Corso di *Tecniche Avanzate per la Grafica*

Rappresentazione di Oggetti Tridimensionali

Docente:

Massimiliano Corsini

Laurea Specialistica in Informatica

Facoltà di Scienze MM. FF. NN.
Università degli Studi di Ferrrara

Overview

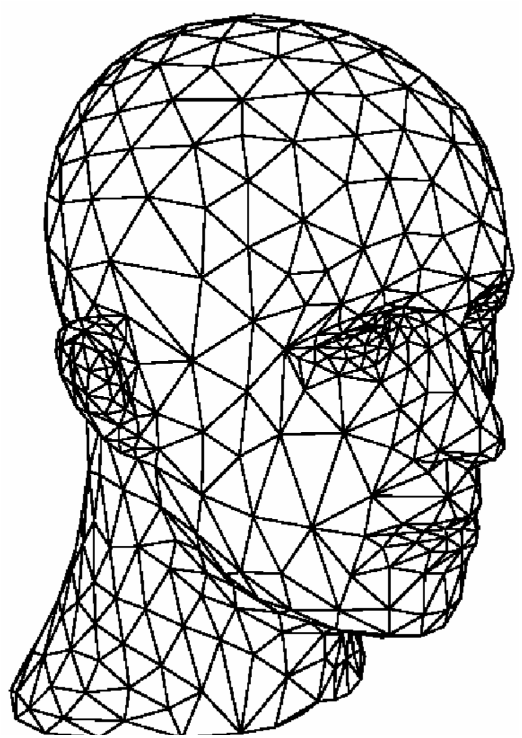
- Rappresentazione Oggetti 3D
  - Rappresentazione Poligonale (meshes)
  - Rappresentazione Implicita
  - Rappresentazione Parametrica
    - Polinomi di Bernstein
    - Curve e Superfici di Bèzier
    - Curve e Superfici B-Spline
    - NURBS
  - Voxels (cenni)
  - Constructive Solid Geometry (CSG) (cenni)
  - Superfici di Suddivisione
  - Approfondimenti sulle mesh poligonali

Modellazione Geometrica

- La rappresentazione di modelli 3D si può suddividere in due categorie:
  - Boundary-based: è la superfice dell'oggetto 3D ad essere rappresentata (*boundary representation* or *b-rep*)
    - Esempi: mesh poligonali, rapp. Implicite e parametriche
  - Volume-based: è il volume ad essere rappresentato
    - Esempi: voxels, CSG.
- Modellazione Geometrica può essere:
  - Automatica (a partire da immagini → Computer Vision)
  - Manuale (Maya©, Rhino3D©, 3DStudio Max©)
  - Procedurale (frattali, grammatiche)
- Una volta che la geometria di un oggetto è rappresentata questa può essere *renderizzata* per produrre l'immagine finale.

Rappresentazione Poligonale (Mesh)

La superfice dell'oggetto 3D è rappresentata da un insieme di poligoni nello spazio.



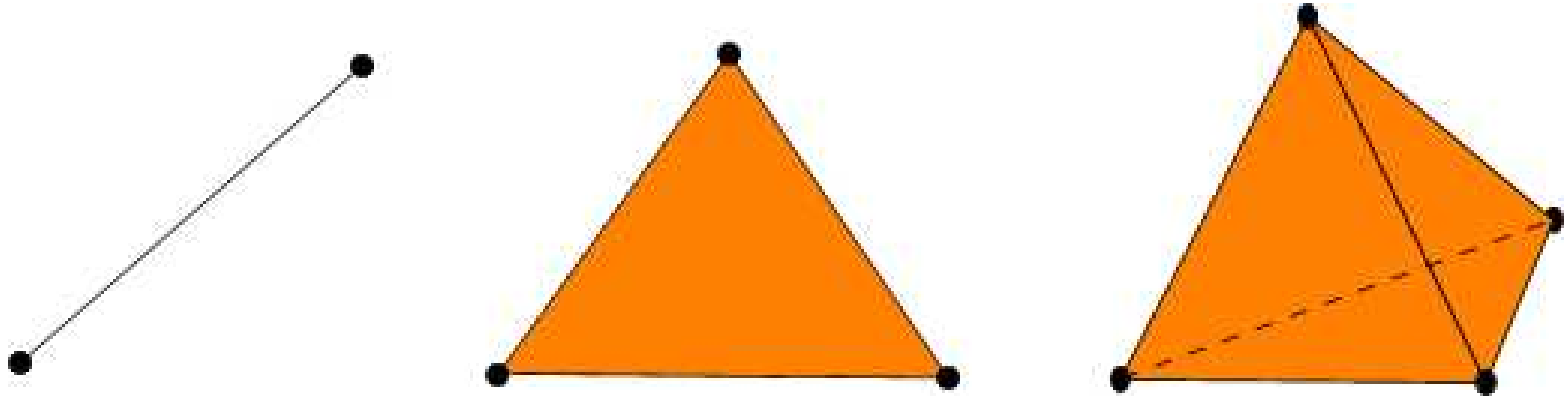
Mesh Poligonale

Intuitivamente, una mesh è un insieme di vertici, lati e facce.

Matematicamente, una mesh è una tupla (K,V) dove V è l'insieme dei punti nello spazio (vertici), e K, insieme dei *complessi simpliciali*, in pratica contiene le informazioni sulla connettività (topologia) tra I punti contenuti in V (ossia lati e facce).

