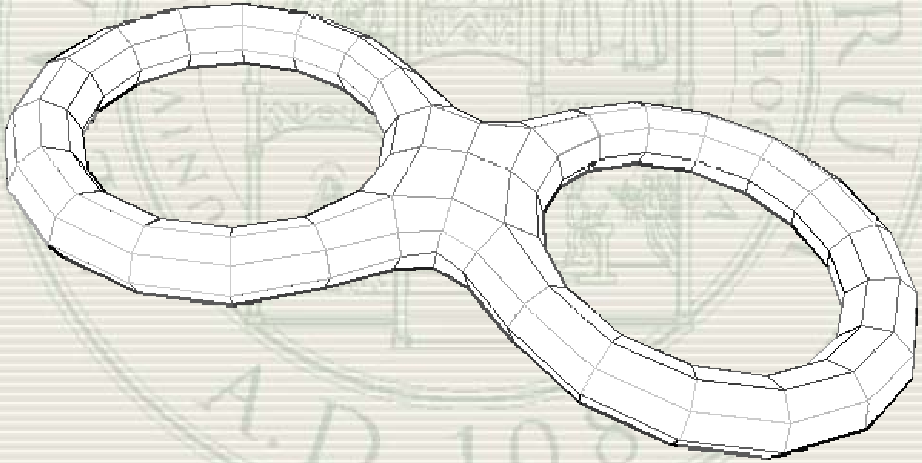
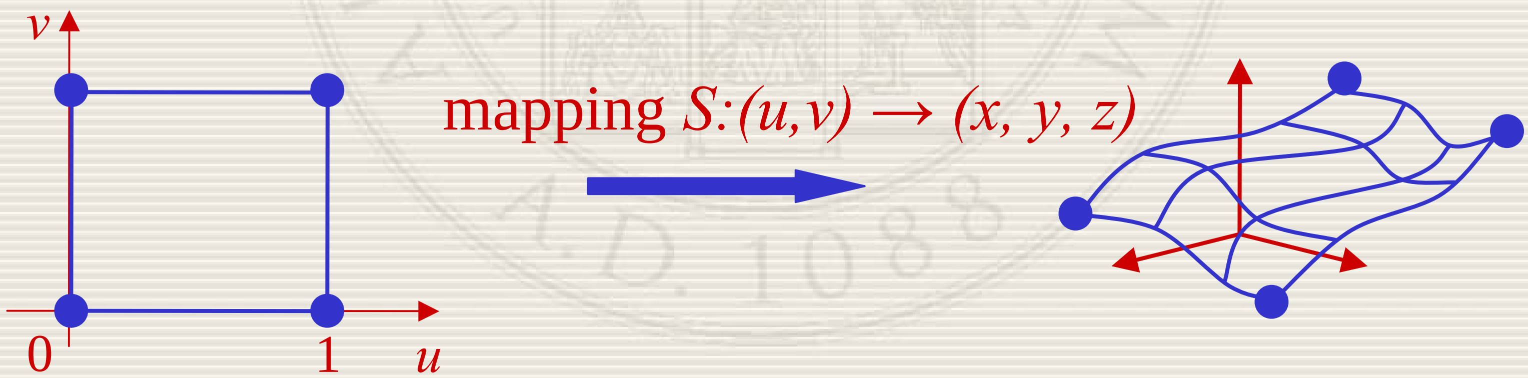
Superfici nella Computer Graphics



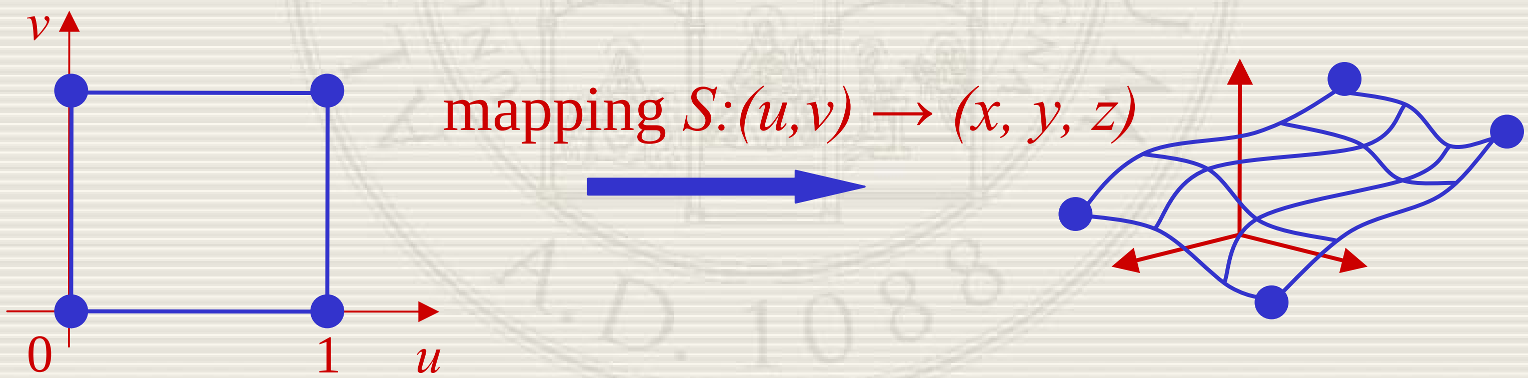
Superfici Parametriche

- Definiamo uno spazio parametrico
  - 2D (per superfici)
- Definiamo un mapping fra lo spazio dei parametri e lo spazio 2D o 3D
  - una funzione che prende valori parametrici (coppie (u,v)) e restituisce punti 2D/3D
- Il risultato è una superficie in forma parametrica (funzione vettoriale)



Superfici Parametriche

- Le superfici parametriche che andiamo a definire hanno una topologia rettangolare (dominio rettangolare)
- In CG, le superfici parametriche sono a volte chiamate *patches*, *curved surfaces*, o solo *surfaces*
- In CG ci sono anche superfici non-parametriche (implicite o algebriche), ma sono molto meno usate



Motivazioni

- Nella CG le mesh poligonali servono per rappresentare gli oggetti 3D; le curve e superfici in forma parametrica sono uno strumento utile per progettare oggetti e definire mesh poligonali di miglior qualità.

Le mesh poligonali hanno tanti parametri quanti vertici

- Le curve e superfici sono definite da pochi parametri e questo rende semplice progettarle e/o modificarle
- Le normali possono essere definite correttamente in ogni punto
- Le curve e superfici parametriche sono più facili da animare che le mesh poligonali

Superfici Parametriche

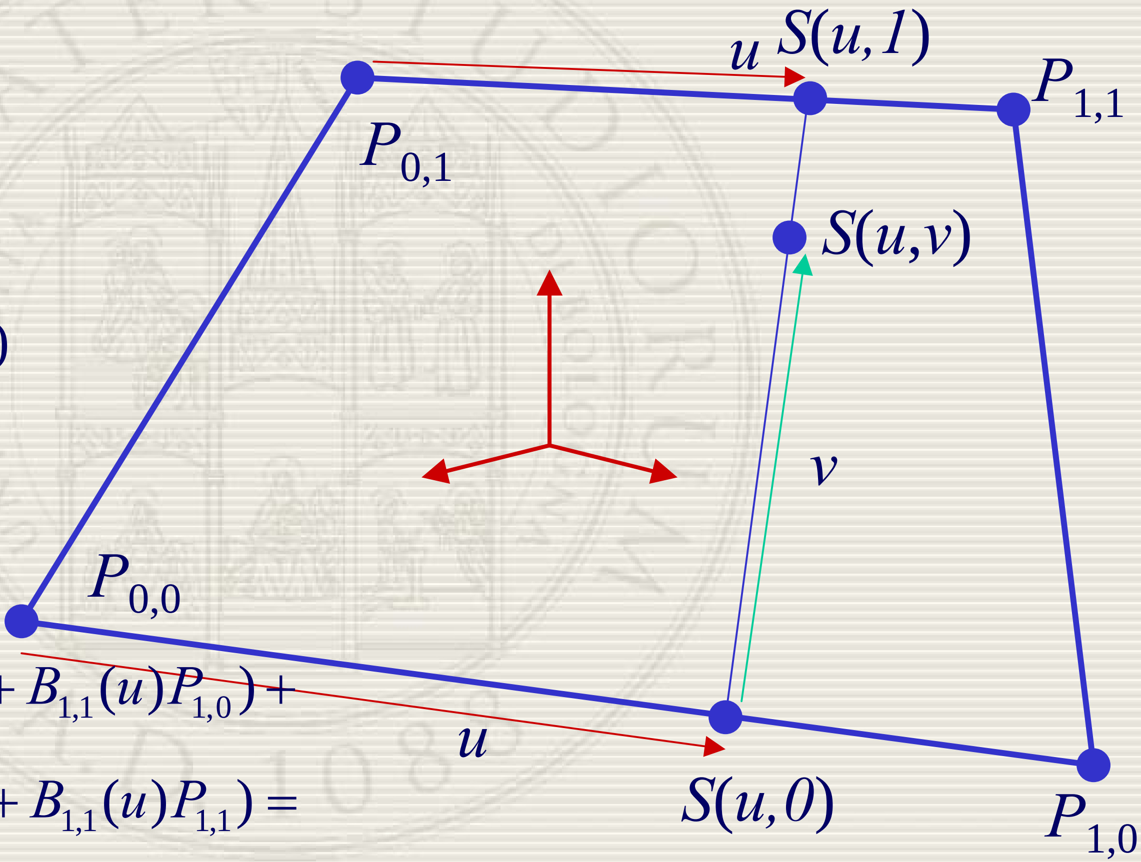
Definiamo punti di una superficie in termini di due parametri.

Il caso più semplice è la superficie bilineare ottenuta per interpolazione lineare di due curve.

$$\hat{C}(u) = (1-u)P_{0,0} + uP_{1,0}$$

$$\tilde{C}(u) = (1-u)P_{0,1} + uP_{1,1}$$

$$S(u,v) = (1-v)\hat{C}(u) + v\tilde{C}(u)$$



$$B_{0,1}(u) = 1-u, \quad B_{1,1}(u) = u$$

$$B_{0,1}(v) = 1-v, \quad B_{1,1}(v) = v$$

$$S(u,v) = B_{0,1}(v)(B_{0,1}(u)P_{0,0} + B_{1,1}(u)P_{1,0}) +$$

$$+ B_{1,1}(v)(B_{0,1}(u)P_{0,1} + B_{1,1}(u)P_{1,1}) =$$

$$= \sum_{i=0}^{1}\sum_{j=0}^{1} P_{i,j}B_{i,1}(u)B_{j,1}(v)$$ Superficie prodotto tensoriale

Superficie Prodotto Tensoriale

- Viene definita su un dominio rettangolare
  - I parametri variano all'interno della regione rettangolare: $0 \le \mathbf{u} \le 1, \;\; 0 \le \mathbf{v} \le 1$
- Per definire una superficie si usa una griglia rettangolare di punti di controllo
  - 4 punti nel caso bi-lineare (come nella slide precedente), molti di più negli altri casi
- La superficie ha la seguente forma:

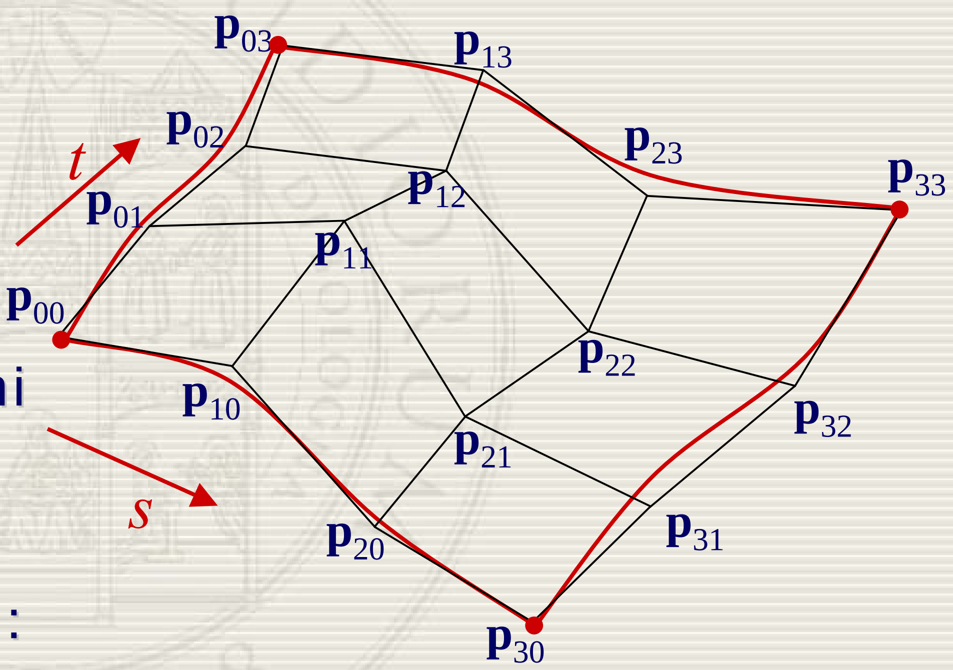
$$S(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} \mathbf{P}_{i,j} B_{i,n}(u) B_{j,m}(v)$$

per certe funzioni $B_{i,n}(u)$ and $B_{j,m}(v)$

Patch di Bézier

I patch di Bézier sono una estensione delle curve di Bézier

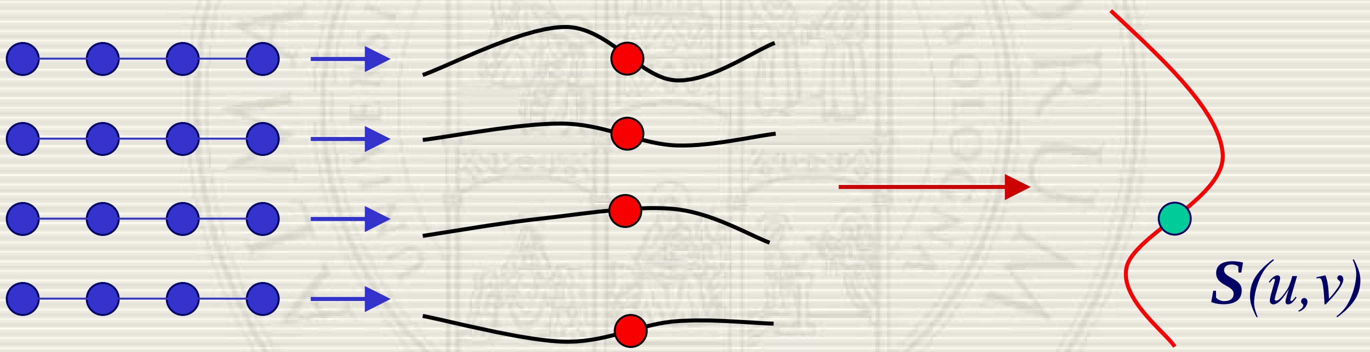
$$S(u,v)=\sum_{i=0}^{n}\sum_{j=0}^{m}\mathbf{P}_{i,j}B_{i,n}(u)B_{j.m}(v)$$



- Come per le curve di Bézier, $B_{i,n}(u)$ e $B_{j,m}(v)$ sono i polinomi di Bernestein rispettivamente di grado n ed m
- Il caso più frequente è $n=m=3$: *cubic Bezier patch*
  - In questo caso servono 4x4=16 punti di controllo, $P_{i,j}$

