

Curve e Superfici di Bézier

Carlo Rossanigo

Dicembre 2024

Indice

1	Introduzione	2
1.1	Premessa	2
2	Curve di Bézier	3
2.1	Inviluppo convesso	5
2.2	Molteplici punti di controllo	5
2.3	Vettore Tangente	7
2.4	Concatenazione di due curve di Bézier	7
2.5	Algoritmo di De Casteljau	9
2.6	Splitting delle curve di Bézier	10
2.7	Intersezione fra due curve di Bézier	10
3	Superfici di Bézier	12
3.1	Proprietà delle superfici di Bézier	13
3.1.1	Inviluppo convesso	14
3.1.2	Piano Tangente	14
3.1.3	Unione di due patch di Bézier	14
3.1.4	De Boor-Coox	15

1 Introduzione

Questo lavoro si pone come obiettivo quello di esplorare le curve e le superfici di Bézier presentandone alcune delle loro proprietà fondamentali, adottando un approccio descrittivo piuttosto che rigorosamente formale.

Per una trattazione più approfondita, che include dimostrazioni formali e un'analisi dettagliata, si invita il lettore a consultare il libro di Danaila, Joly, Kaber e Postel (Danaila et al. [2023]). L'argomento risale agli anni 60' del secolo scorso in ambito industriale - Bézier stesso lavorò presso la Renault - e fornisce una introduzione a diverse aree della scienza computazionale moderna, quali per esempio la grafica computazionale, la realtà virtuale e la modellizzazione geometrica¹.

1.1 Premessa

Prima di fornire la definizione di curva di Bézier, è utile ricordare cosa si intenda per curva.

Def 1.1 (Curva): Sia Ω un aperto di \mathbb{R}^n . Si chiama curva una qualsiasi mappa continua $\varphi : [a, b] \subseteq \mathbb{R} \rightarrow \Omega$.

Spesso si confonde volutamente la curva con la sua immagine, denotata con $\gamma = \varphi([a, b])$, detta anche *sostegno* della curva.

Una curva può presentare alcune proprietà, per esempio può essere:

- **Chiusa:** se gli estremi coincidono, cioè $\varphi(a) = \varphi(b)$.
- **Semplice:** se non si autointerseca, cioè φ è iniettiva su $[a, b]$ e su $(a, b]$.
- **Regolare:** se φ è di classe C^1 sul suo dominio $[a, b]$ e la sua derivata non si annulla, cioè $\varphi'(t) \neq 0$ per ogni $t \in [a, b]$. Questo ci permette di definire un vettore tangente e, di conseguenza, un versore tangente $\tau = \frac{\varphi'(t)}{\|\varphi'(t)\|}$.

Ricordiamo inoltre che si può parlare anche di lunghezza di una curva:

Def 1.2 (Lunghezza di una curva): Data φ una curva di classe C^1 su $[a, b]$, la lunghezza L della curva φ è data da:

$$L = \int_a^b |\varphi'(t)| dt,$$

che naturalmente coincide con la definizione alternativa della lunghezza, ossia:

$$\ell(\gamma) := \sup \{ \text{lunghezza delle spezzate con vertici in } \gamma \}.$$

In maniera analoga, possiamo fornire una definizione operativa di superficie.

Def 1.3 (Superficie): Dato A aperto di \mathbb{R}^2 , definiamo una superficie come immagine di una funzione differenziabile e iniettiva di due variabili reali nello spazio \mathbb{R}^3 :

¹Per gli amanti dei giochi Nintendo, i tracciati di Mario Kart sono costruiti mediante le curve di Bezier.

$$\varphi : A \rightarrow \mathbb{R}^3$$

Per ottenere una superficie liscia, è necessario che φ sia una immersione: il differenziale $d\varphi_x$ di φ deve essere iniettivo in ogni punto x .

2 Curve di Bézier

Detto ciò, possiamo ora fornire la definizione di curva di Bézier.