Complessi Simpliciali

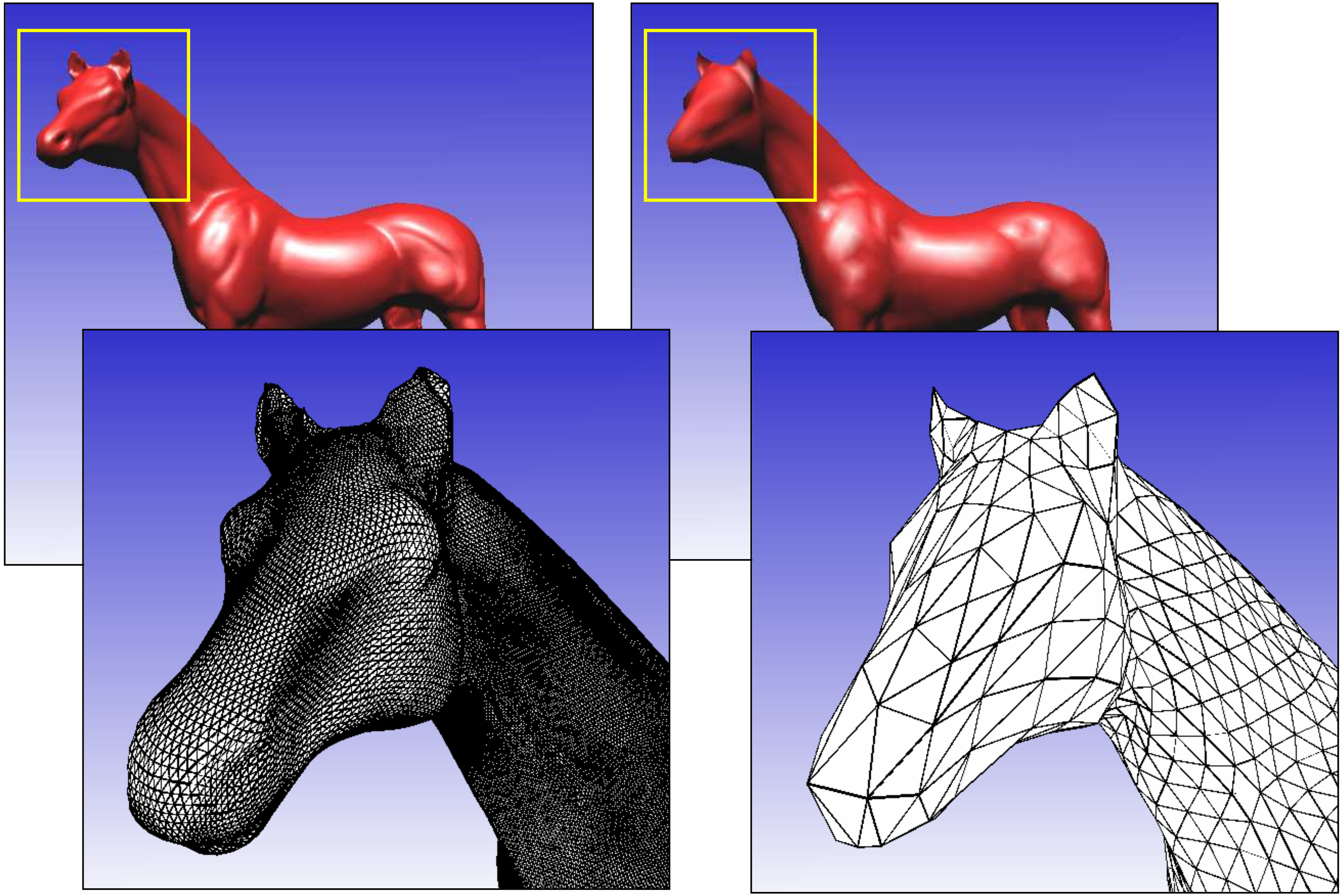
- Un *simplesso* di ordine *k* è la combinazione convessa dei k+1 punti che lo compongono
  - Simplesso di ordine 1 → segmento
  - Simplesso di ordine 2 → triangolo
  - Simplesso di ordine 3 → tetraedro
- L'insieme dei simplessi (che rispettano certe proprietà) forma il complesso simpliciale *K* visto precedentemente.



Meshes

- Le mesh poligonali sono un'approssimazione discreta di una superficie.
- Può essere vista come il minimo comune denominatore di tutte le altre rappresentazioni.
- Tipicamente vengono utilizzate particolari tipi di mesh:
  - *Mesh triangolari:* ossia mesh le cui facce sono triangoli (moderne schede grafiche).
  - *Mesh quadrangolari:* ossia mesh le cui facce sono quadrilateri (es: Geographic Information System (GIS)).

Esempi di Meshes



Equazione di Eulero

$$\boxed{\#V - \#E + \#F = 2 - G}$$

#V = numero di vertici
#E = numero di lati
#F = numero di facce
G = genus (sfera ha G = 0 , toro ha G = 1)

Questa relazione è valida solamente per mesh poligonali *chiuse*.

Limitazioni delle Meshes

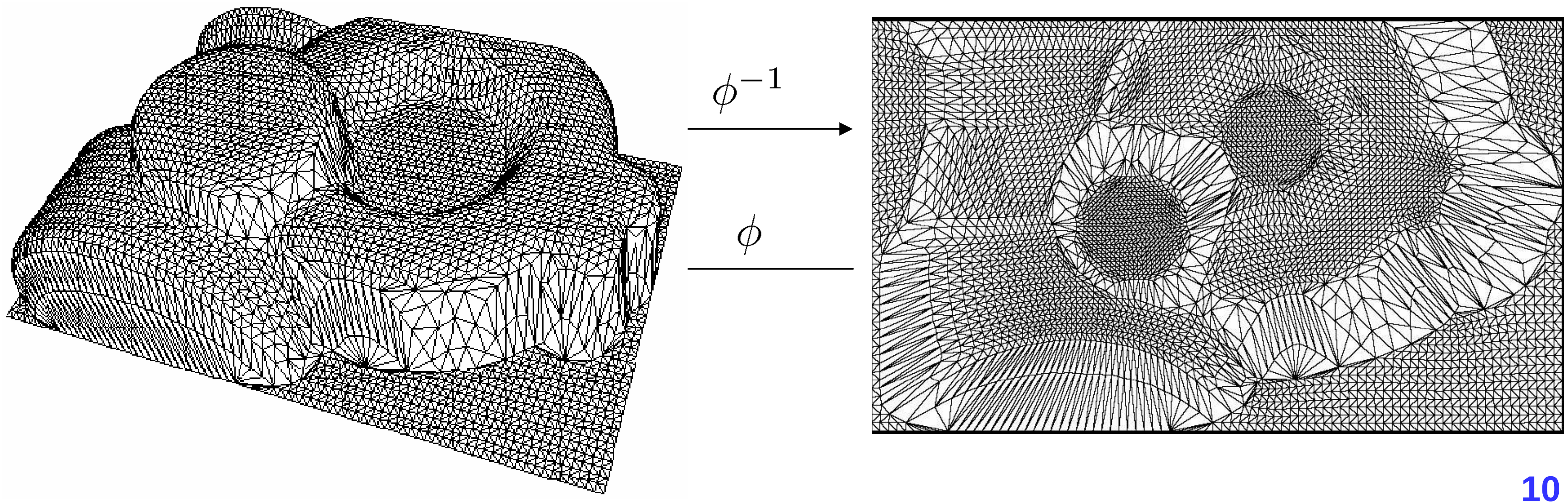
- Facce Planari
- Non sono una rappresentazione compatta – i.e. modelli con elevatissimo dettaglio richiedono grandi quantità di dati
- Editing diretto è “difficile”
- L’elaborazione con algoritmi geometrici è intrinsicamente onerosa
- Non hanno una parametrizzazione naturale, la parametrizzazione di mesh è un settore ancora attivo di ricerca

Parametrizzazione

La parametrizzazione di una mesh consiste nel trovare una funzione capace di mappare i vertici della mesh in un dominio planare.

$$\phi : T \subset R^2 \mapsto S$$

La parametrizzazione introduce inevitabilmente distorsioni. Le tecniche odierne cercano di preservare una o più proprietà della superifice come angoli o aree.