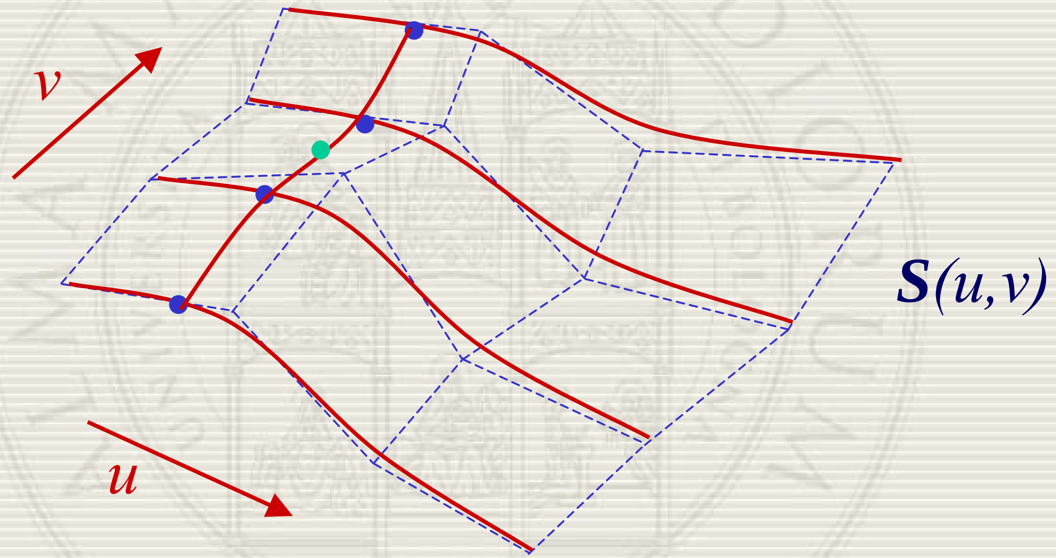
Patch di Bézier

- Ogni riga di 4 punti di controllo definisce una curve di Bézier in u
- Valutare ciascuna di queste curve in uno stesso valore parametrico u fornisce 4 punti di controllo virtuali
- I punti di controllo virtuali definiscono una curva di Bézier in v
- Valutare questa curva in v dà il punto $S(u,v)$ della superficie



Valutazione

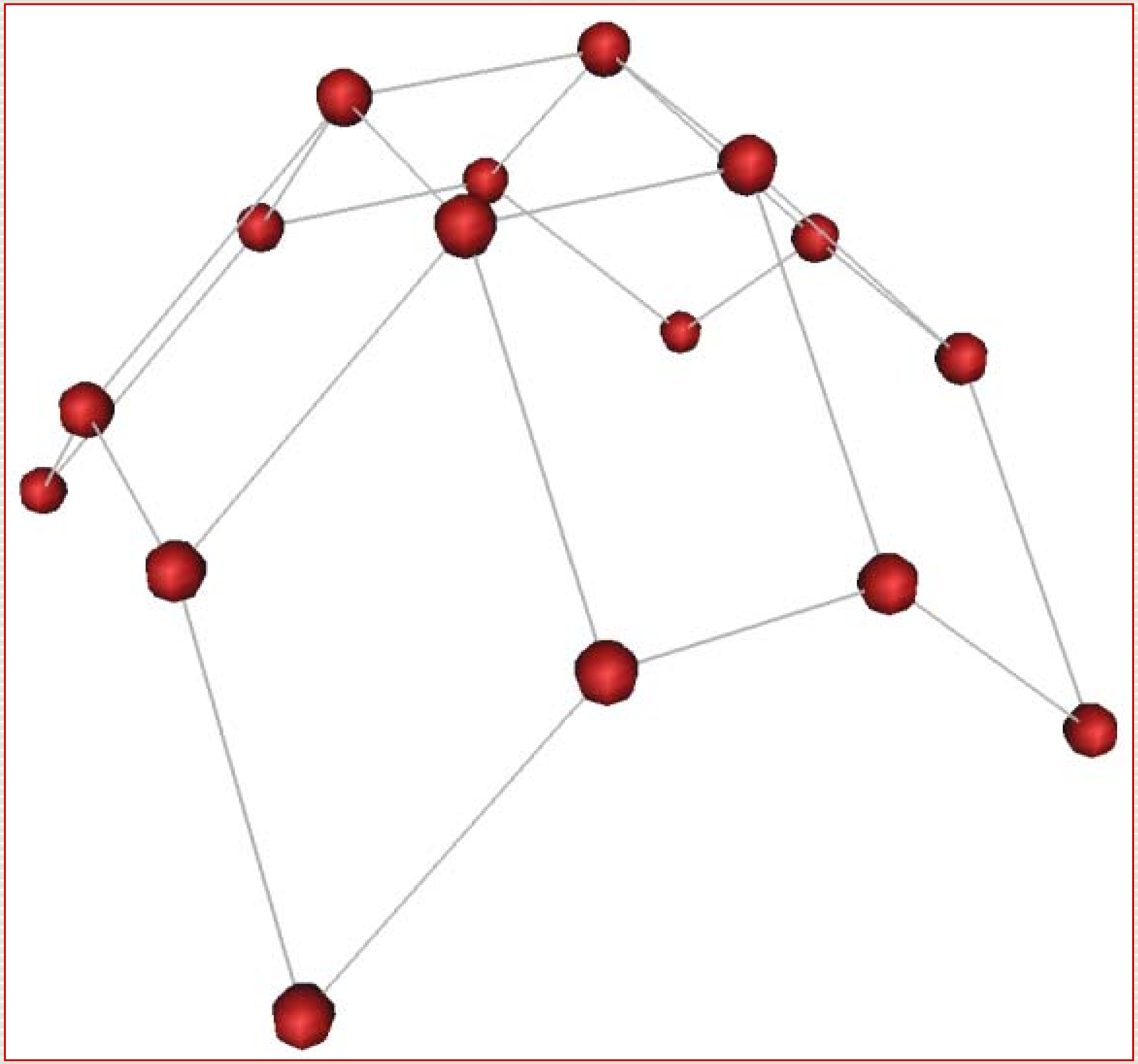
Quanto descritto definisce una superficie, ma è anche un algoritmo efficiente di valutazione della superficie prodotto tensoriale in genere, e di un patch di Bézier in particolare



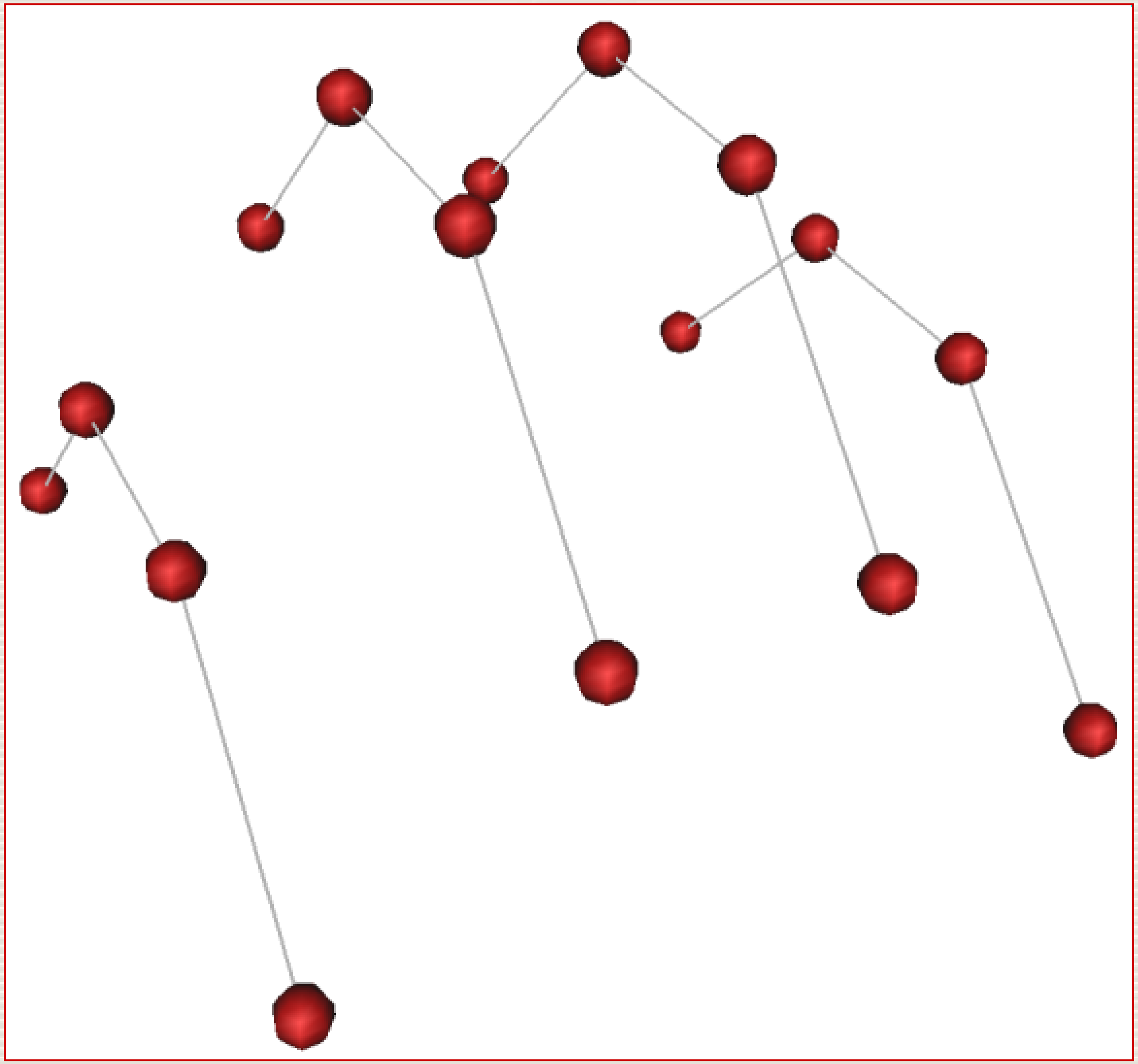
•Le curve di bordo della superficie sono curve 3D di Bézier

