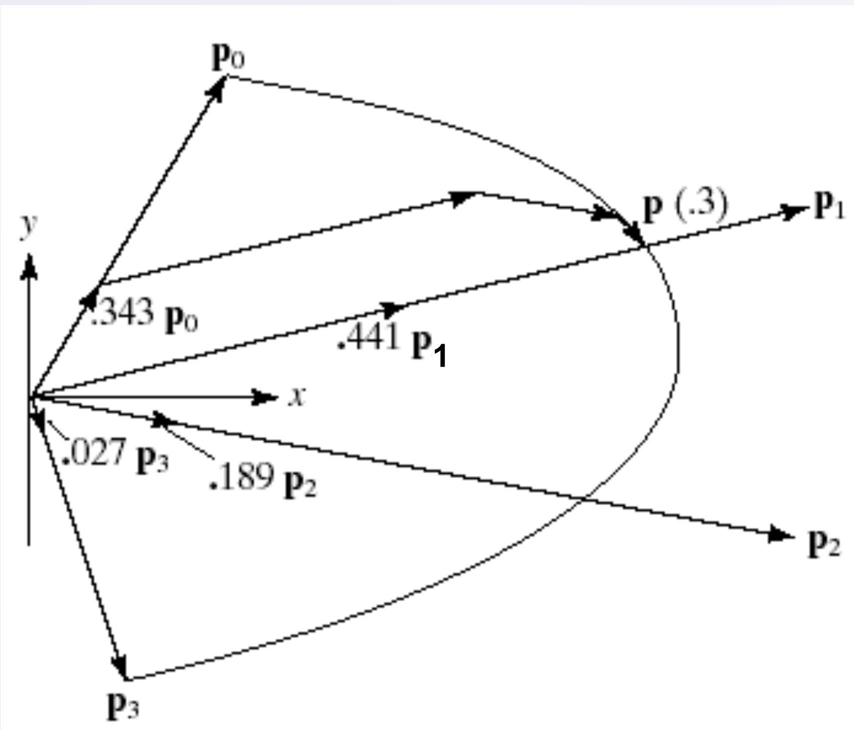
- A soma destes polinômios totaliza 1 para qualquer valor de t, e representa o resultado da expansão de $(1 – t + t)^3$:

$$((1-t)+t)^3 = (1-t)^3 + 3(1-t)^2 t + 3(1-t)t^2 + t^3$$

Polinômios de Bernstein (2)



Ponderação de pontos com polinômios de Bernstein



- Considere pontos como vetores na origem (ex., $P_0 = \mathbf{p_o}$, etc.) e t = 0.3. Então $\mathbf{p}(0.3) = 0.343\ \mathbf{p_0} + 0.441\ \mathbf{p_1} + 0.189\ \mathbf{p_2} + 0.027\ \mathbf{p_3}$.
- Nesta figura os quatro vetores são modulados e os resultados são adicionados para formar o vetor **p**(0.3).

Generalização das Curvas de Bézier

- A curva resultante será:

$$B_k^L(t) = \binom{L}{k}(1-t)^{L-k}\, t^k \qquad P(t) = \sum_{k=0}^{L} \mathrm{P}_k\, B_k^L(t)$$

- onde o coeficiente binomial é:

$$\binom{L}{k} = \frac{L!}{k!\,(L-k)!}$$

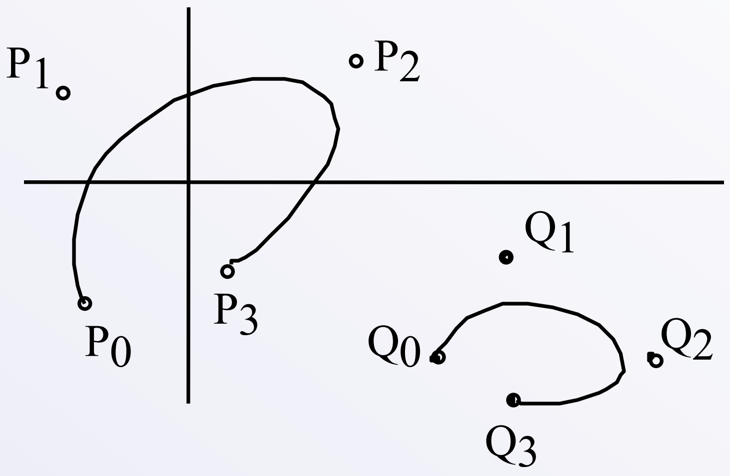
Propriedades das Curvas Bézier

- As propriedades das curvas de Bézier fazem com que elas sejam perfeitas para o uso em CAD.
  - **Interpolação dos pontos finais:** A curva Bézier $P(t)$ baseada nos pontos de controle P_0, P_1, . . ., P_L sempre interpola os pontos P_0 e P_L.
  - **Invariância afim:** para aplicar uma transformação afim T em todos os pontos $P(t)$ da curva, transformamos os pontos de controle uma vez, e usamos os novos pontos para recriar a curva transformada Q(t) em qualquer *t.*

$$Q(t) = \sum_{k=0}^{L} T(P_k) B_k^L(t) = T\left(\sum_{k=0}^{L} P_k B_k^L(t)\right)$$

Propriedades das curvas Bézier

- Exemplo: Uma curva Bezier é baseada em quatro pontos de controle P_0,..., P_3. Os pontos são rotacionados, escalados e transladados para as novas posições $Q_{k.}$
- A curva Bezier resultante para Q_k é desenhada. Ela é idêntica ao resultado da transformação da curva Bezier original.



Propriedades das curvas Bézier

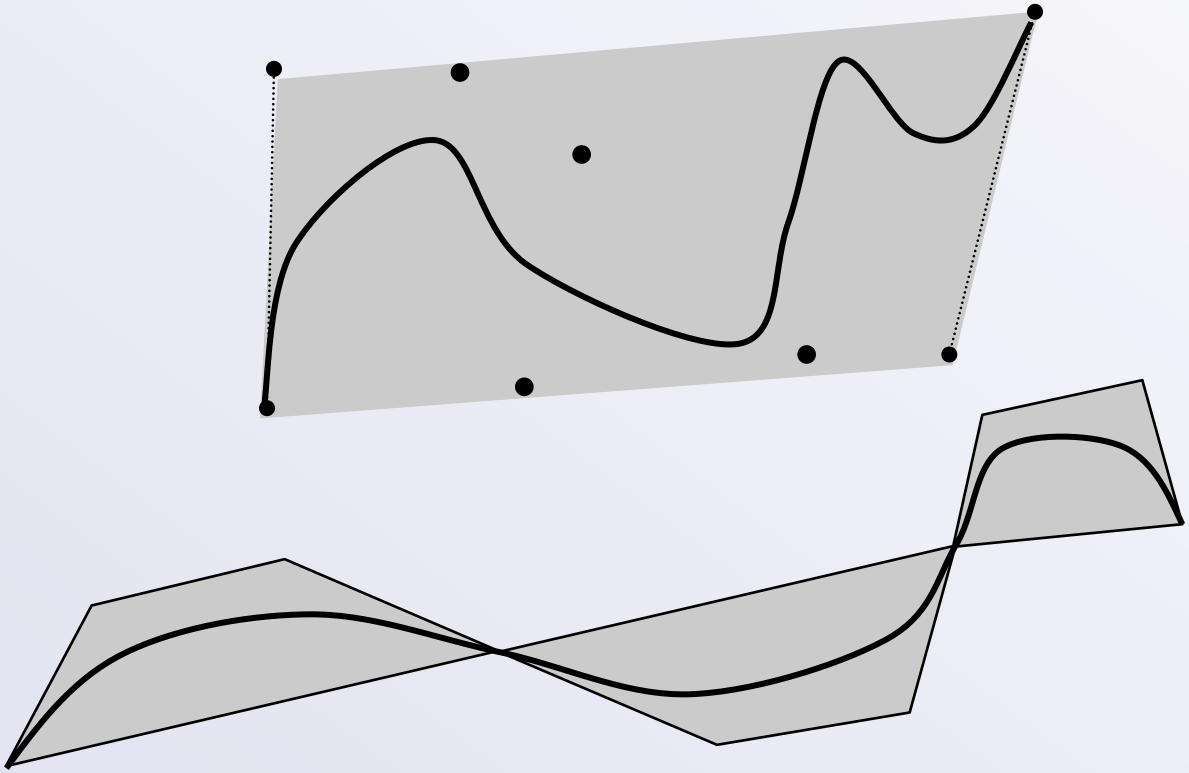
- Uma curva Bezier $P(t)$, nunca deixa o seu **fecho convexo**.
- O fecho convexo do conjunto de pontos V_0, V_1,.. ., V_i é o conjunto de todas as suas *combinações convexas*; isto é, o conjunto de todos os pontos dados por

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

onde cada α_i é positivo, e a soma é igual 1.