Rappresentazione Implicita

La superficie viene definita in termine delle sue coordinate cartesiane:

$$f\,(x, y, z) = 0$$

- Esempi :
  - Sfera di raggio r : $x^2 + y^2 + z^2 - r^2 = 0$
  - Piano generico : $ax + by + cz + d = 0$
- Rappresentazione estremamente compatta (!!)
- Pochi utilizzi (design meccanico)

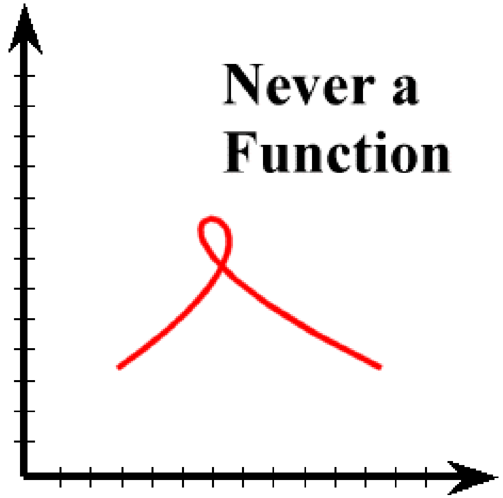
Rappresentazione Parametrica

La superficie viene definita in forma parametrica utilizzando tre funzioni bi-variate:

$$S(u,v) = (X(u,v), Y(u,v), Z(u,v))$$

dove u *e* v *variano tra* 0 *ed* 1

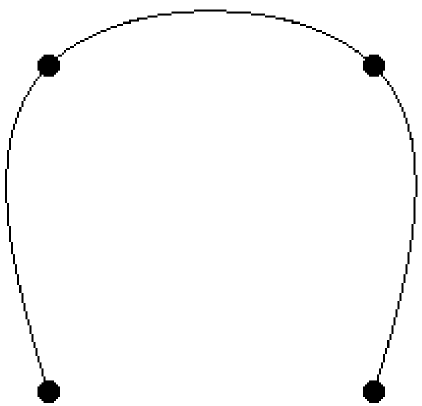
- Le curve parametriche sono molto flessibili
- Non sono vincolate ad essere funzioni



Curve Parametriche

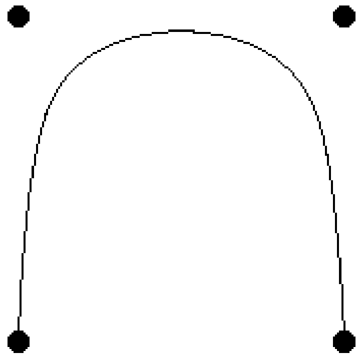
$$P(t) = \sum_{i=0}^{n} B_i(t) P_i$$

Funzione di Blending



Interpolation

La curva deve necessariamente passare sui punti di controllo



Approssimazione

La forma della curva è influenzata dai punti di controllo

Un esempio di Blending Function

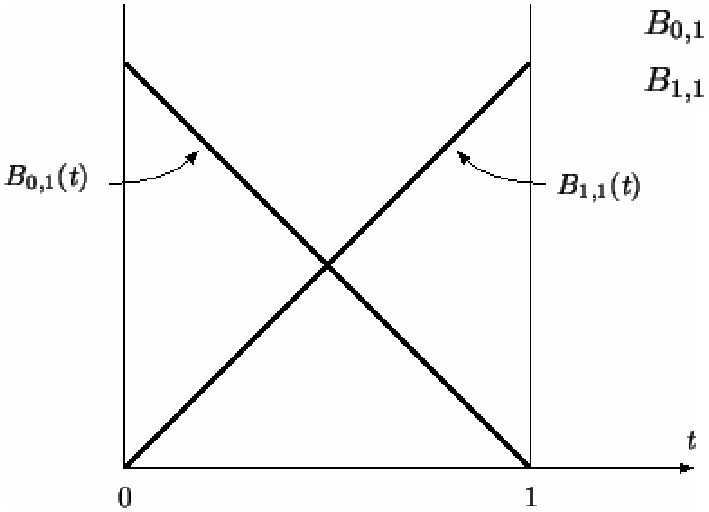
Polinomio di Bernstein di grado *n*

$$B_{i,n}(t)=\binom{n}{i}t^i(1-t)^{n-i} \qquad i=0\ldots n$$

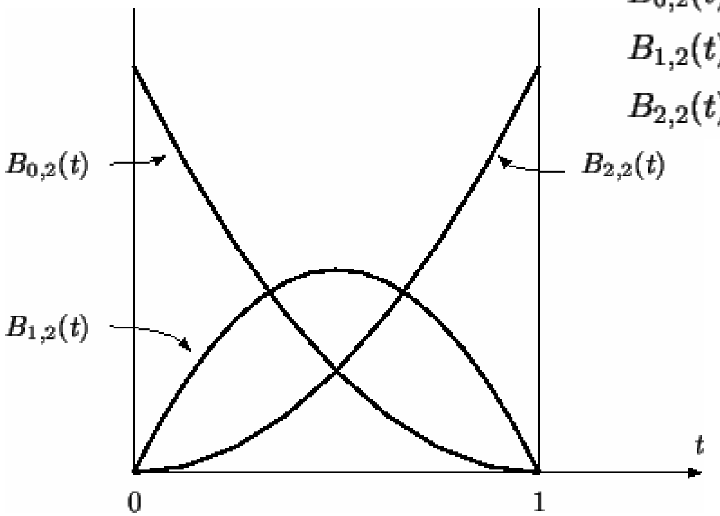
$$\binom{n}{i}=\frac{n!}{i!(n-i)!}$$

- Polinomi sono "semplici" :-)
- Polinomi formano uno spazio vettoriale, $\{1,t,t^2,\ldots,t^n\}$ è la base di questo spazio
- $\{B_{0,n}, B_{1,n}, \ldots, B_{n,n}\}$ è chiamata *base di Bernstein*
- $B_{i,n-1}$ è combinazione lineare di $B_{i,n}$
- Partizione dell'unità ("la loro somma dà uno")