•Le curve della superficie con u o v costanti sono curve 3D di Bézier

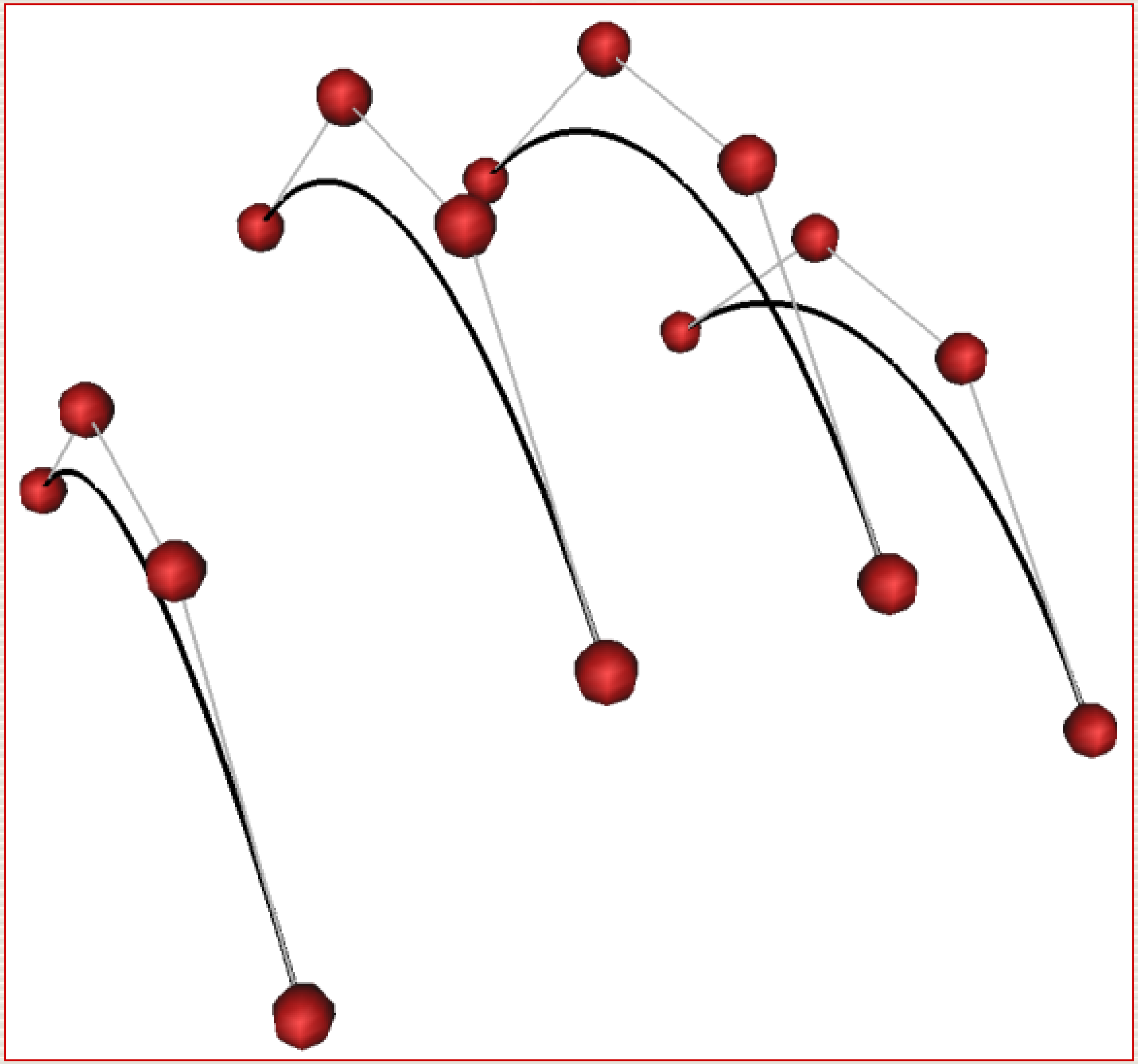
Bézier Patch



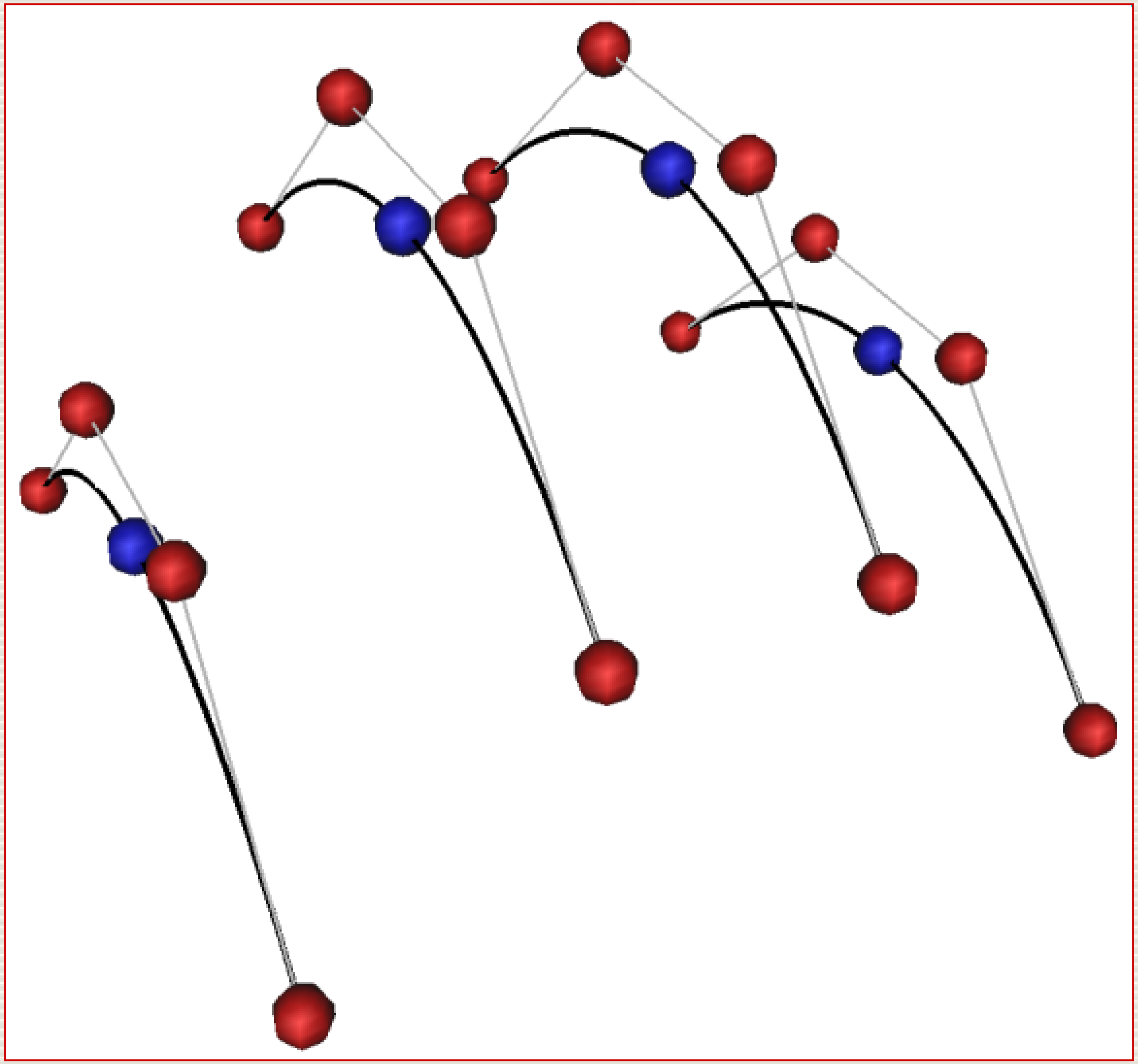
Bézier Patch



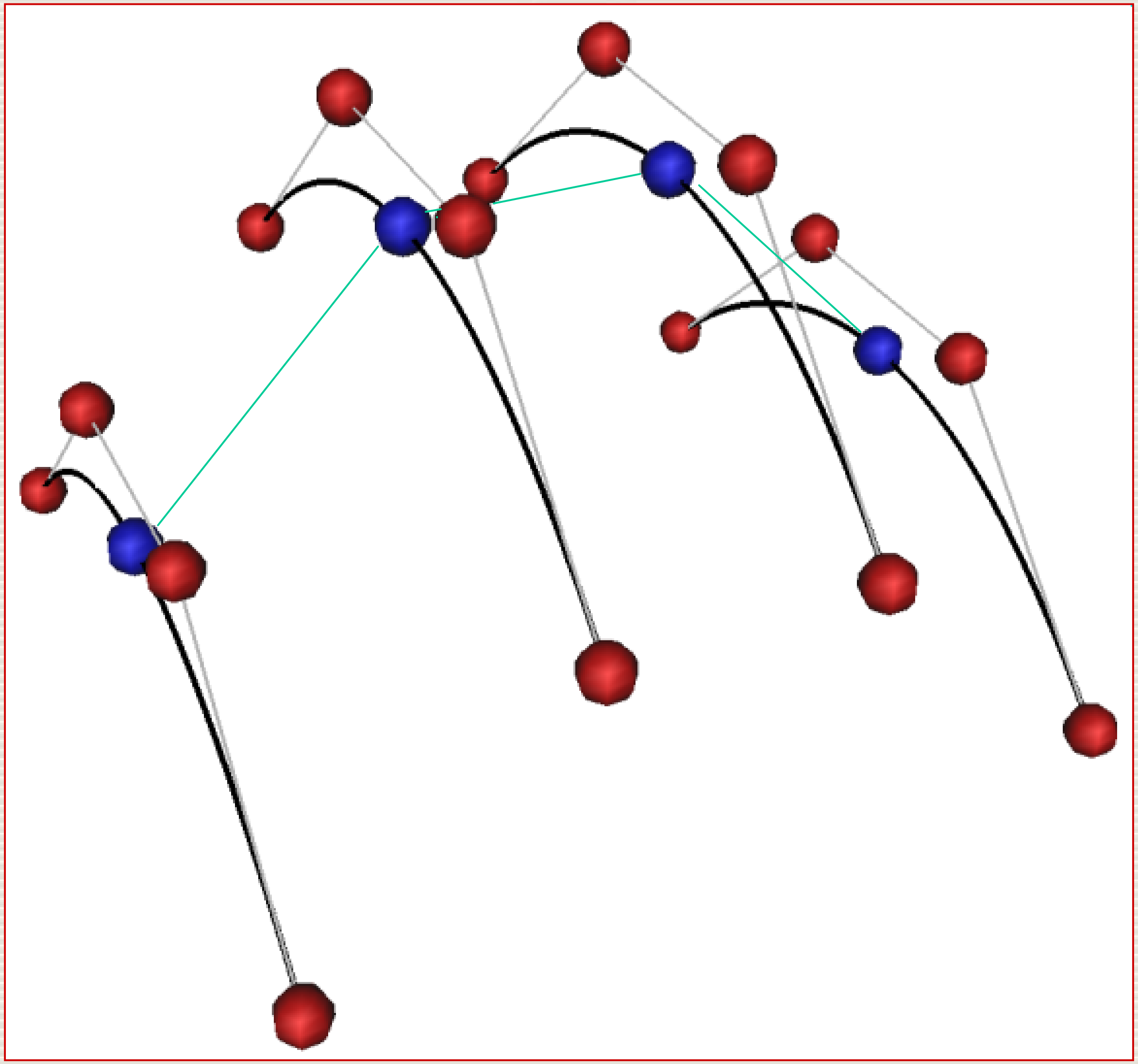
Bézier Patch



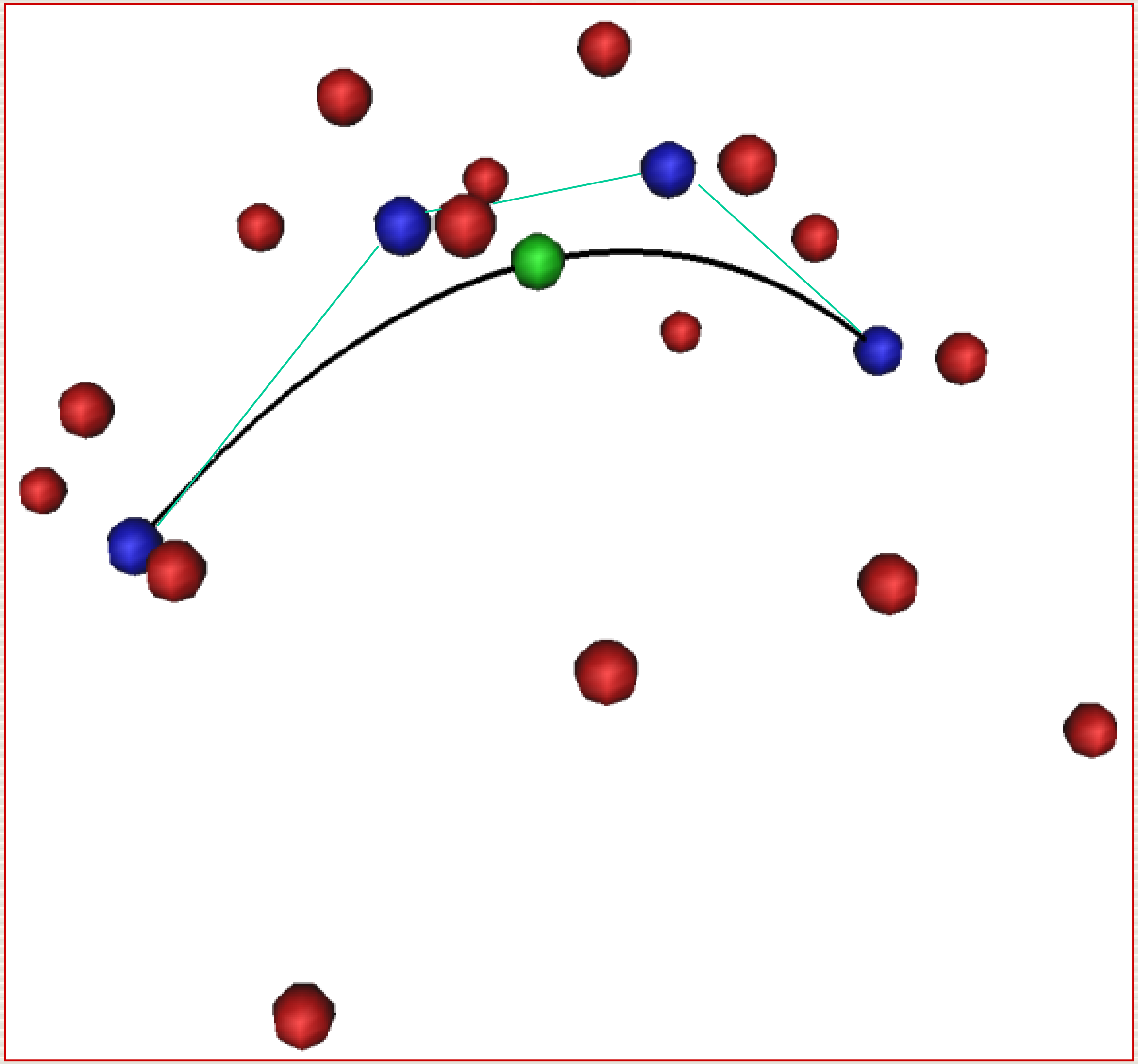
Bézier Patch



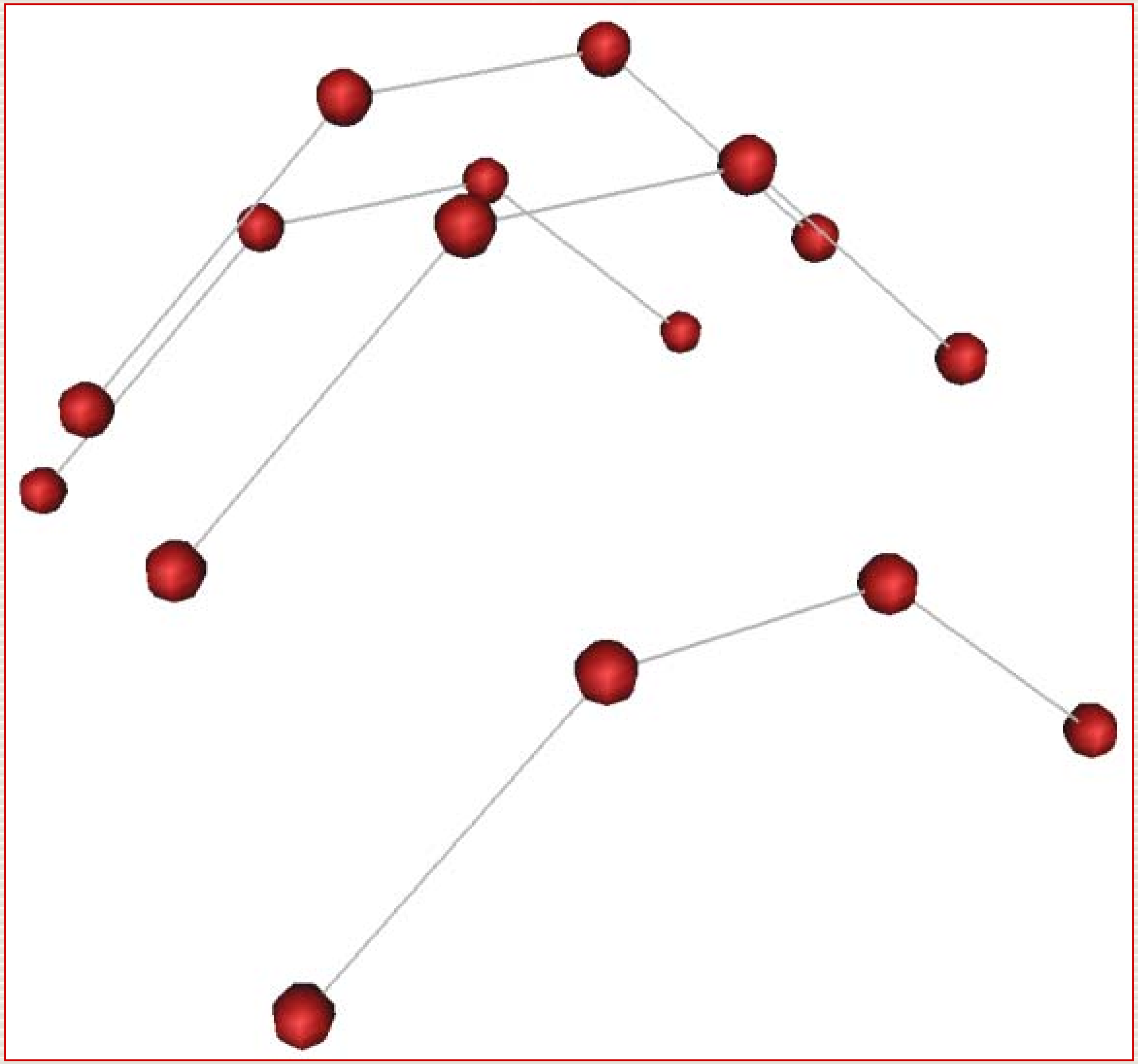
Bézier Patch



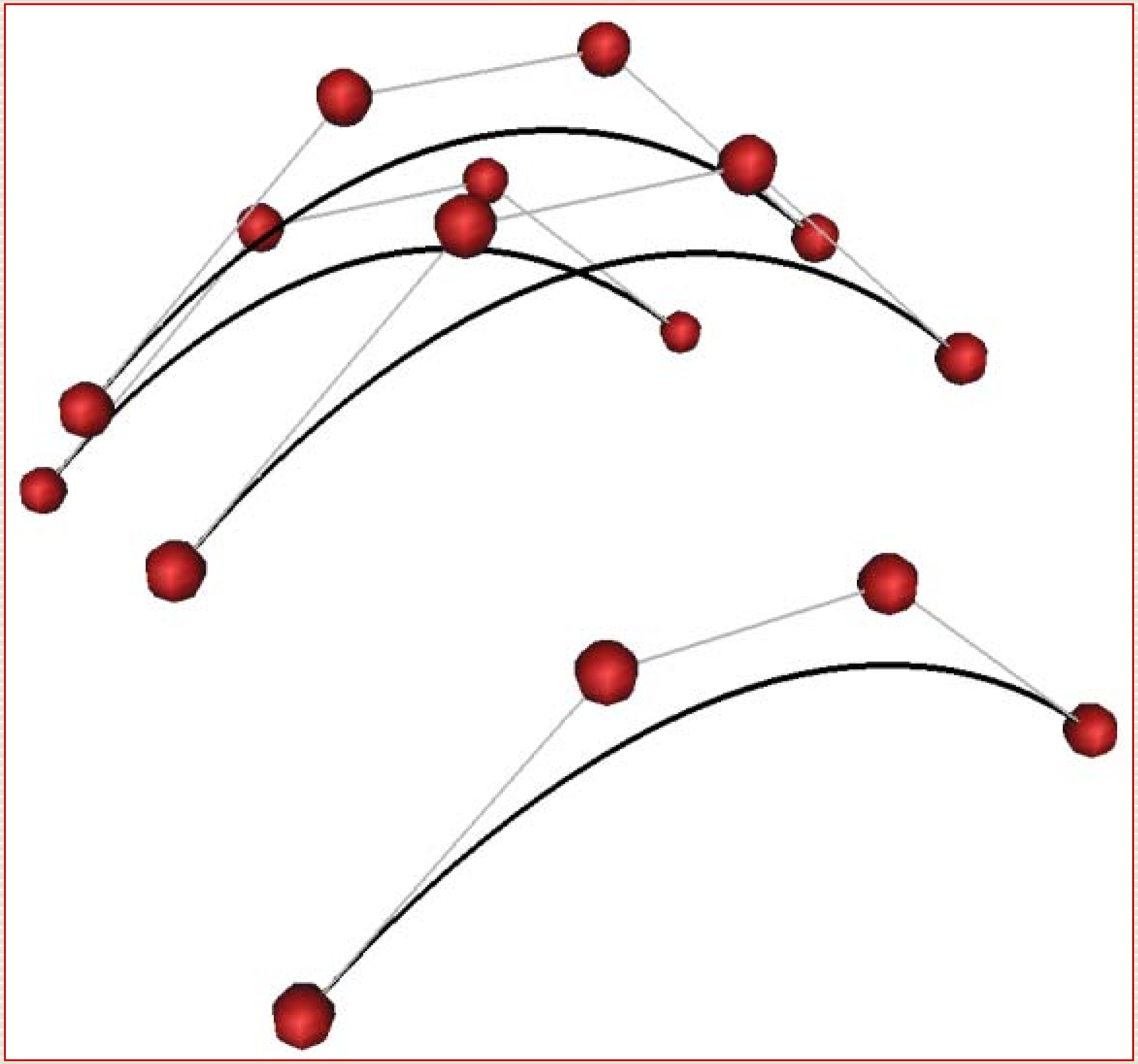
Bézier Patch



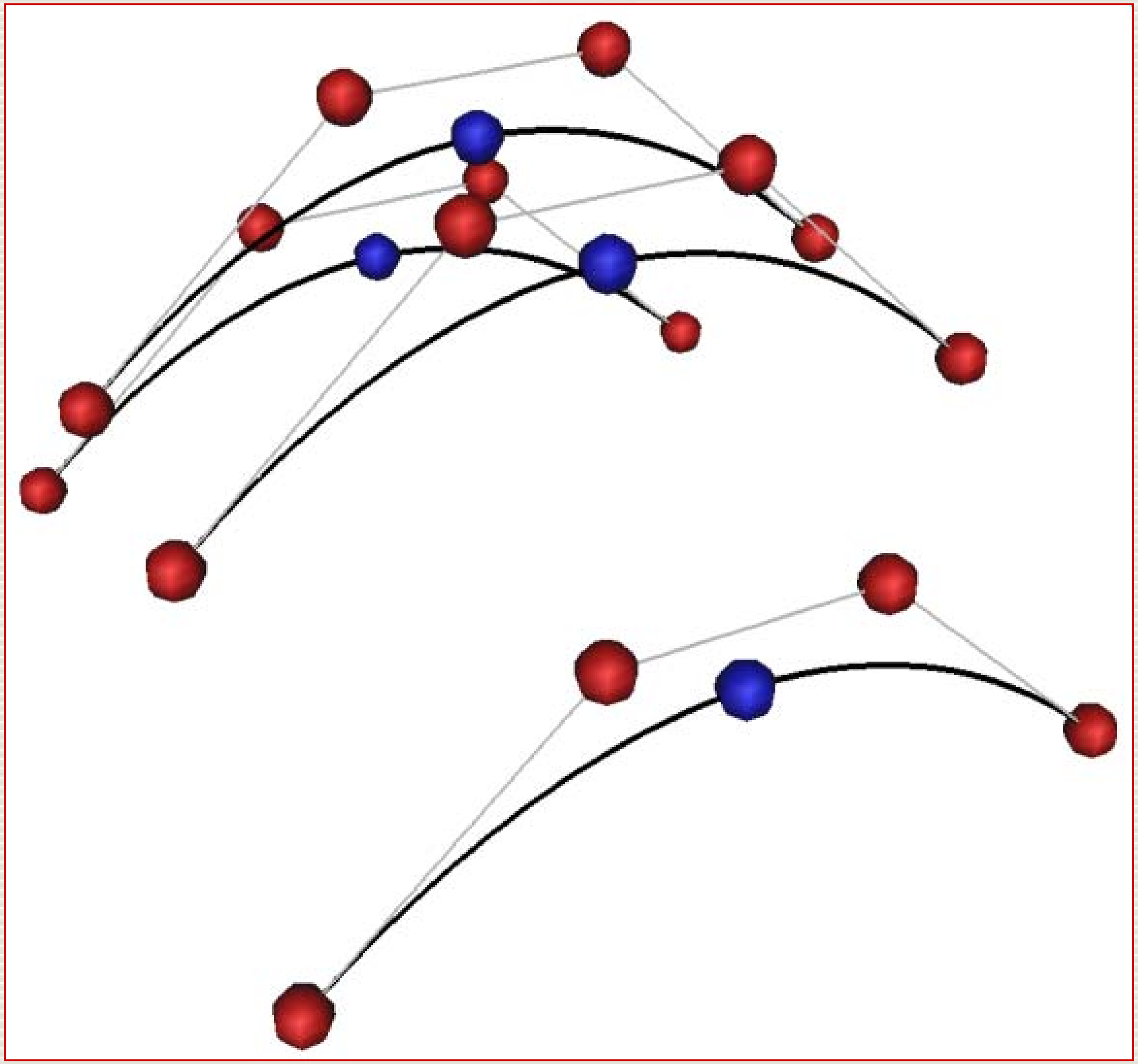
Bézier Patch