Fecho Convexo

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



Propriedades das curvas Bezier

- A primeira derivada de uma curva Bézier é definida como:

$$P^{'}(t) = L\sum_{k=0}^{L-1} \Delta P_k B_k^{L-1}(t)$$

onde $\Delta P_k = P_{k+1} - P_k$

- A velocidade é outra curva Bézier, construída do novo conjunto de vetores de controle ΔP_k .
- A derivada diminui a ordem da curva em 1: a derivada de uma Bézier cúbica é uma curva Bézier quadrática.

Desenhando Curvas Bézier

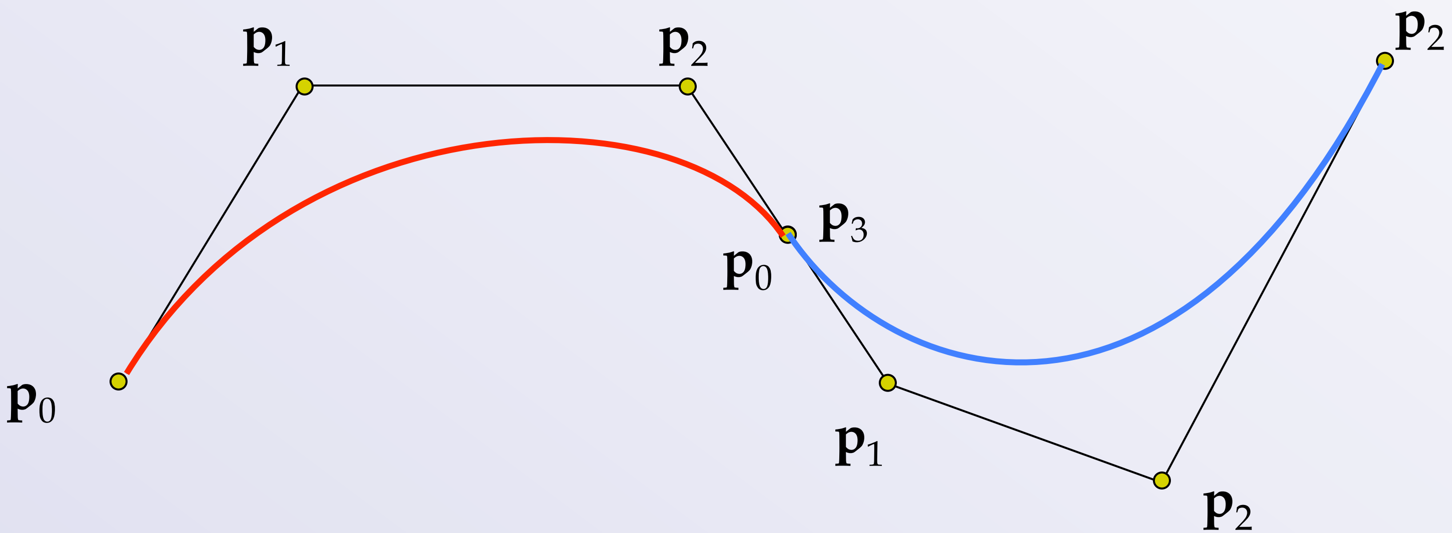
- Curva normalmente é aproximada por uma linha poligonal
- Pontos podem ser obtidos avaliando a curva em t = t_1, t_2 ... t_k
  - Avaliar os polinômios de Bernstein
  - Usar o algoritmo recursivo de De Casteljau
- Quantos pontos?
  - Mais pontos em regiões de alta curvatura
- Idéia: subdividir recursivamente a curva em trechos até que cada trecho seja aproximadamente “reto”

Curvas Longas

- 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



Tarefa de casa

- Como desenhar um círculo com curvas Bézier ?