Def 2.1 (Curva di Bézier): Siano $n \geq 2$ e $m > 0$ due interi, e consideriamo $t \in I = [0, 1]$ un parametro. Presi $m + 1$ punti P_0, P_1, \dots, P_m , detti punti di controllo, definiamo il punto $P(t)$ come combinazione lineare dei polinomi di Bernstein:

$$P(t) = \sum_{k=0}^m B_m^k(t) P_k$$

dove $B_m^k(t)$ è il polinomio di Bernstein di grado m , dato dalla formula:

$$B_m^k(t) = \binom{m}{k} t^k (1-t)^{m-k}$$

La curva di Bézier, indicata con \mathbf{B}_m , è la traiettoria del punto $P(t)$ al variare del parametro t nell'intervallo unitario.

Si noti che nella definizione del punto $P(t)$ compaiono come pesi i polinomi di Bernstein: tali polinomi svolgono un ruolo importante nell'ambito dell'analisi numerica, in quanto formano una base per lo spazio dei polinomi di grado m a coefficienti in $[0, 1]$, $P_m([0, 1])$. In particolare, i polinomi di Bernstein presentano le seguenti proprietà:

- Sono non negativi:

$$\forall t \in [0, 1], \quad 0 \leq B_m^k(t) \leq 1$$

- Sono a somma unitaria:

$$\sum_{k=0}^m B_m^k(t) = 1 \quad \forall t \in [0, 1].$$

- Per $t = 0$, solo il termine $B_m^0(0)$ è diverso da zero, mentre per $t = 1$, solo il termine $B_m^m(1)$ è diverso da zero.

Dalla definizione del punto $P(t)$ (2.1) e dalle proprietà elencate dei polinomi di Bernstein — in particolare l'ultima — possiamo notare che esistono due punti di controllo che appartengono sicuramente alla curva di Bézier: quello iniziale P_0 e quello finale P_m ; infatti si ha $P(0) = P_0$ e $P(1) = P_m$. In generale, gli altri punti di controllo non appartengono alla curva di Bézier.

Inoltre, dalle proprietà dei polinomi di Bernstein, ne discende che due curve di Bézier con lo stesso numero di punti di controllo coincidono se e solo se presentano gli stessi punti di controllo, nello stesso ordine: cambiando ad esempio l'ordine anche di soli due punti si genera una curva completamente diversa.

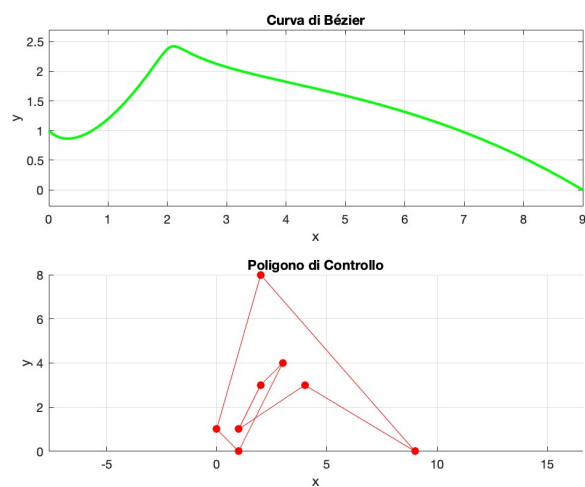


Figura 1: Curva di Bézier generata dai 8 punti di controllo P_0, P_1, \dots, P_8 .

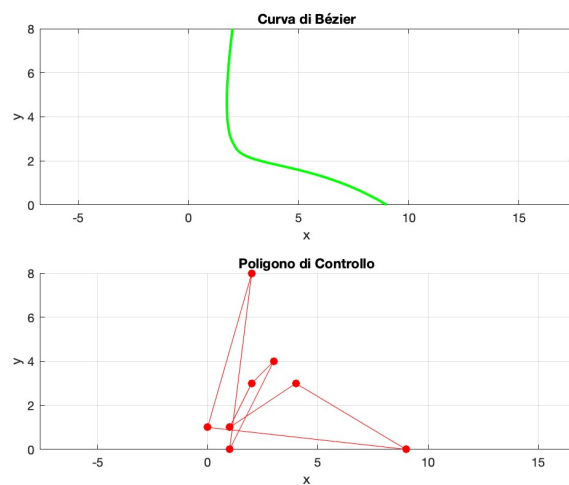


Figura 2: Curva di Bézier generata dagli stessi punti di controllo, ma scambiando il primo con l'ultimo punto .