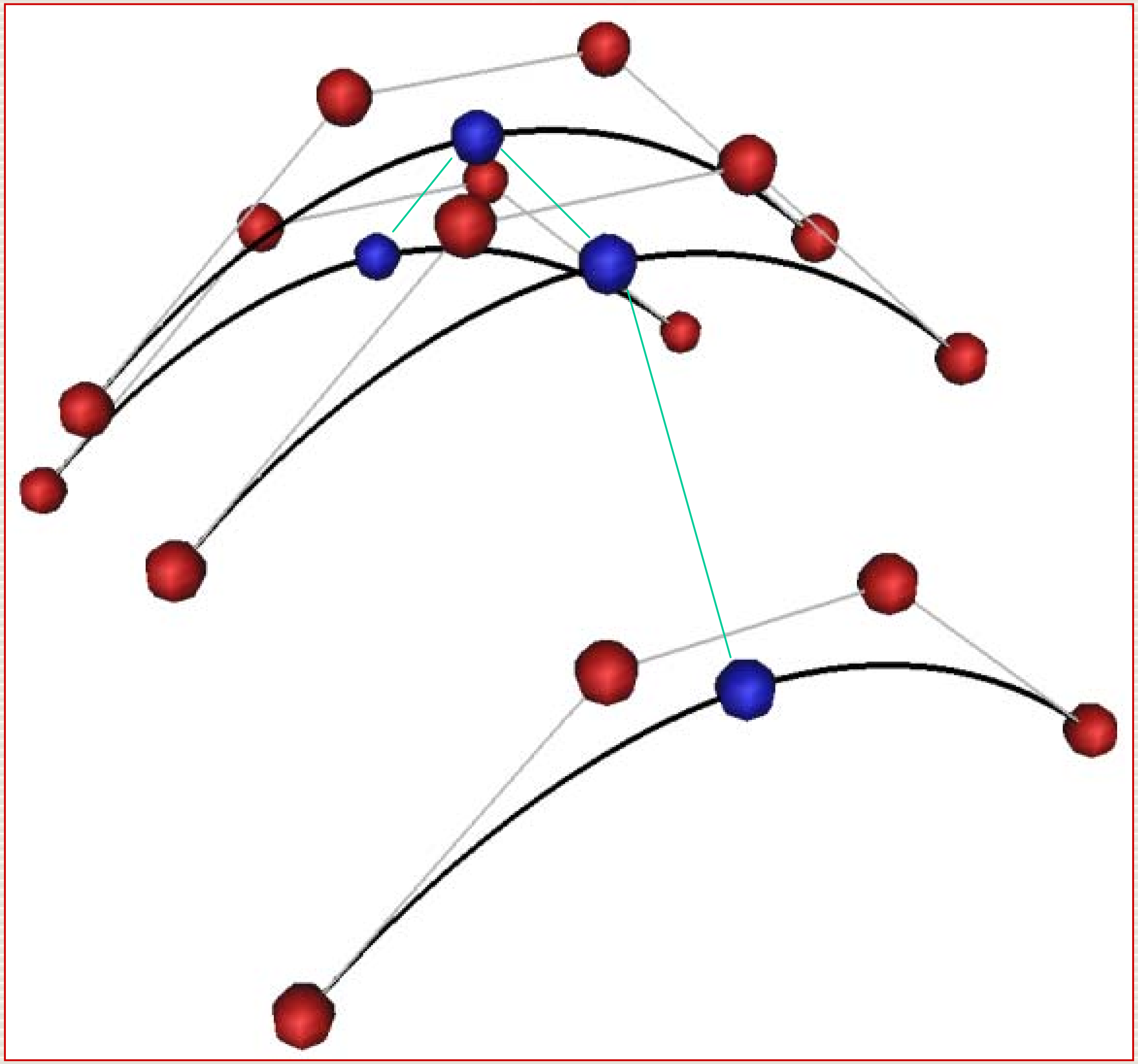
Bézier Patch



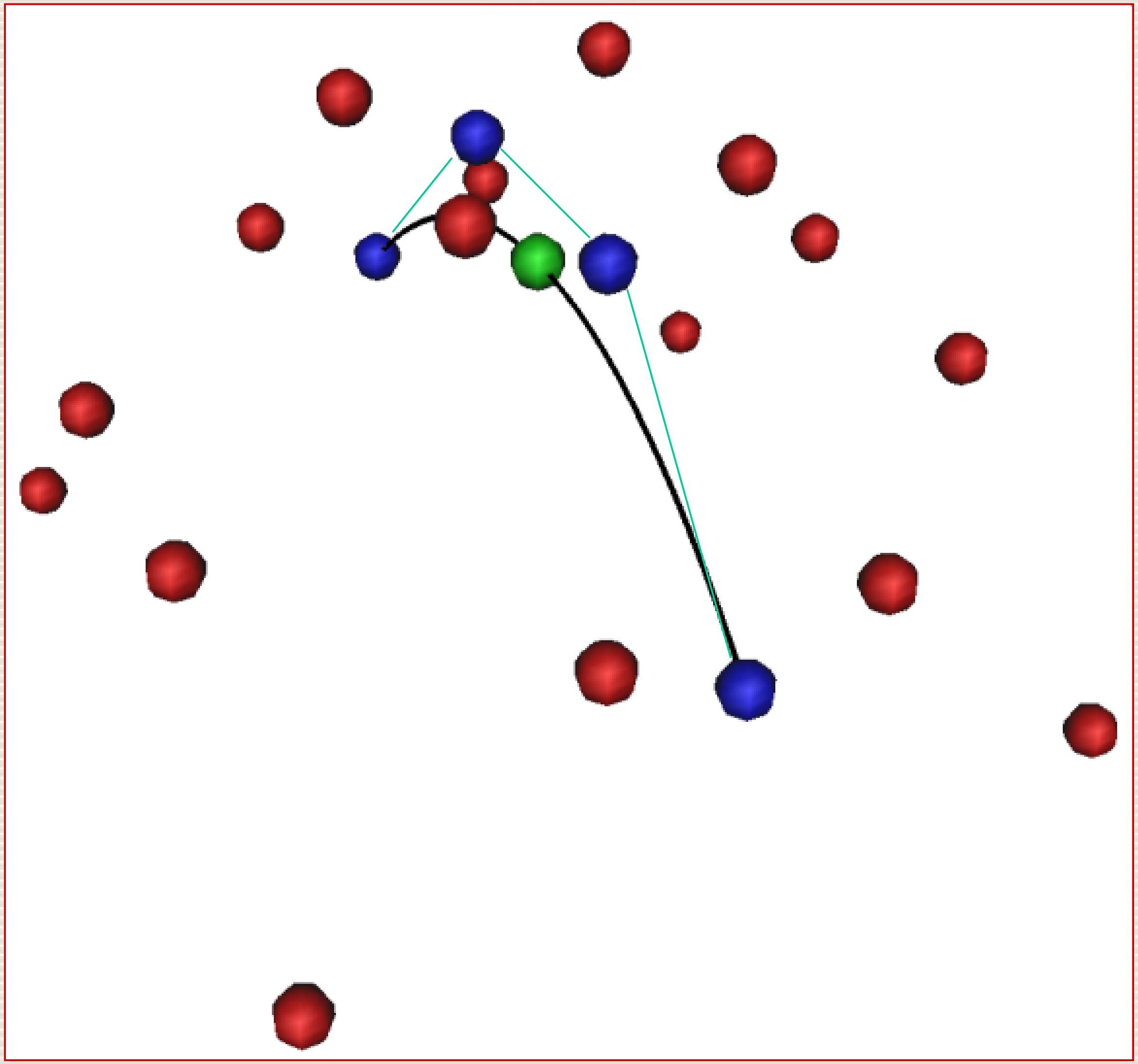
Bézier Patch



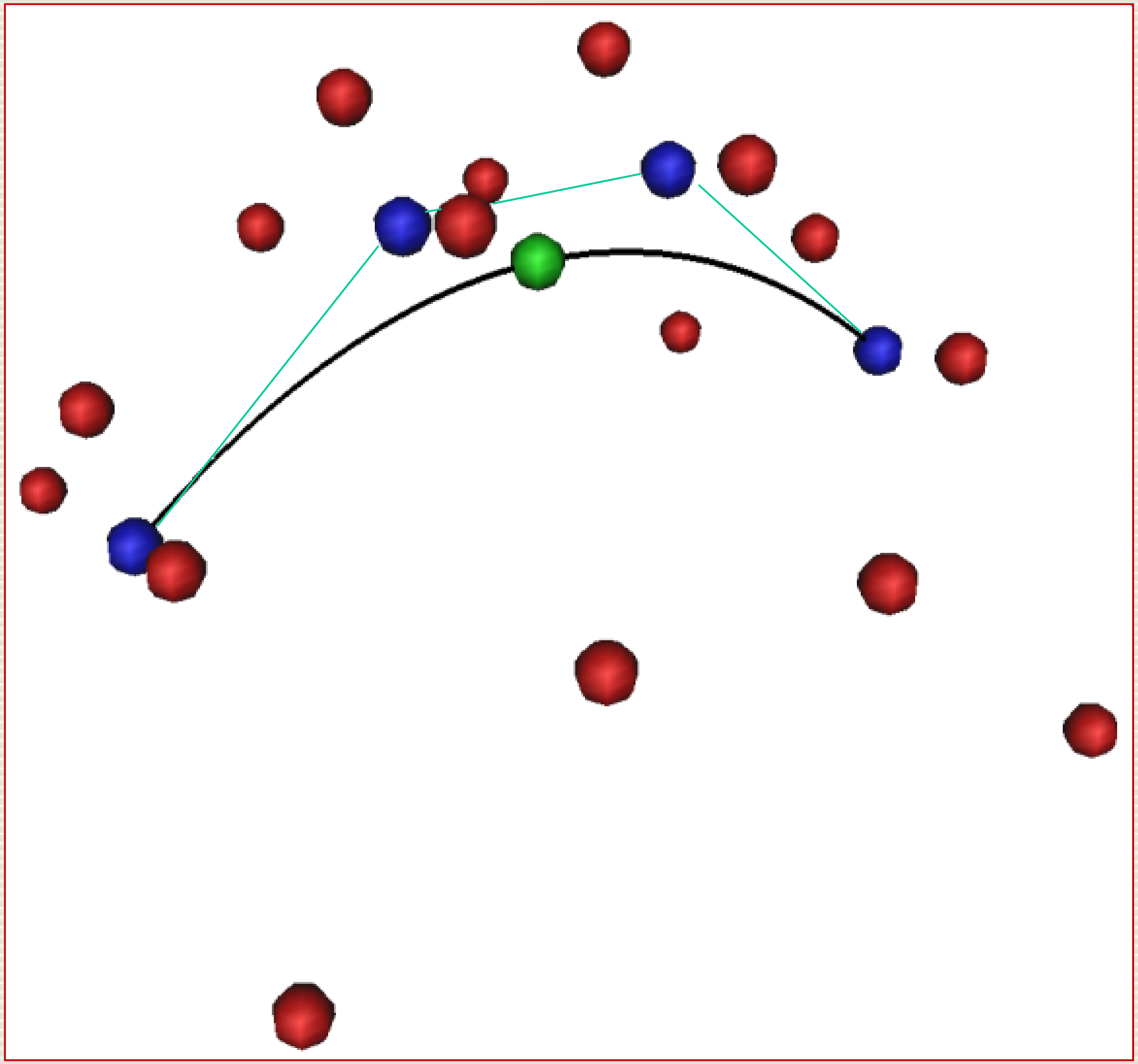
Bézier Patch



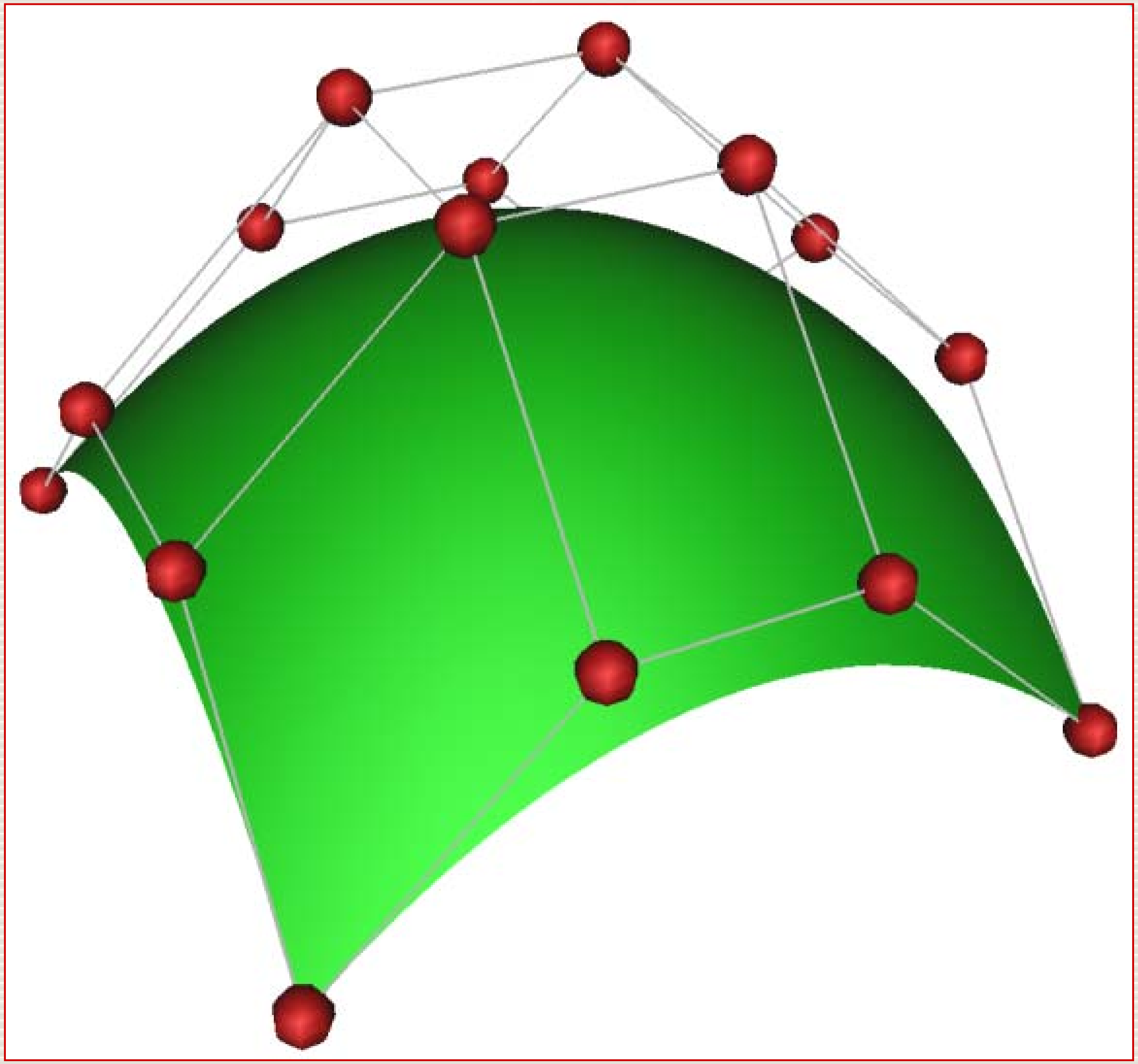
Bézier Patch



Bézier Patch



Bézier Patch



Proprietà dei Patch di Bézier

- I patch di Bézier interpolano alcuni punti di controllo? Quali? Perché?
- Che cosa si può dire sui piani tangenti al patch nei 4 vertici d'angolo?
- Il patch sta nel guscio convesso dei suoi punti di controllo?