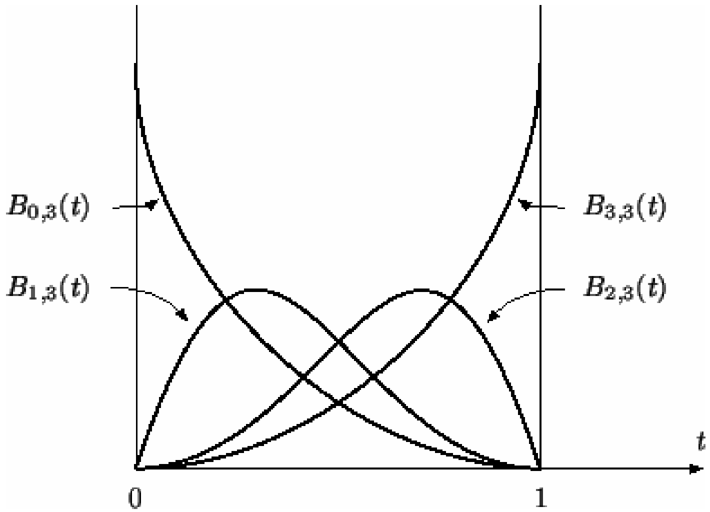
Polinomi di Bernstein



$$B_{0,1}(t) = 1 - t$$
$$B_{1,1}(t) = t$$

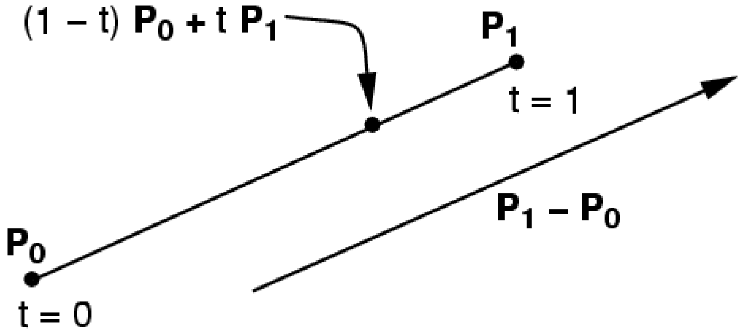


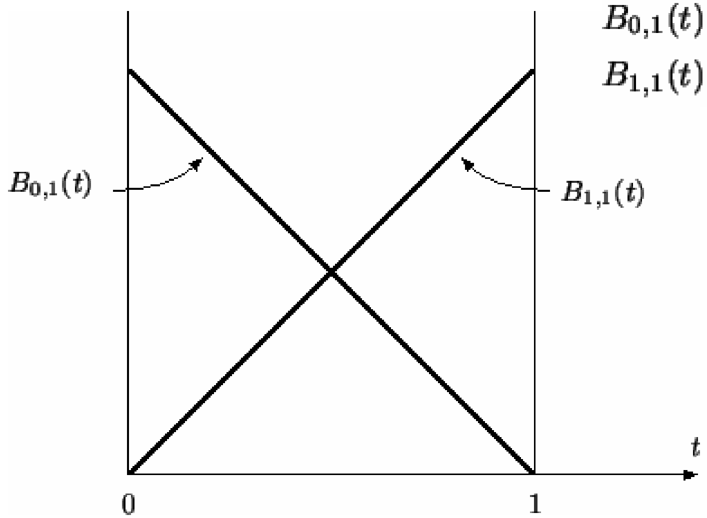
$$B_{0,2}(t) = (1-t)^2$$
$$B_{1,2}(t) = 2t(1-t)$$
$$B_{2,2}(t) = t^2$$



$$B_{0,3}(t) = (1-t)^3$$
$$B_{1,3}(t) = 3t(1-t)^2$$
$$B_{2,3}(t) = 3t^2(1-t)$$
$$B_{3,3}(t) = t^3$$

Interpolazione Lineare





$B_{0,1}(t) = 1 - t$

$B_{1,1}(t) = t$

$$P(t) = (1-t)P_0 + tP_1 = \begin{bmatrix}(1-t) & t\end{bmatrix}\begin{bmatrix}P_0 \\ P_1\end{bmatrix} = \begin{bmatrix}1 & t\end{bmatrix}\begin{bmatrix}1 & 0 \\ -1 & 1\end{bmatrix}\begin{bmatrix}P_0 \\ P_1\end{bmatrix}$$

$$B_1(t) = \begin{bmatrix}B_{0,1}(t) & B_{1,1}(t)\end{bmatrix} = \begin{bmatrix}(1-t) & t\end{bmatrix} = \begin{bmatrix}1 & t\end{bmatrix}\begin{bmatrix}1 & 0 \\ -1 & 1\end{bmatrix}$$

Rappresentazione Matriciale dei Polinomi di Bernstein

$$\left[\begin{array}{ccccc} B_{0,n}(t) & B_{1,n}(t) & B_{2,n}(t) & \dots & B_{n,n}(t) \end{array}\right] = \left[\begin{array}{ccccc} 1 & t & t^2 & \dots & t^n \end{array}\right] \left[\begin{array}{ccccc} b_{0,0} & 0 & 0 & \dots & 0 \\ b_{1,0} & b_{1,1} & 0 & \dots & 0 \\ b_{2,0} & b_{2,1} & b_{2,2} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{n,0} & b_{n,1} & b_{n,2} & \dots & b_{n,n} \end{array}\right]$$

Caso quadratico (n=2)

$$B_2(t) = \left[\begin{array}{ccc} 1 & t & t^2 \end{array}\right] \left[\begin{array}{ccc} 1 & 0 & 0 \\ -2 & 2 & 0 \\ 1 & -2 & 1 \end{array}\right]$$

Caso cubico (n=3)

$$B_3(t) = \left[\begin{array}{cccc} 1 & t & t^2 & t^3 \end{array}\right] \left[\begin{array}{cccc} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ 1 & 3 & -3 & 1 \end{array}\right]$$

Curve di Bézier

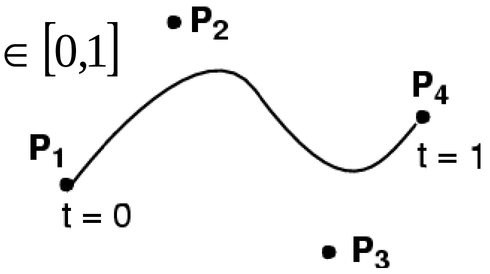
Curve di Bezier di grado *n*

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

$B_{i,n}(t)$ sono polinomi di Bernstein di grado n definiti per $0 \leq t \leq 1$

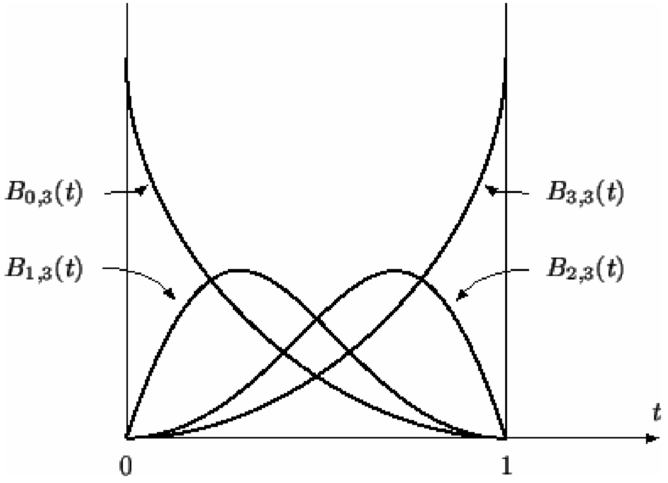
<u>Definizione analitica:</u> $P(t)=\sum_{i=0}^{3} P_i B_{i,3}(t)$