Tuttavia, è bene sottolineare che si può generare la stessa curva di Bézier considerando un numero diverso di punti di controllo. Ad esempio, considerando due punti P_0 e P_1 , posso definire il segmento P_0P_1 come una curva di Bézier di grado 1:

$$P(t) = \sum_{k=0}^1 B_1^k(t) P_k$$

Se invece considero il punto medio di questo segmento, posso ottenere la stessa curva di Bézier considerando 3 punti: $Q_0 = P_0$, $Q_1 = \frac{P_0+P_1}{2}$, $Q_2 = P_1$. Questo definisce una curva di Bézier di grado 2:

$$Q(t) = \sum_{k=0}^2 B_2^k(t) Q_k$$

Detto ciò, possiamo enunciare alcune proprietà delle curve di Bézier:

2.1 Inviluppo convesso

Prima di discutere dell'inviluppo convesso della nostra curva di Bézier, ricordiamo cos'è l'inviluppo convesso:

Def 2.2 (Inviluppo Convesso): Dato un insieme A , l'inviluppo convesso di A , denotato con C_A , è il più piccolo insieme convesso che contiene A :

$$C_A = \bigcap \{C \mid C \text{ è convesso e } A \subseteq C\}$$

Ora, dalla definizione (2.1), possiamo notare che il punto $P(t)$ non è nient'altro che il baricentro di $m+1$ punti con i rispettivi pesi, che sono i polinomi di Bernstein. Ne segue quindi che la sua traiettoria, cioè la curva di Bézier, giace interamente nell'inviluppo convesso generato da questi punti. In particolare, tale inviluppo convesso contiene anche il poligono di controllo, cioè il poligono generato dai punti di controllo. Questo poligono può essere convesso e coincidere con l'inviluppo convesso, ma in generale non lo è.

2.2 Molteplici punti di controllo

Scegliendo punti di controllo in maniera casuale oppure ripetendo gli stessi punti di controllo si possono costruire curve di Bézier più o meno complesse.

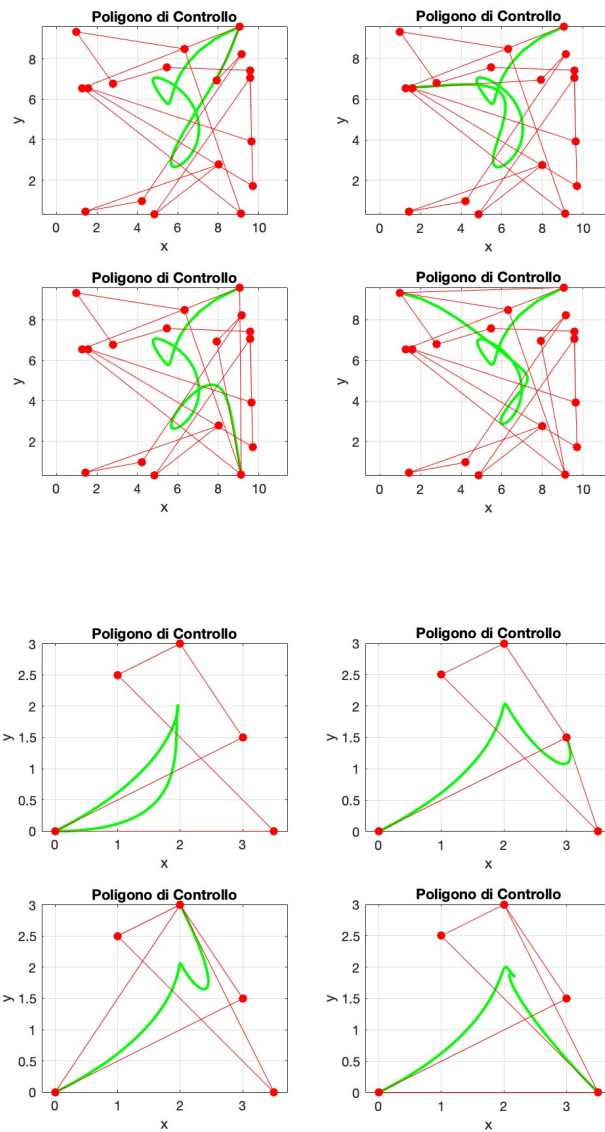


Figura 3: Esempi di Curve di Bézier in cui si ripetono i punti di controllo

In particolare, si noti che basta scegliere un punto finale uguale a quello iniziale per ottenere una curva di Bézier chiusa.