Proprietà dei Patch di Bézier

- Il patch interpola i suoi 4 punti di controllo d’angolo
  - Questo deriva dalla proprietà di interpolazione delle curve di Bézier
- Il piano tangente in ogni vertice d’angolo passa per il punto di controllo d’angolo e per i due segmenti della griglia di controllo adiacenti al punto di controllo d’angolo
  - Il piano tangente alla superficie in un punto è il piano che è perpendicolare al vettore normale alla superficie in quel punto
  - La proprietà del piano tangente deriva dalle proprietà della tangente alla curva e da come si calcola il vettore normale
- Il patch sta nel guscio convesso dei suoi punti di controllo
  - Le funzioni base sono a somma 1 e sono non negativa (combinazione convessa)

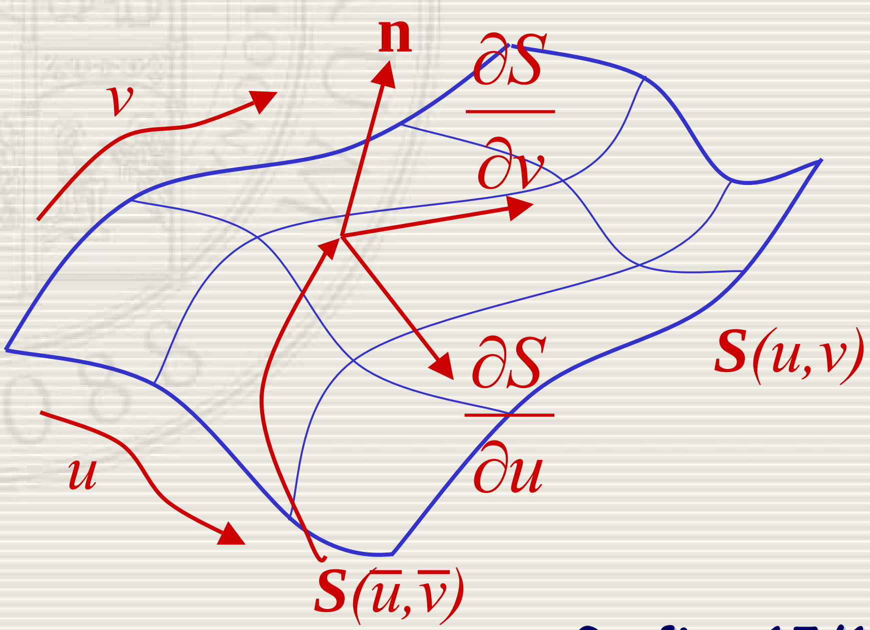
Calcolo dei Vettori Normali

- La derivata parziale nella direzione u è un vettore tangente
- La derivata parziale nella direzione v è un altro
- Si consideri il loro prodotto vettoriale, lo si normalizzi, e si ottiene il vettore normale alla superficie

$$\left.\frac{\partial S}{\partial u}\right|_{\bar{u},\bar{v}} = \sum_{i=0}^{n}\sum_{j=0}^{m} \mathbf{P}_{i,j} \left.\frac{dB_{i,n}(u)}{du}\right|_{\bar{u}} B_{j,m}(\bar{v})$$

$$\left.\frac{\partial S}{\partial v}\right|_{\bar{u},\bar{v}} = \sum_{i=0}^{n}\sum_{j=0}^{m} \mathbf{P}_{i,j} B_{i,n}(\bar{u}) \left.\frac{dB_{j,m}(v)}{dv}\right|_{\bar{v}}$$

$$\mathbf{n} = \left.\frac{\partial S}{\partial u}\right|_{\bar{u},\bar{v}} \times \left.\frac{\partial S}{\partial v}\right|_{\bar{u},\bar{v}} \qquad \hat{\mathbf{n}} = \frac{\mathbf{n}}{\|\mathbf{n}\|}$$



Superfici Spline

Sia data una griglia $(N+n+1)$x$(M+m+1)$ di punti di controllo, si definisca una partizioni nodale estesa in u ed una in v; scelta una direzione, per esempio u, si definiscano gli $M+m+1$ punti di controllo virtuali relativi a $\overline{u}$

$$C_j(\overline{u}) = \sum_{i=1}^{N+n+1} P_{i,j} B_{i,n}(\overline{u}) \quad \text{per } j = 1, \ldots M + m + 1$$

Si definisca quindi la curva

$$S(\overline{u}, v) = \sum_{i=1}^{M+m+1} C_j(\overline{u}) B_{j,m}(v)$$

Questo procedimento al variare di u e v definisce i punti della superficie

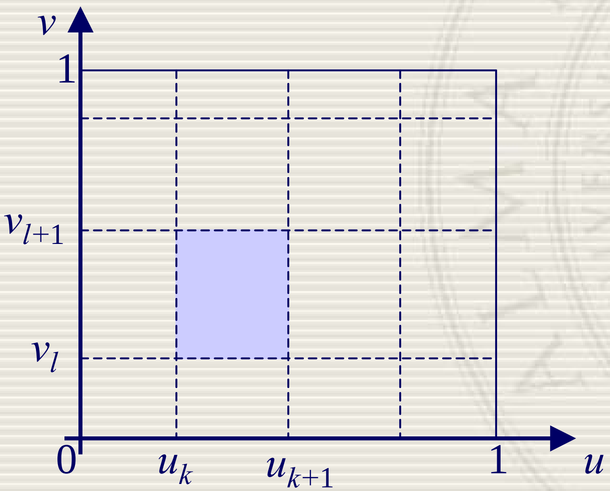
$$S(u, v) = \sum_{i=1}^{N+n+1} \sum_{j=1}^{M+m+1} P_{i,j} B_{i,n}(u) B_{j,m}(v)$$

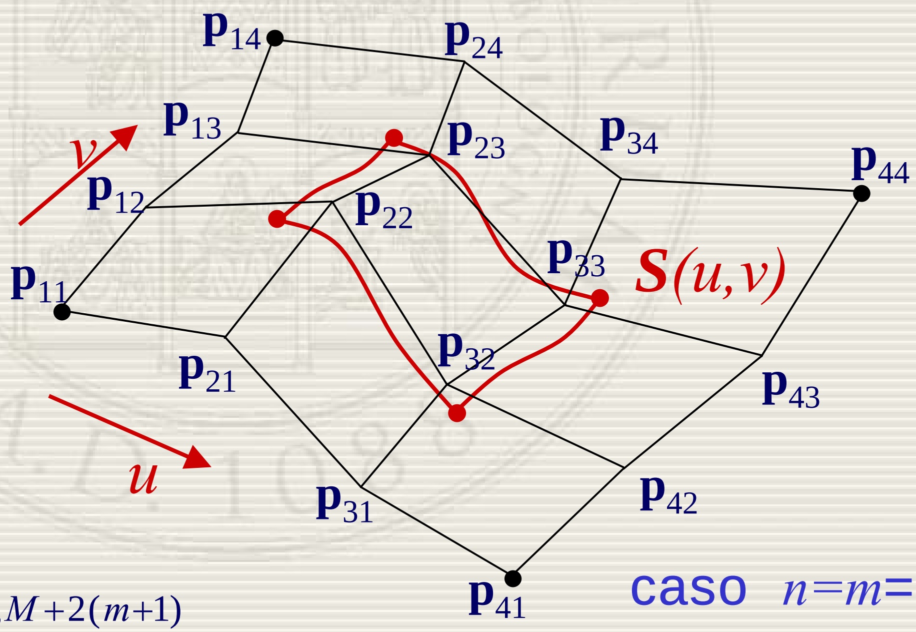
Patch Spline

Nel caso bicubico ($n=m=3$), ogni 4x4 punti di controllo resta definito un patch spline

$$S(u,v)=\sum_{i=k-n}^{k}\sum_{j=l-m}^{l}P_{i,j}B_{i,3}(u)B_{j,3}(v)$$

$$(u,v)\in[u_k,v_l)\mathrm{x}[u_{k+1},v_{l+1})=[k,l)\mathrm{x}[k+1,l+1)$$





Partizioni estese

$\{u_i\}_{i=1,\ldots,N+2(n+1)}$ $\{v_j\}_{j=1,\ldots,M+2(m+1)}$

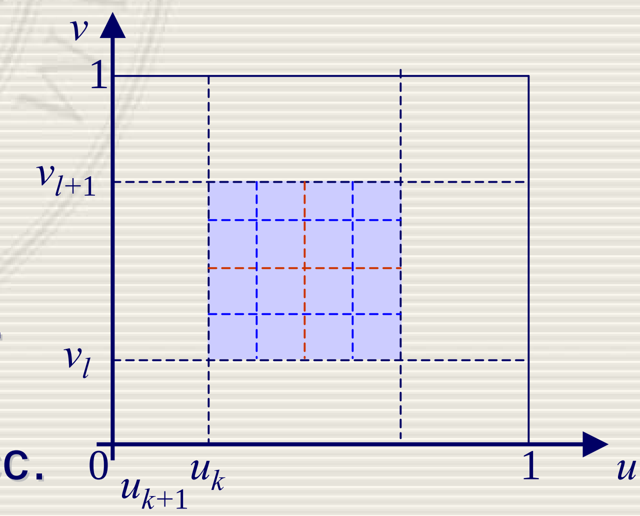
caso $n=m=3$, $k=l=4$

Valutazione Patch Spline via raffinamento

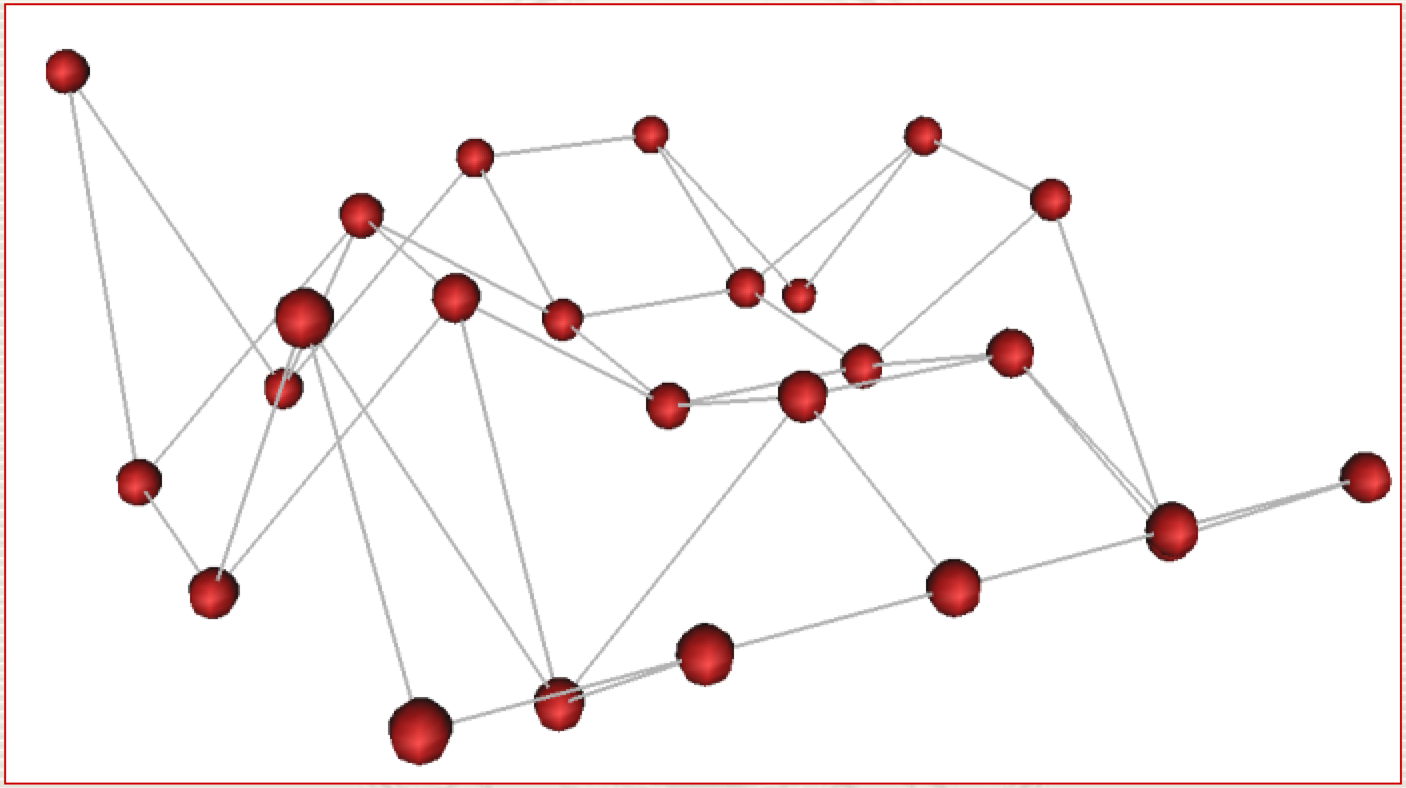
Sia dato un patch spline bicubico (n=m=3); dopo il primo raffinamento il rettangolo nodale [u_k,v_l], viene diviso in 2x2, per cui la griglia necessaria per la valutazione è (n+2)x(m+2),
dopo il secondo raffinamento il rettangolo resta diviso in 4x4, e la griglia è (n+4)x(m+4), ecc.

La regola è molto semplice; ricordiamo che uno spazio spline ha dimensione N+n+1 con N nodi interni.
Intervalli: 1, 2, 4, 8, 16, ...
nodi negli intervalli: 0, 1, 3, 7, 15, ...
numero di CP (n =3): 4, 5, 7, 11, 19, ...
in generale: n+1, n+2, n+4, n+8, n+16,...

e il patch (n+1)x (n+1), (n+2)x (n+2), ecc.