$t \in [0,1]$



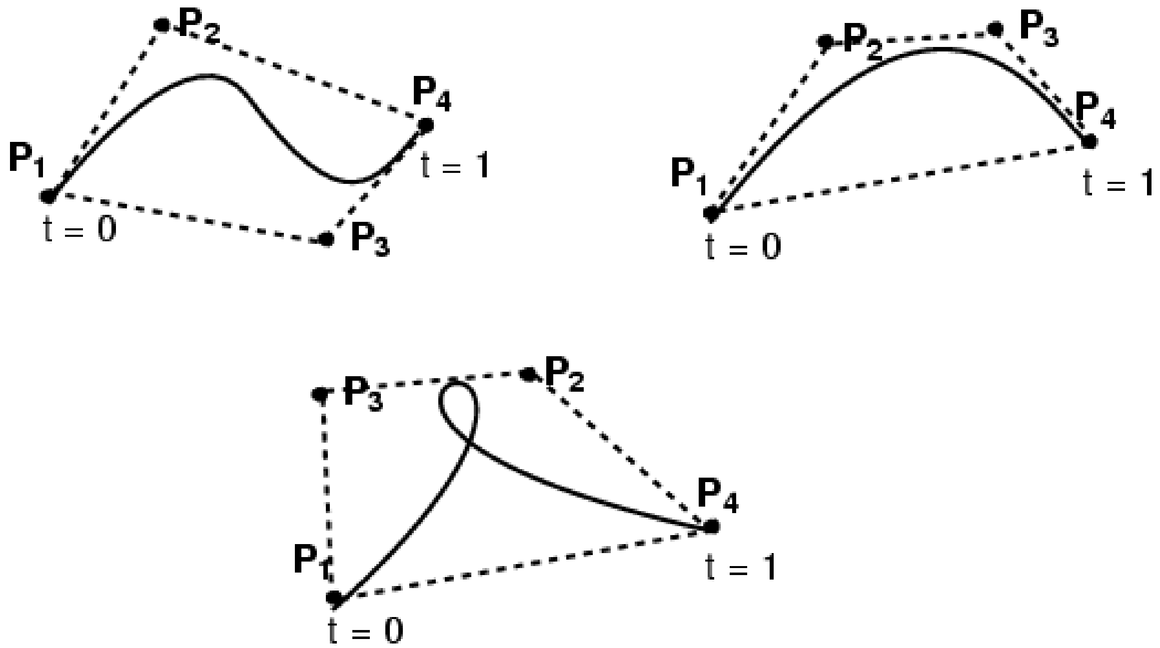
<u>Notazione Matriciale:</u> $\boxed{P(t)} = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t)P_2 + t^3 P_3 =$

$$= \begin{bmatrix}(1-t)^3 & t(1-t)^2 & t^2(1-t) & t^3\end{bmatrix}\begin{bmatrix}P_0\\P_1\\P_2\\P_3\end{bmatrix} = \boxed{\begin{bmatrix}1 & t & t^2 & t^3\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & 0\\-3 & 3 & 0 & 0\\3 & -6 & 3 & 0\\1 & 3 & -3 & 1\end{bmatrix}\begin{bmatrix}P_0\\P_1\\P_2\\P_3\end{bmatrix}}$$



Curva cubica di Bézier

- 4 punti di controllo
- La curva passa tra il primo e l'ultimo punto di controllo
- La curva è tangente in $\mathbf{P_1}$ a ($\mathbf{P_1}$-$\mathbf{P_2}$) ed in $\mathbf{P_4}$ a ($\mathbf{P_4}$-$\mathbf{P_3}$)



Bézier Patch

Bezier Patch di grado *n*

$$P(u,v) = \sum_{j=0}^{m} \sum_{i=0}^{n} P_{i,j} B_{i,n}(u) B_{j,m}(v)$$

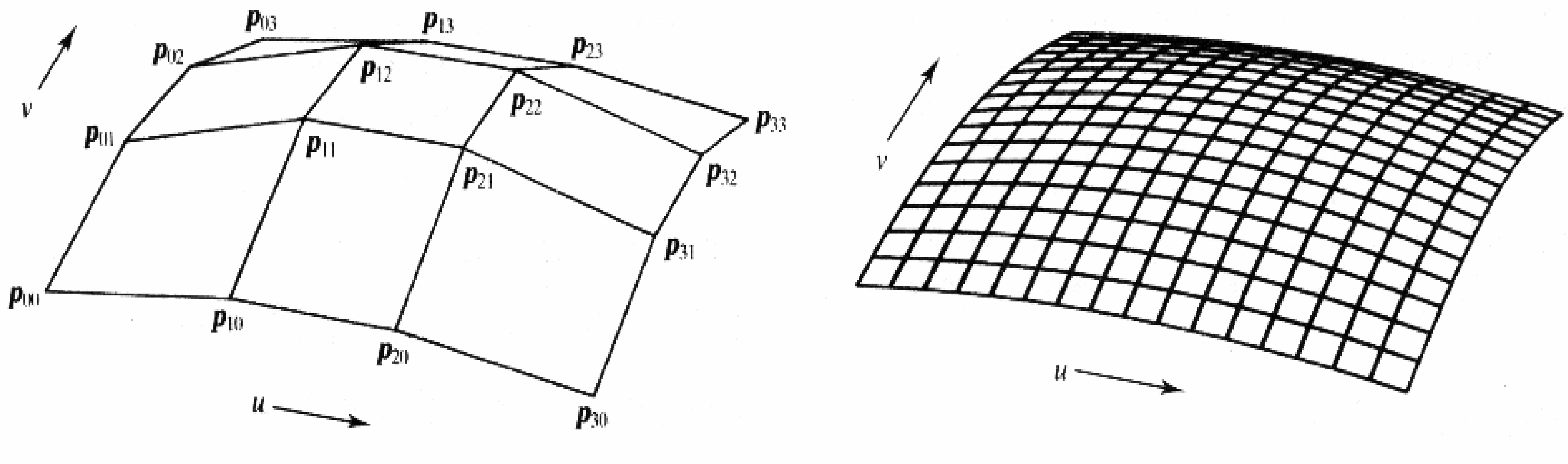
$P_{i,j}$: patch di controllo

$B_{i,n}(u)$: Polinomio di Bernstein di grado n definito per $0 \le u \le 1$

$B_{j,m}(v)$: Polinomio di Bernstein di grado m definito per $0 \le v \le 1$

Patch bicubico di Bézier

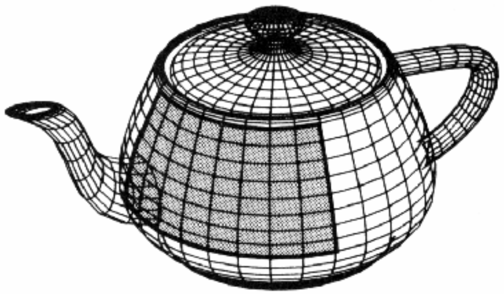
- Poliedro di controllo formato da 16 punti e il risultante patch bi-cubico:



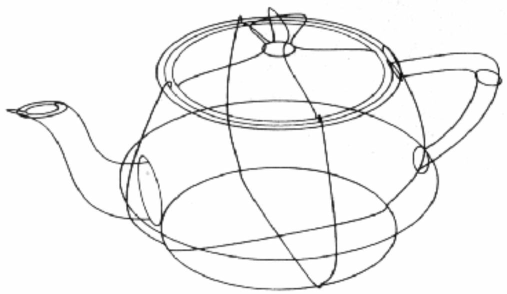
(a) (b)