2.3 Vettore Tangente

Per definizione, le curve di Bézier sono curve di classe C^1 , il che ci permette di definire un vettore tangente, dato dalla derivata rispetto al parametro t :

$$\tau(t) = \frac{d}{dt}P(t) = \sum_{k=0}^m \frac{d}{dt}B_m^k(t)P_k$$

Ora, da un punto di vista analitico, la derivata del polinomio di Bernstein è:

$$\frac{d}{dt}B_m^k(t) = \begin{cases} -m(1-t)^{m-1} & \text{per } k=0, \\ m(1-t)^{m-2}(1-mt) & \text{per } k=1, \\ \binom{m}{k}(k-mt)t^{k-1}(1-t)^{m-k-1} & \text{per } 1 < k < m-1, \\ mt^{m-2}(m-1-mt) & \text{per } k=m-1, \\ mt^{m-1} & \text{per } k=m. \end{cases}$$

In particolare, ne segue che quando $P_0 \neq P_1$, la derivata in P_0 è:

$$\tau(0) = mP_0P_1,$$

cioè la curva è tangente in P_0 al lato P_0P_1 del poligono di controllo. Analogamente, quando $P_{m-1} \neq P_m$, la derivata in P_m è:

$$\tau(1) = mP_{m-1}P_m,$$

cioè la curva è tangente in P_m al lato $P_{m-1}P_m$ del poligono di controllo.

Oss 2.1: Esiste una espressione molto utile per calcolare il vettore tangente:

$$\frac{d}{dt}B_m^k(t) = m(B_{m-1}^{k-1}(t) - B_{m-1}^k(t))$$

2.4 Concatenazione di due curve di Bézier

Le osservazioni fatte nella sezione precedente sono molto utili quando si vuole trattare la concatenazione di due curve di Bézier: consideriamo infatti due curve di Bézier distinte \mathbf{B}_m e \mathbf{B}'_m con lo stesso numero di punti di controllo, rispettivamente P_0, P_1, \dots, P_m e P'_0, P'_1, \dots, P'_m . Possiamo imporre tre tipi di concatenazione: una connessione con continuità C^0 , una connessione tangenziale G^1 , e una di classe C^1 — la più forte delle tre -.

Per la connessione C^0 , basta imporre che l'ultimo punto della prima curva sia uguale al primo punto della seconda curva. Ciò si traduce in:

$$P_m = P'_0.$$

Per la connessione tangenziale, che implica la continuità di classe C^0 , poichè la prima curva \mathbf{B}_m è tangente nel punto P_m al segmento $mP_{m-1}P_m$ e la seconda curva \mathbf{B}'_m è tangente nel punto P'_0 al segmento $mP'_0P'_1$ imponiamo che i punti $P_{m-1}, P_m(=P'_0), P'_1$ siano allineati. Questo ci fornisce già un buon layout, ma è

una connessione meno forte di quella C^1 , dove invece, denotati con τ e τ' i due vettori tangenti alle due curve di Bézier, imporremo che:

$$\tau(1) = \tau'(0),$$

cioè:

$$mP_mP_{m-1} = mP'_0P'_1,$$

ossia $P'_0 = P_m$ è il punto medio del segmento $P_{m-1}P'_1$: $P'_0 = \frac{P_{m-1}P'_1}{2}$.