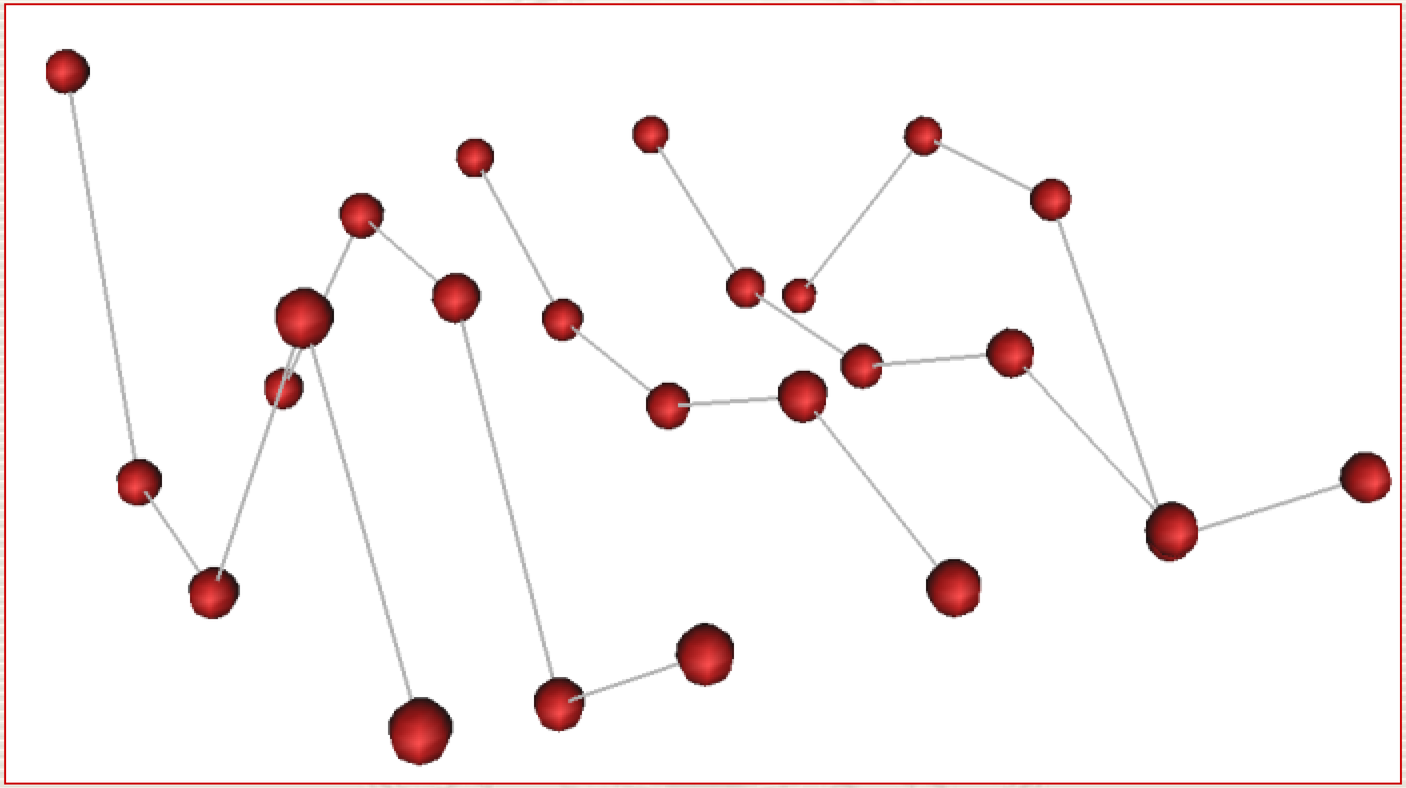


Patch Spline



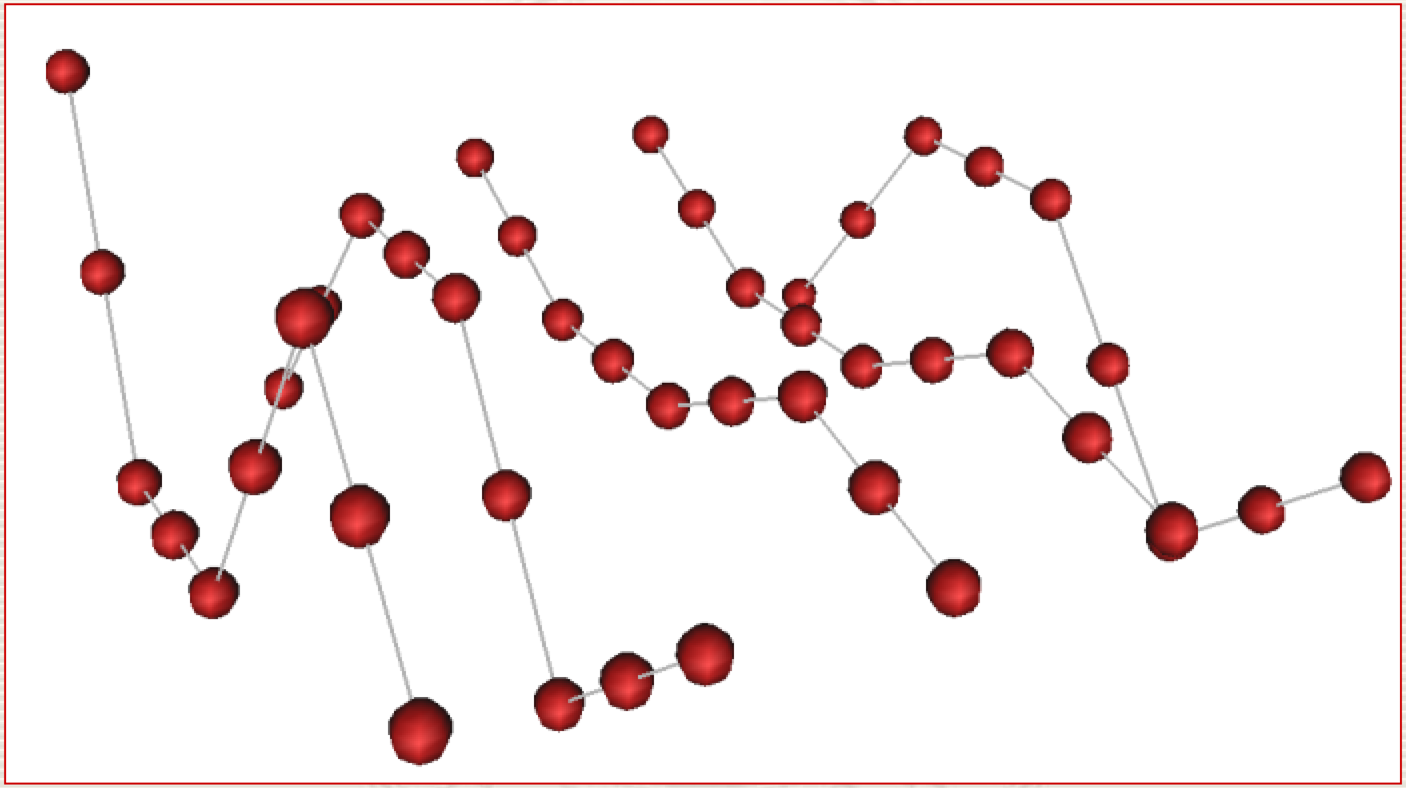
Valutazione approssimata via raffinamento

Patch Spline



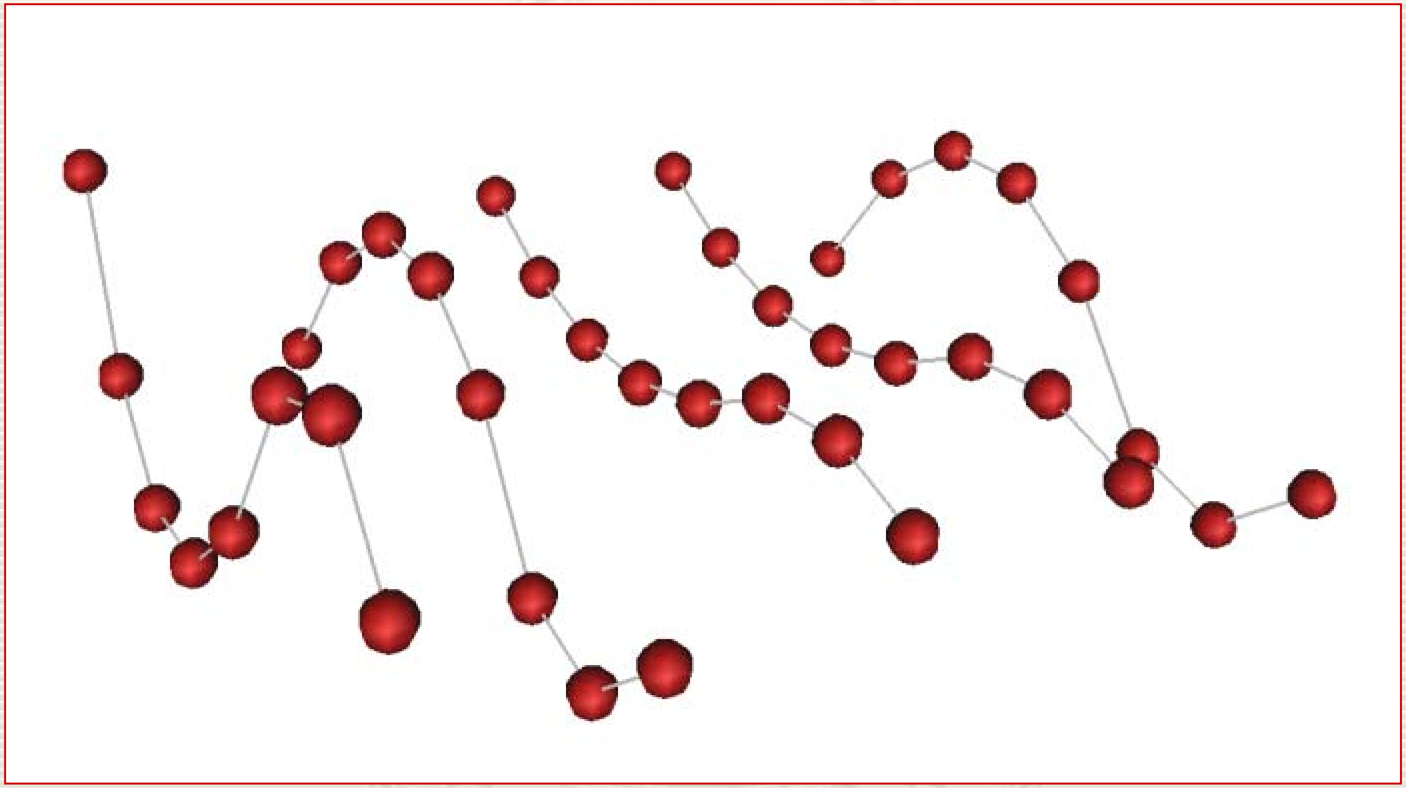
Valutazione approssimata via raffinamento

Patch Spline



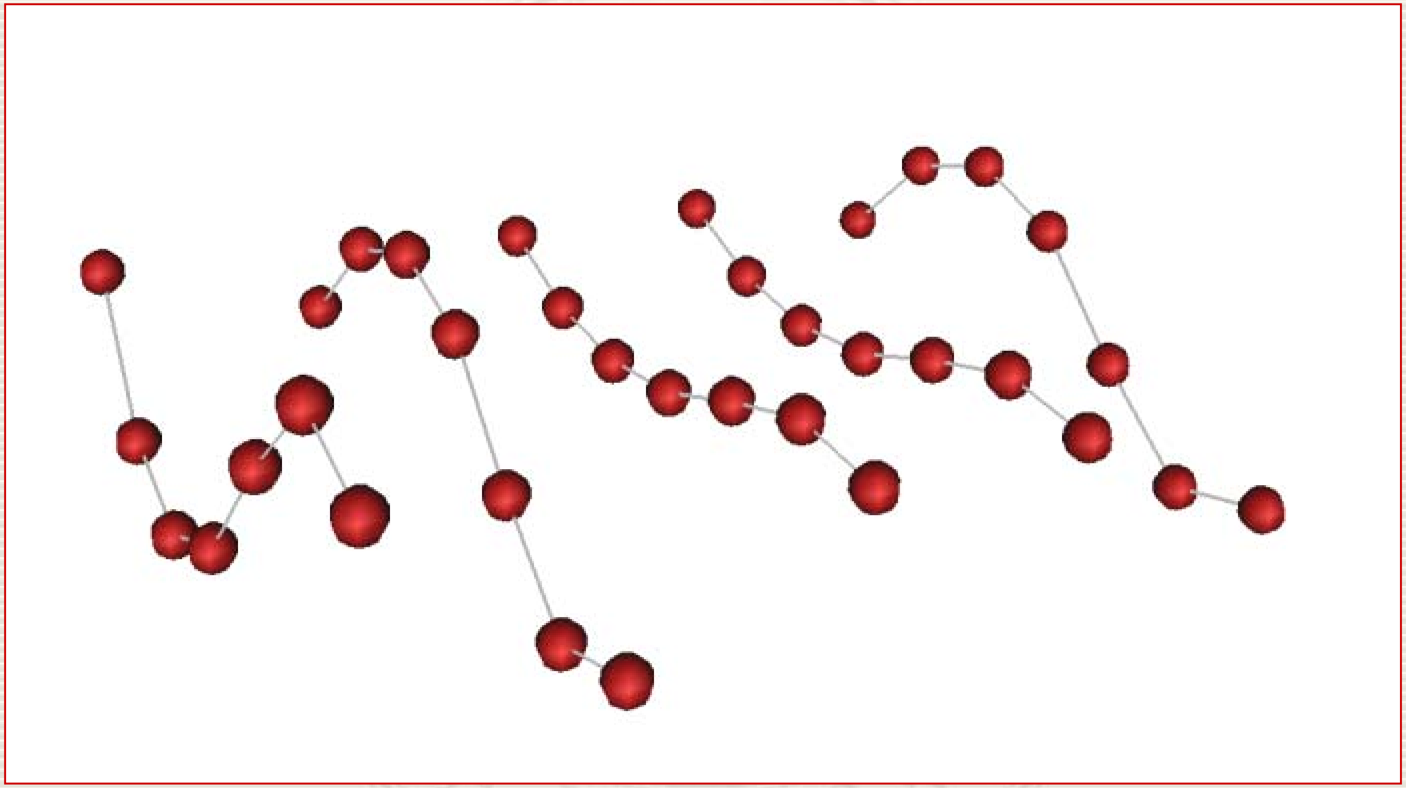
Valutazione approssimata via raffinamento

Patch Spline



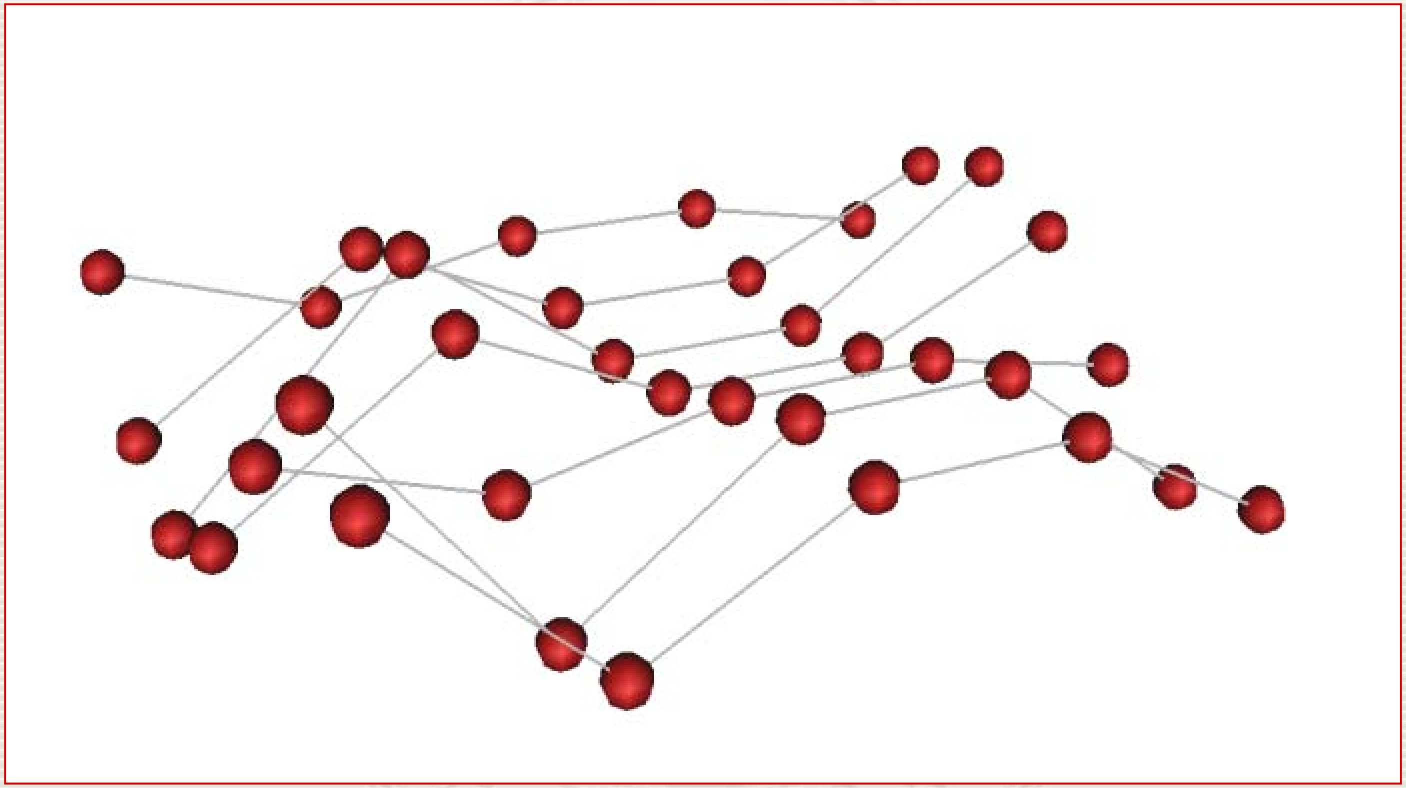
Valutazione approssimata via raffinamento