Esempio di Modellazione con Bézier Patches

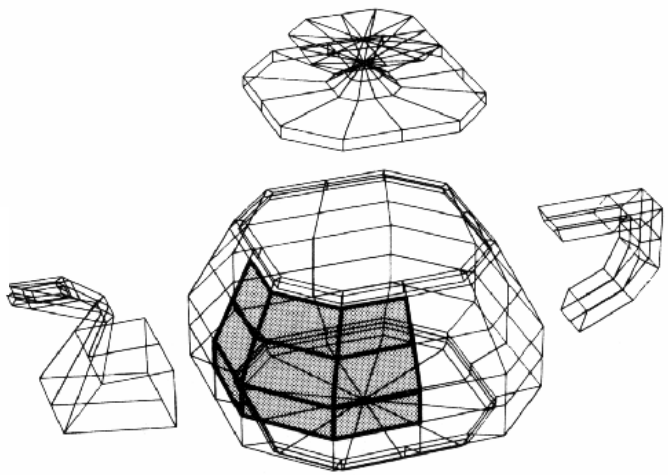
Utah teapot , 32 patches



Singolo patch evidenziato



lati dei patches



wireframe dei punti di controllo

B-Spline di grado *n*

$$P(t) = \sum_{i=0}^{n} P_i N_{i,k}(t)$$

Funzione di Blending $N_{i,k}(t)$

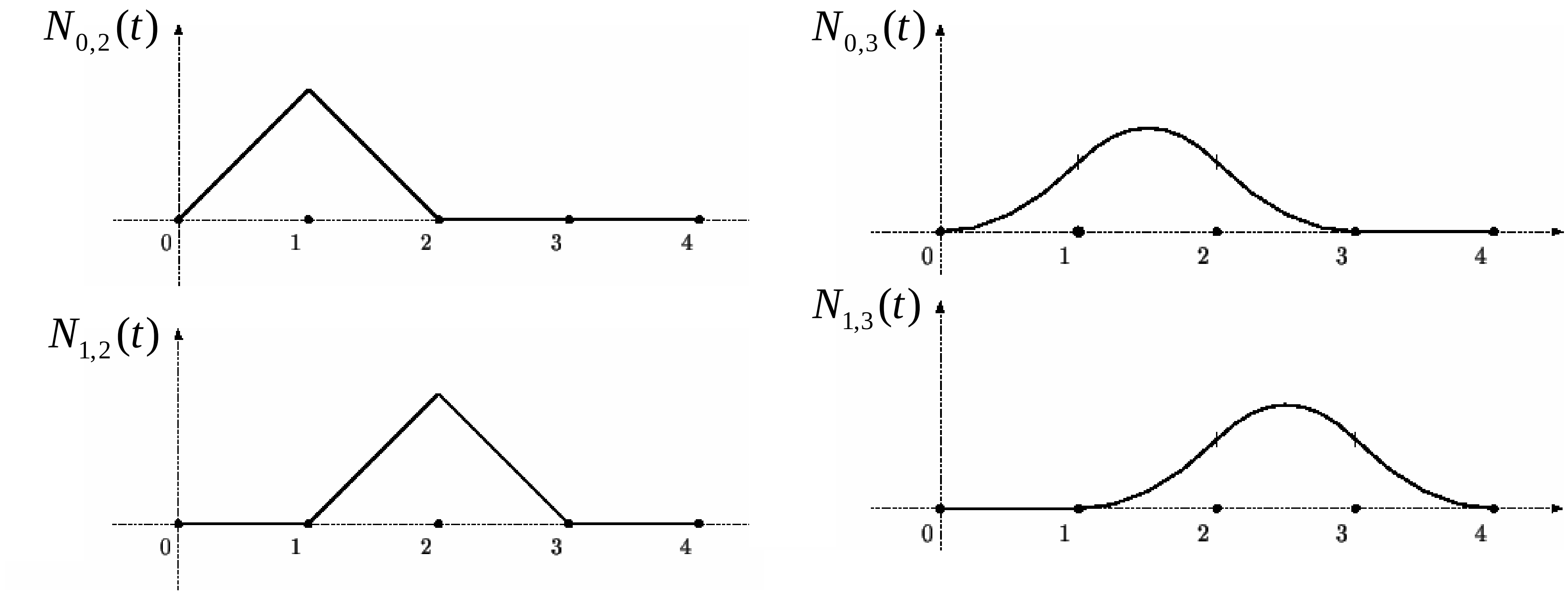
$$N_{i,1}(t) = \begin{cases} 1 & t \in [t_i, t_{i+1}) \\ 0 & \text{otherwise} \end{cases}$$

$\{t_0, t_1, \ldots, t_{n+k}\}$ knots sequence

$k > 1:$

$$N_{i,k}(t) = \left(\frac{t - t_i}{t_{i+k-1} - t_i} \right) N_{i,k-1}(t) + \left(\frac{t_{i+k} - t}{t_{i+k} - t_{i+1}} \right) N_{i+1,k-1}(t) \qquad t \in [t_{k-1}, t_{n+1})$$

B-Splines



$$N_{0,2}(t) = \frac{t - t_0}{t_1 - t_0} N_{0,1}(t) + \frac{t_2 - t}{t_2 - t_1} N_{1,1}(t) = \begin{cases} t & 0 \le t < 1 \\ 2 - t & 1 \le t < 2 \\ 0 & \text{otherwise} \end{cases}$$

$$N_{1,2}(t) = \frac{t - t_1}{t_2 - t_1} N_{1,1}(t) + \frac{t_3 - t}{t_3 - t_2} N_{2,1}(t) = \begin{cases} t - 1 & 1 \le t < 2 \\ 3 - t & 2 \le t < 3 \\ 0 & \text{otherwise} \end{cases}$$

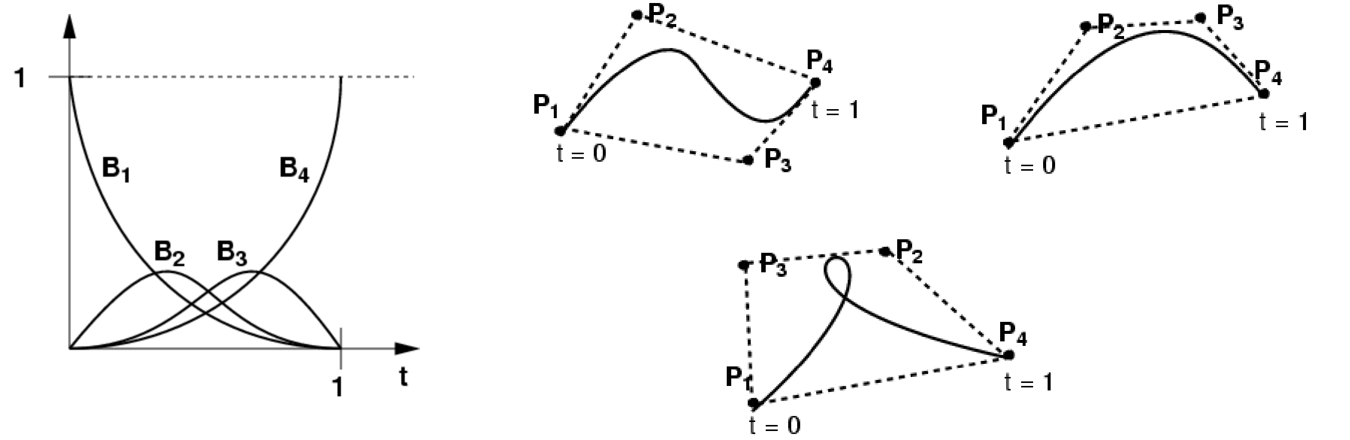
B-Splines

- Il grado della B-Spline è definito dal numero di nodi (k)
- Se i nodi sono equispaziati: $N_{i+1,k}(t) = N_{i,k}(t-t_i)$
- Partizione dell'unità
- Bezier vs B-Splines:
  - n punti => grado n (curva di Bezier)
  - La curva di Bezier deve passare dal punto di controllo iniziale e finale
  - Le B-Splines sono <u>locali</u> ($N_{i,p}(t) \neq 0$, $t \in [t_i , t_{i+p+1})$)

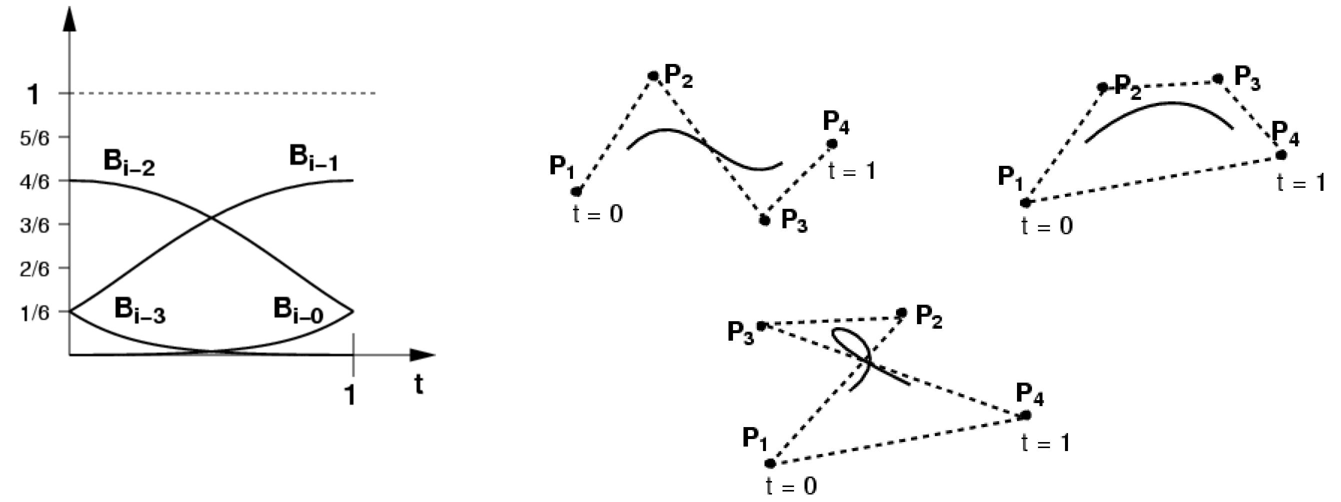
=> B-Splines sono più complesse ma più flessibili!!

- Le relazioni tra I punti di controllo sono differenti

Bézier



BSpline



Non-Uniform Rational B-Spline

$$P(t) = \frac{\sum_{i=0}^{n} N_{i,k}(t) w_i P_i}{\sum_{i=0}^{n} N_{i,k}(t) w_i}$$

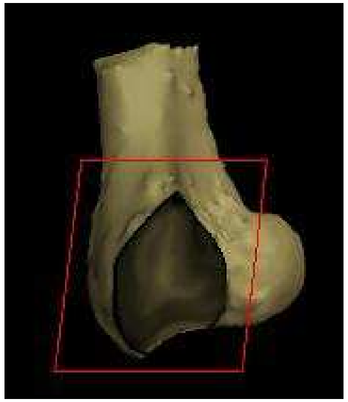
- Non-uniform = spazio non uniforme tra le funzioni di blending (in altre parole i nodi non sono equispaziati)
- Rational = rapporto di polinomi
  - *Le coniche possono essere rappresentate solamente con il rapporto di polinomi*

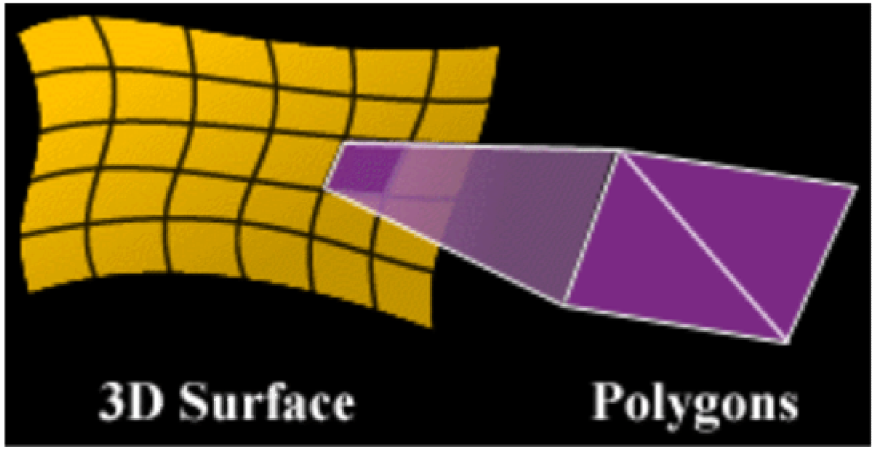
Cerchio unitario: $\left[\frac{1-t^2}{1+t^2}, \frac{2t}{1+t^2}\right] \quad 0 \le t \le 1$

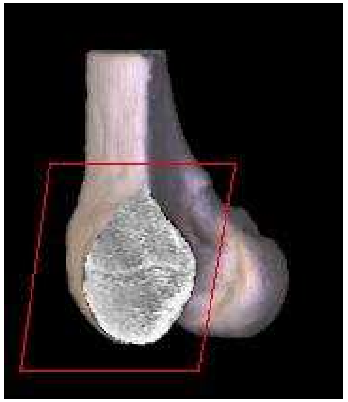
- Le superfici parametriche possono essere convertite facilmente a mesh poligonali tramite campionamento.
- La parametrizzazione è ovvia.
- L'elaborazione geometrica che vi si può fare è analitica.
- Le superfici che si ottengono sono smooth e praticamente qualsiasi forma può essere rappresentata con un buon grado di approssimazione.
- E' una rappresentazione efficiente dal punto di vista della memoria (è sufficiente memorizzare i punti di controllo).