Patch Spline



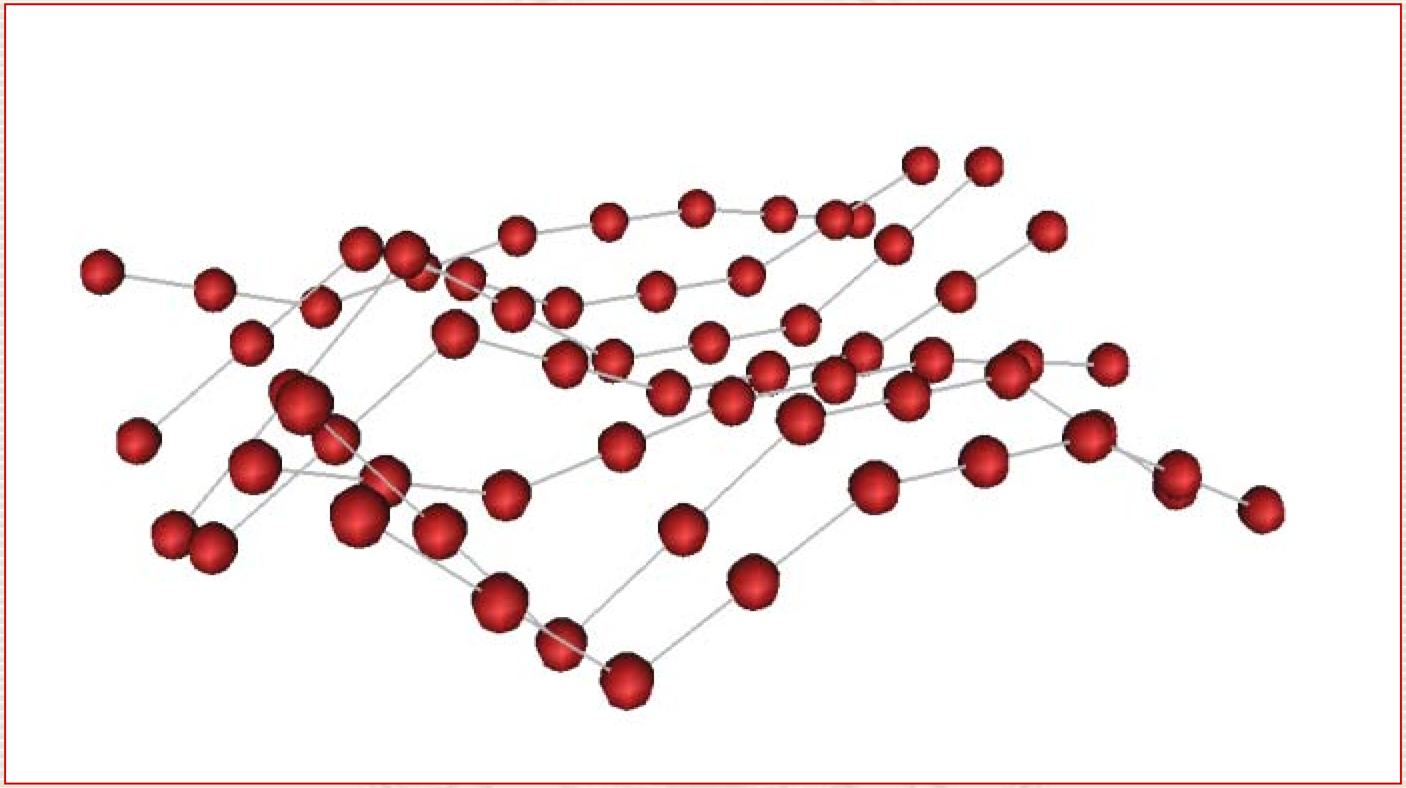
Valutazione approssimata via raffinamento

Patch Spline



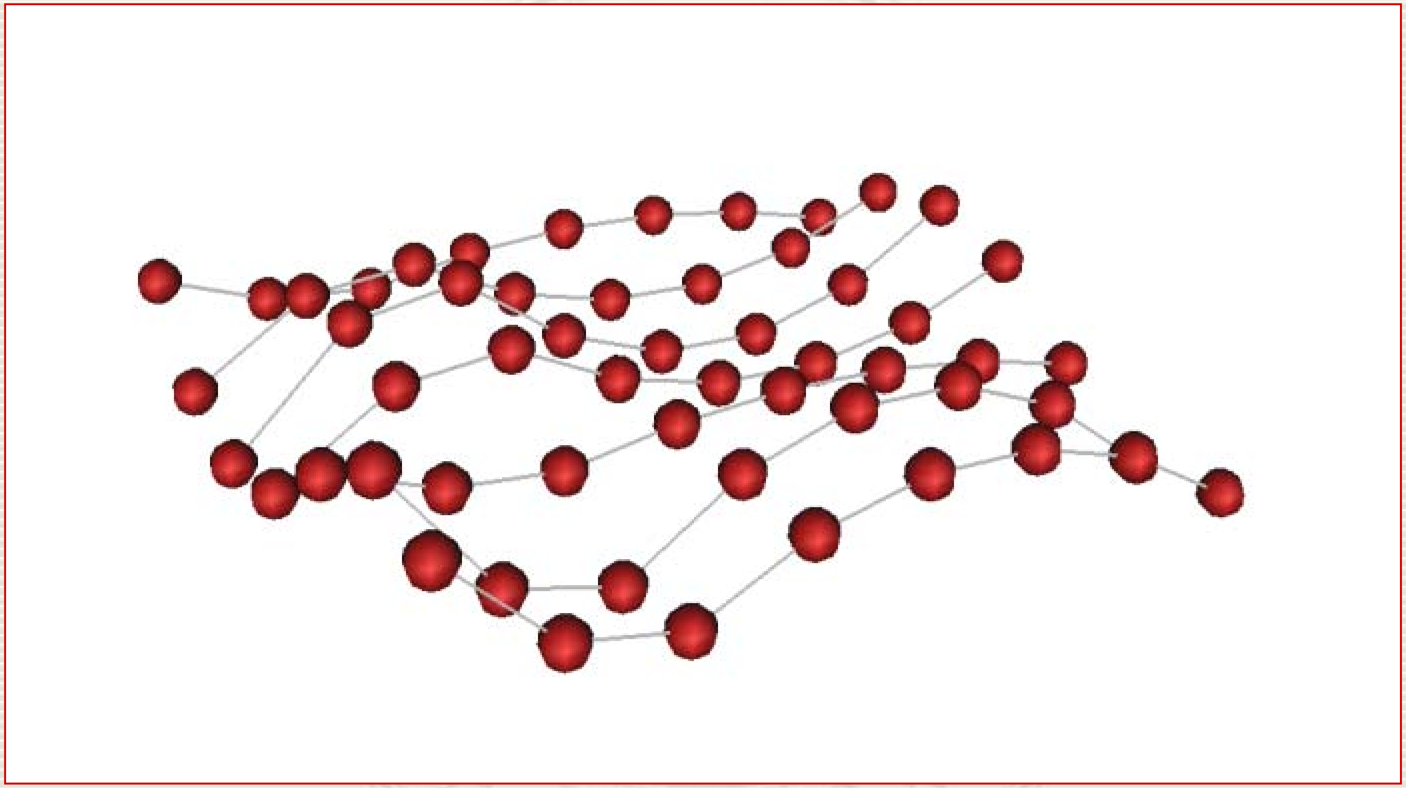
Valutazione approssimata via raffinamento

Patch Spline



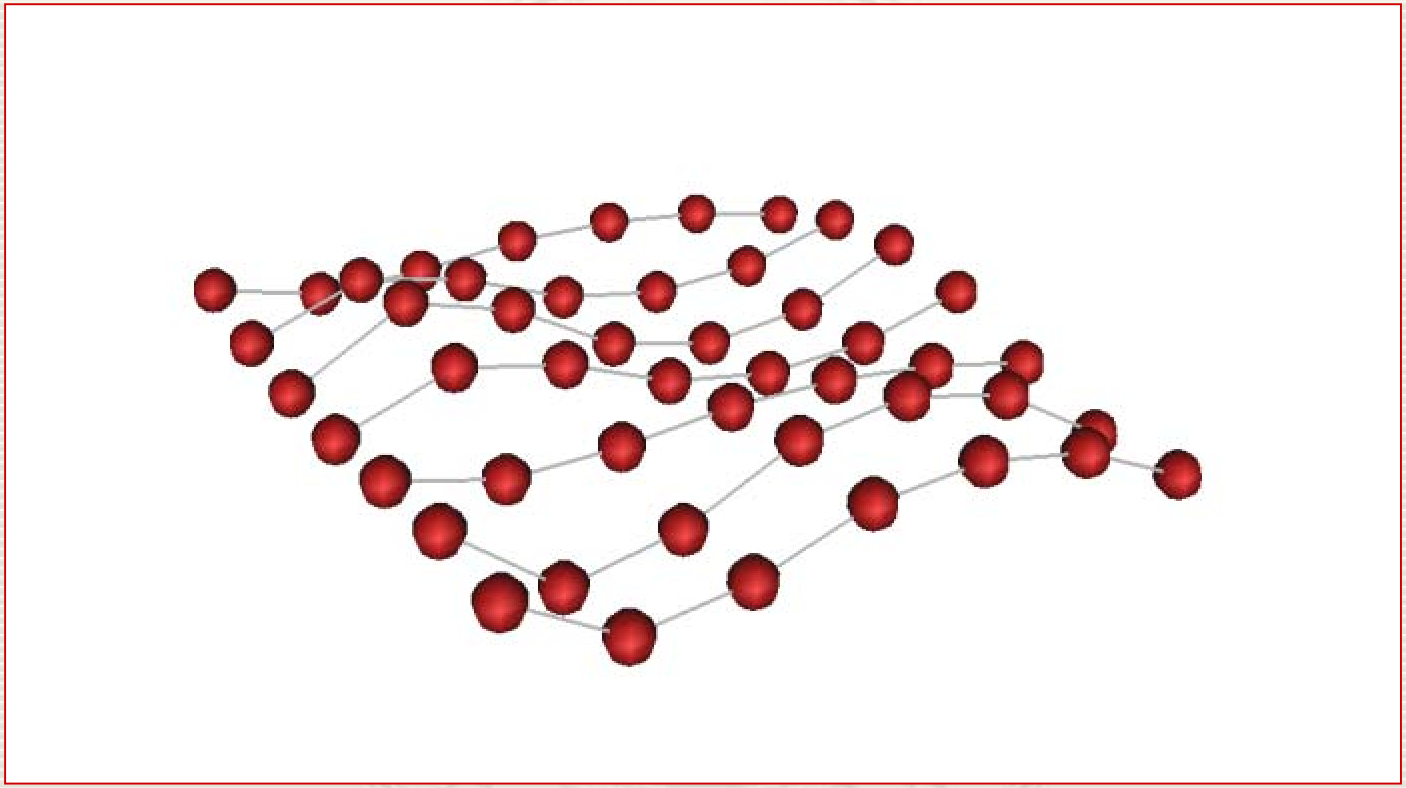
Valutazione approssimata via raffinamento

Patch Spline



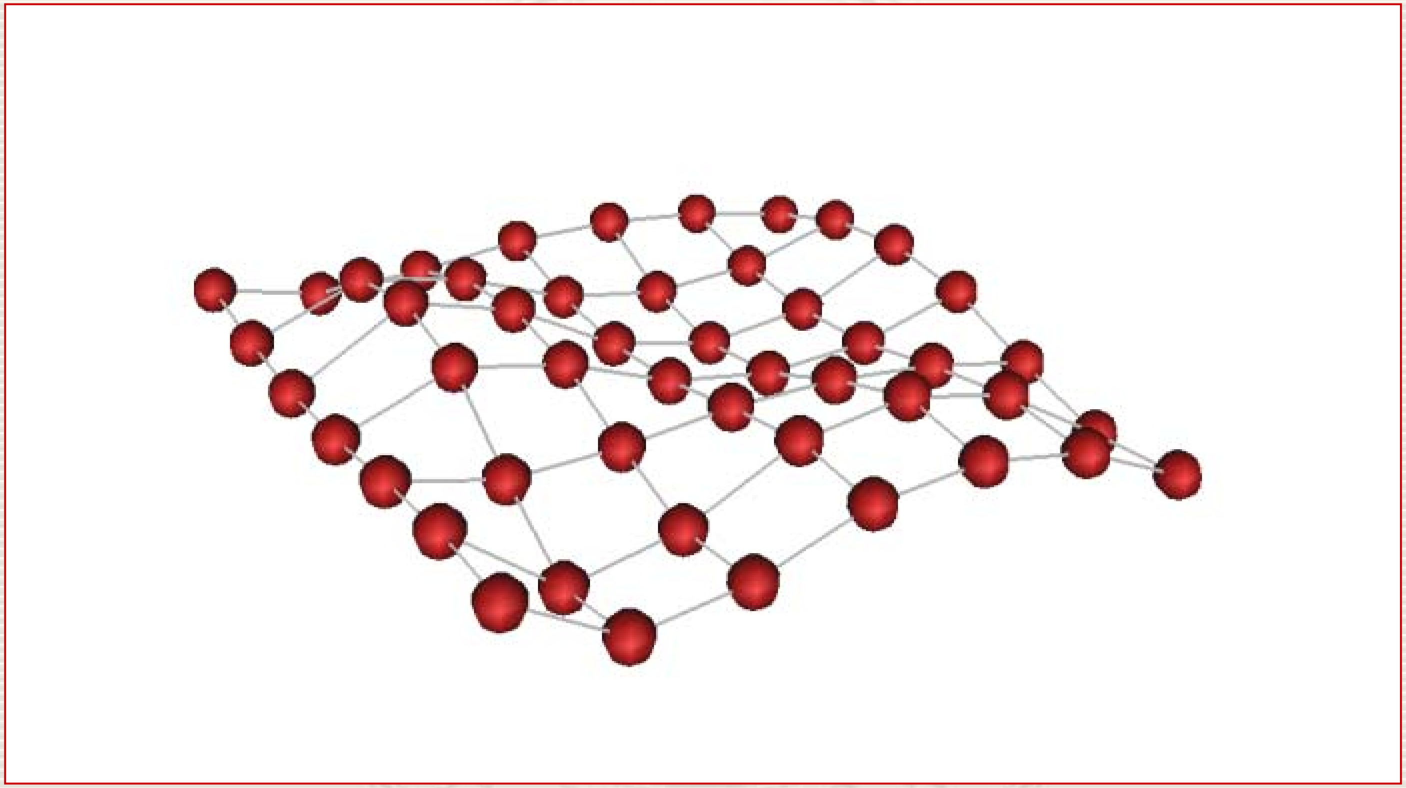
Valutazione approssimata via raffinamento