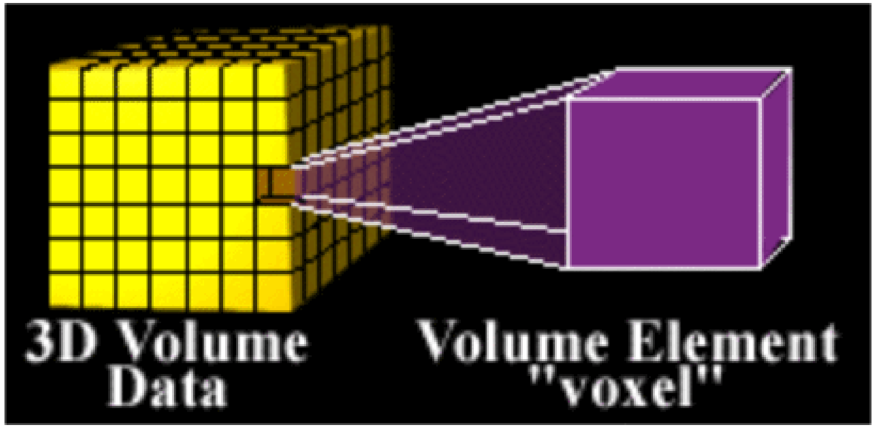
Voxels

Modellazione di Volumi (Voxels)







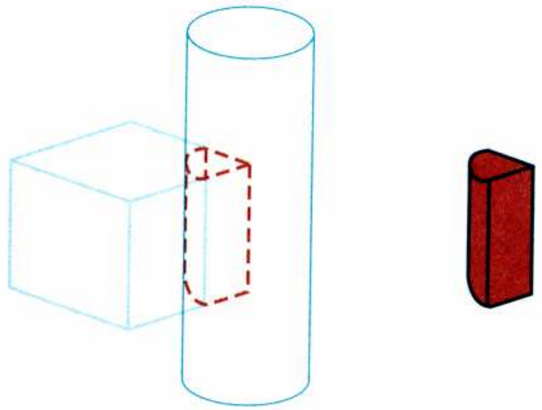


Voxels

- Sono prevalentemente utilizzati quando è importante l'informazione volumetrica (ad esempio nelle applicazioni mediche).
- Richiedono enormi quantità di memoria (e.g. 512x512x512 = 128 MBits).
- Voxels sono utili per rappresentare isosuperfici.
- La conversione da voxel a mesh poligonale è "facile" (algoritmo Marching Cubes). Il viceversa è più complesso e tipicamente computazionalmente costoso.

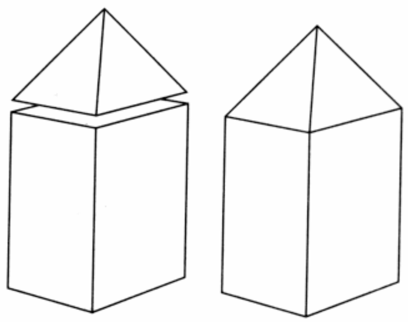
Geometria Solida Costruttiva (CSG)

La forma finale dell'oggetto viene ottenuta combinando I volumi occupati da un insieme sovrapposto di forme tridimensionali tramite operazioni booleane (unione, differenza, intersezione).



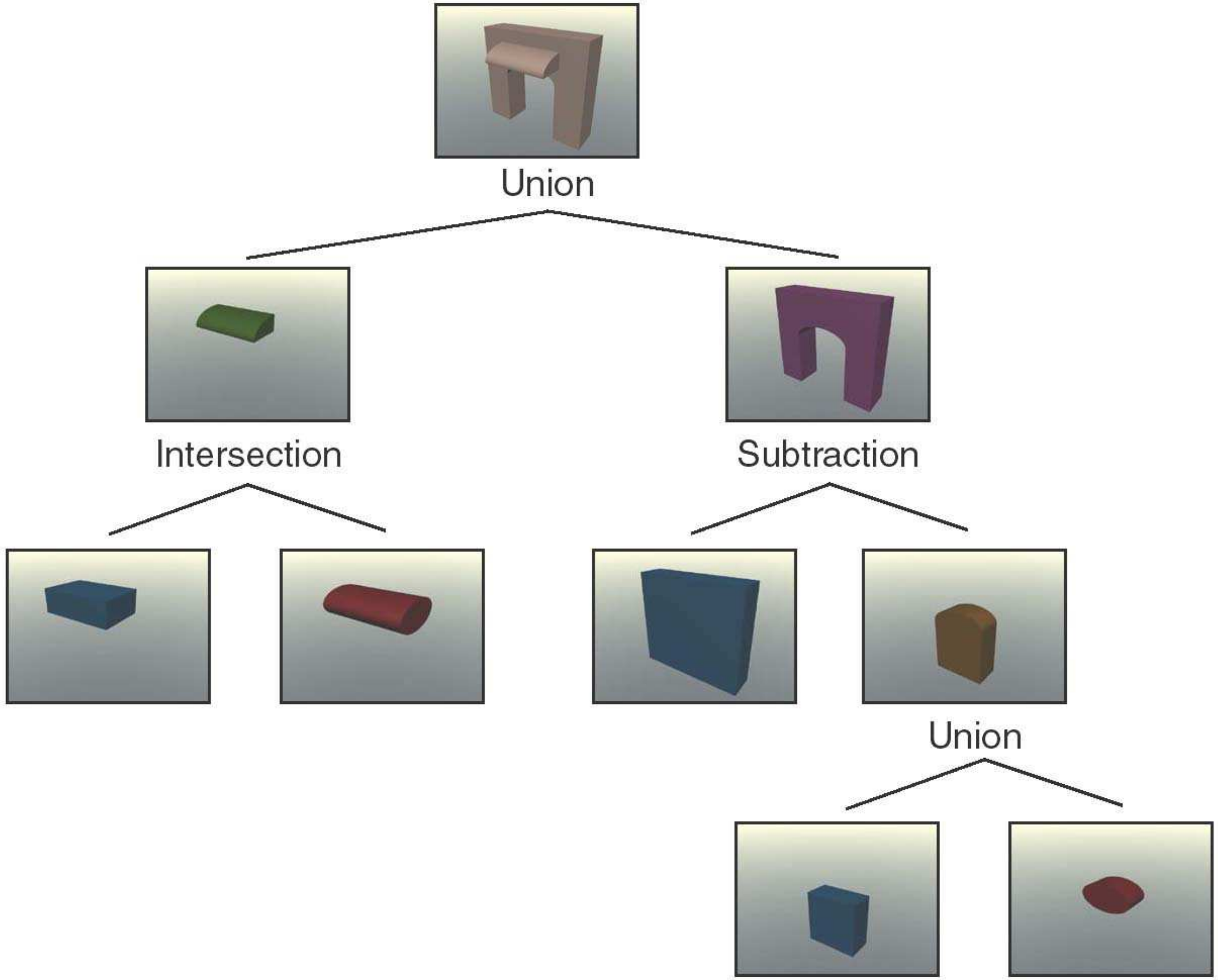
intersezione

differenza



unione

Esempio di CSG



Domande?