Patch Spline



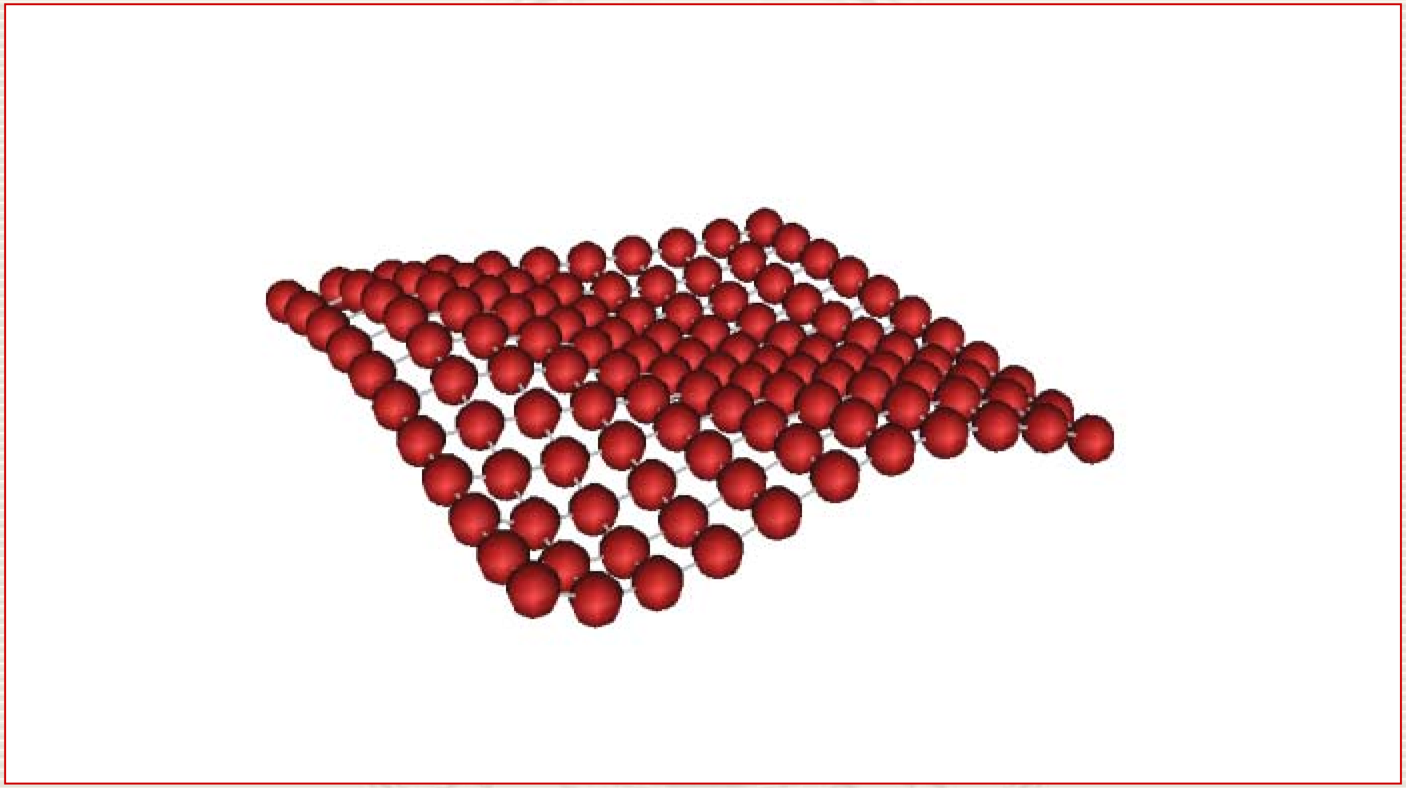
Valutazione approssimata via raffinamento

Patch Spline



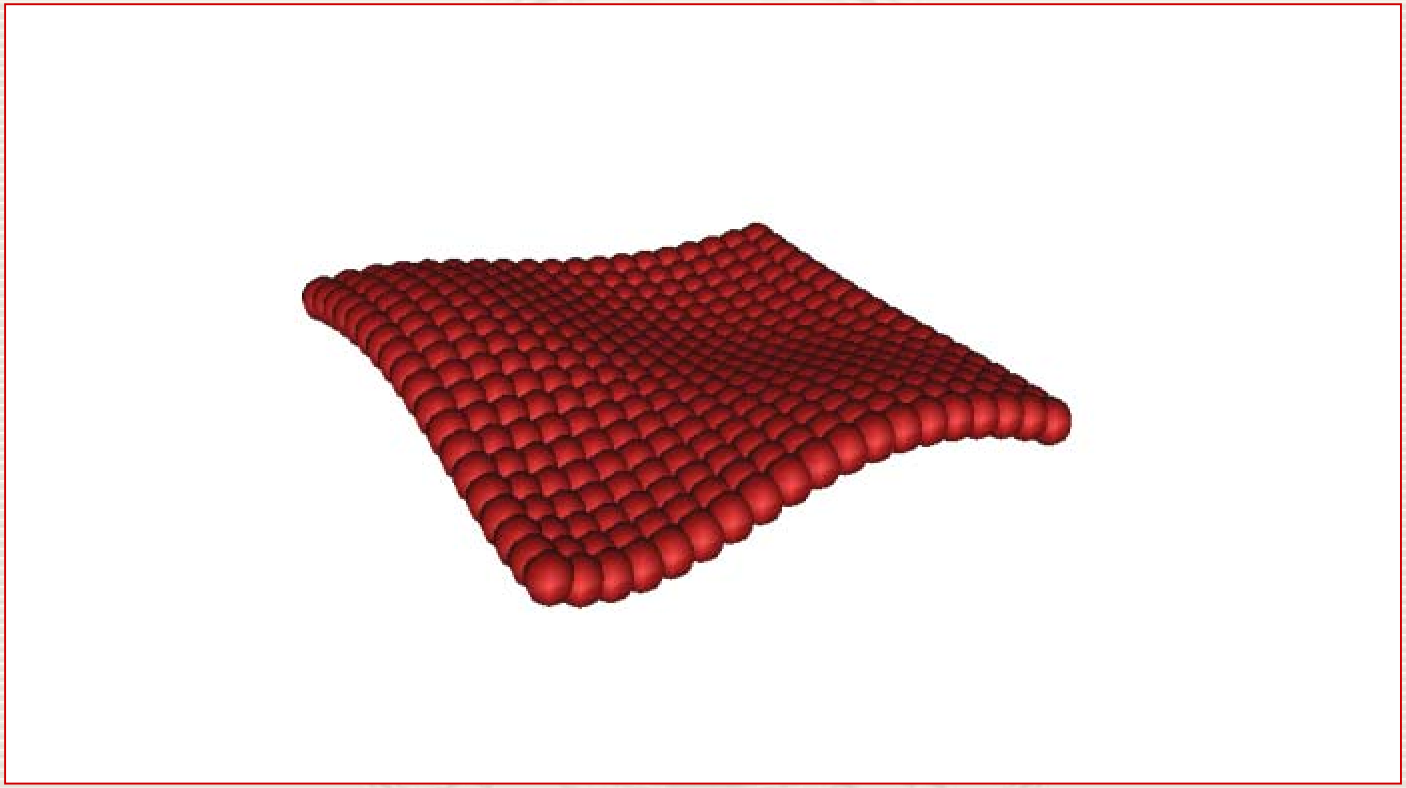
Valutazione approssimata via raffinamento

Patch Spline



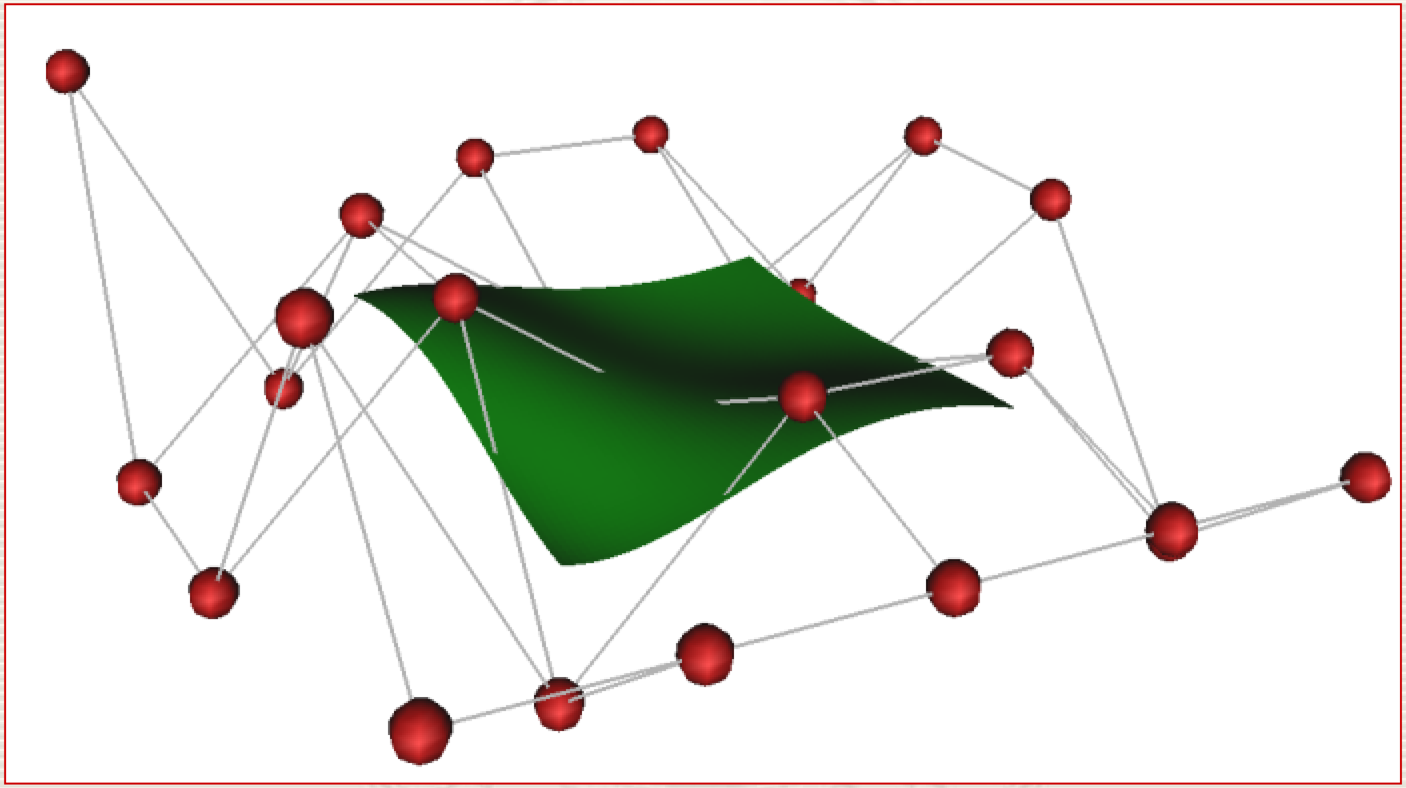
Valutazione approssimata via raffinamento

Patch Spline



Valutazione approssimata via raffinamento

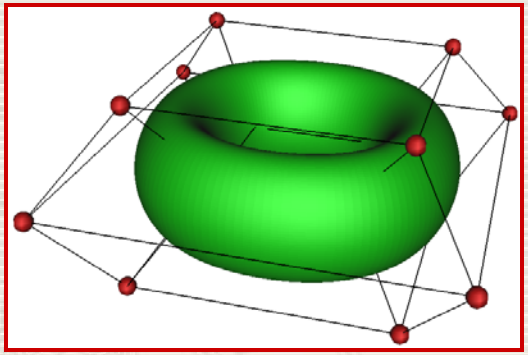
Patch Spline

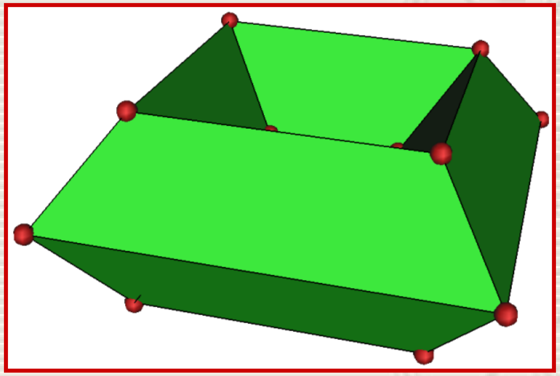


Valutazione approssimata via raffinamento

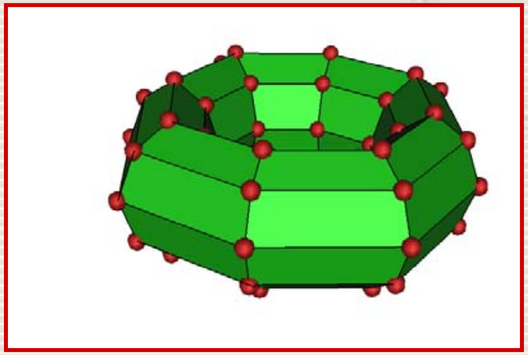
Valutazione Esatta

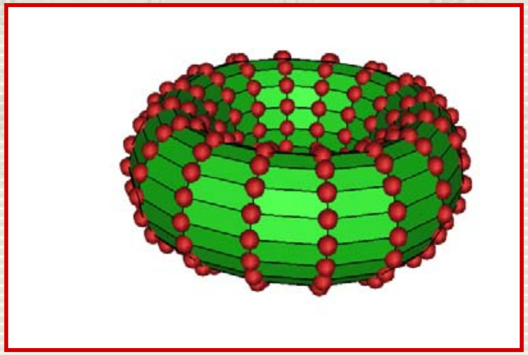
Valutazione Esatta a valori parametrici (u,v) arbitrari

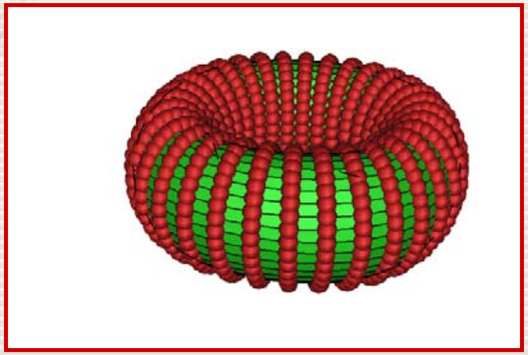




Valutazione approssimata via raffinamenti successivi







Tassellazione

- *La* *Tassellazione* è il processo di considerare una superficie complessa (come un patch bicubico) e approssimarla con un insieme di superfici più semplici (come i triangoli)
- Vogliamo tassellare patch di Bézier, e intere superfici spline/nurbs mediante Valutazione Esatta

Tassellazione Uniforme

- Il modo più diretto di tassellare una superficie in forma parametrica è quello uniforme
- Con questo metodo, si definisce una risoluzione in u e in v, quindi si divide uniformemente il dominio in una griglia, infine si determina una griglia di punti della superficie per Valutazione Esatta
- Questo metodo è molto efficiente, in quanto il costo di valutazione della superficie ha approssimativamente lo stesso costo della valutazione di una curva
- Comunque, poiché la mesh generata è uniforme, può avere più triangoli del necessario in zone piatte e meno del necessario in zone altamente curve

Tassellazione Adattiva

- Molto spesso, l'obiettivo di una tassellazione è quello di determinare il minor numero di triangoli necessari per rappresentare accuratamente la superficie originale
- Per una superficie curva, questo significa che si vogliono più triangoli in certe zone dove la curvatura è alta e meno triangoli in zone dove la curvatura è bassa
- Si vogliono anche più triangoli in zone che sono più vicine alla camera e meno in zone lontane
- *La Tassellazione Adattiva* è definita per venire in contro a queste esigenze

Tassellazione Mista

- Certi software o librerie di rendering usano uno schema di Tassellazione Mista
- Prima, la superficie originale viene suddivisa adattivamente in numerosi sotto-patch, ciascuno approssimativamente della stessa dimensione (circa 10 pixel di lato)
- Poi, ciascuno dei sotto-patch (di tipo rettangolare in u,v) viene tassellato in modo uniforme ad una certa risoluzione (per esempio 10 x 10)
- Il risultato è che la superficie viene tassellata in triangoli approssimatvamente della dimensione di un singolo pixel
- Il maggior costo dell'algoritmo è nella tassellazione uniforme, che può essere implementata in una maniera molto efficiente.

Toro: superficie di Rotazione

Si definisce una curva sezione 2D spline o NURBS; allora la superficie ottenuta ruotando la curva intorno ad un asse può essere rappresentata mediante una superficie NURBS