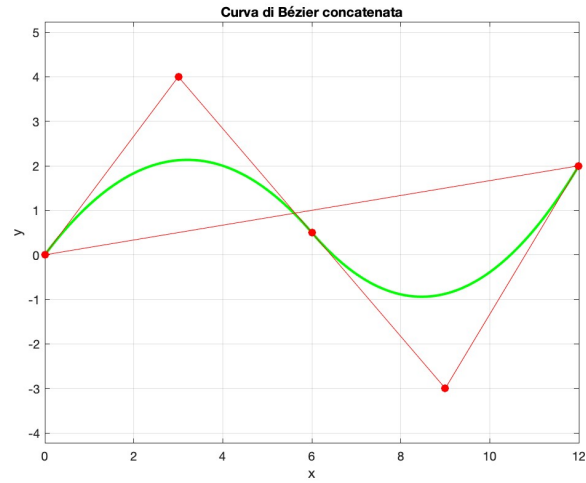
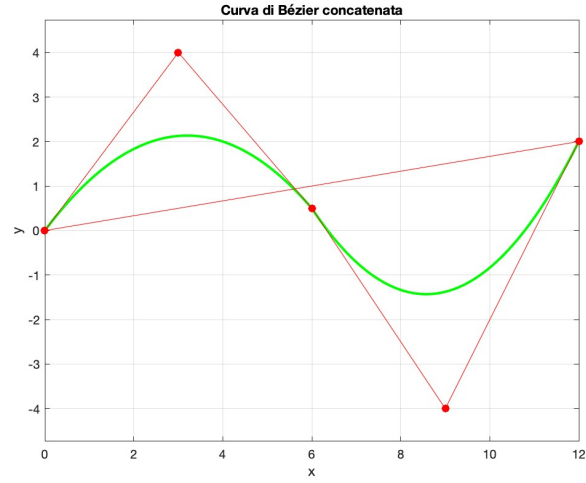


Figura 4: Connessione G^1 in alto e C^1 in basso: nel secondo caso P'_0 è punto medio del segmento $P_1P'_1$

2.5 Algoritmo di De Casteljau

Osservando la definizione di curva di Bézier (2.1), ci rendiamo conto che implementare il calcolo della curva utilizzando tale definizione risulta scomodo. Infatti, intervengono polinomi di grado m , che può essere un numero molto grande, e i coefficienti binomiali: questo si traduce in problemi computazionali dal punto di vista temporale per il calcolatore e in problemi di stabilità numerica, poiché il polinomio che scriviamo sulla carta non corrisponde sempre a quello che otteniamo numericamente.

Per ovviare a tali problemi, De Casteljau ideò un algoritmo che semplifica i calcoli basandosi sulla seguente relazione ricorsiva dei polinomi di Bernstein:

$$B_{m+1}^k(t) = tB_m^{k-1}(t) + (1-t)B_m^k(t)$$

L'idea alla base dell'algoritmo è quella di costruire dei segmenti tra i punti medi dei lati e di iterare tale processo per ottenere la curva di Bézier. In particolare, vogliamo generare i punti P_q^p per $p = 0, 1, \dots, m$ e $q = p, p+1, \dots, m$.

Il procedimento funziona nel seguente modo:

- Per $p = 0$, allora $P_q^0 = P_q$, ovvero generiamo i punti di controllo della curva.
- Per ogni $p = 1, 2, \dots, m$, si calcola P_q^p usando la formula:

$$P_q^p = (1-t)P_{q-1}^{p-1} + tP_q^{p-1},$$

dove t è il parametro che varia nell'intervallo $[0, 1]$.

Facciamo un esempio con $m = 2$ punti di controllo P_0, P_1, P_2 : il calcolo di $P(t)$ mediante (2.1) è

$$\begin{aligned} P(t) &= \sum_{k=0}^2 B_2^k(t) P_k \\ &= (1-t)^2 P_0 + 2t(1-t)P_1 + t^2 P_2 \end{aligned}$$

utilizzando l'algoritmo di De Casteljau effettueremo i seguenti passaggi:

- Per $p = 0$, imponiamo: $P_0^0 = P_0, P_1^0 = P_1, P_2^0 = P_2$,
- Per $p = 1$, calcoleremo

$$P_1^1 = (1-t)P_0^0 + tP_1^0, P_2^1 = (1-t)P_1^0 + tP_2^0$$

- Infine per l'ultimo passo, $p = 2$, otteniamo

$$\begin{aligned} P_2^2 &= (1-t)P_1^1 + tP_2^1 \\ &= (1-t)^2 P_0 + 2t(1-t)P_1 + t^2 P_2 \end{aligned}$$

Questo algoritmo ha un costo computazionale inferiore rispetto al calcolo diretto tramite la definizione, in quanto richiede per generare il punto $P(t)$ solo $m + m - 1 + \dots + 1 = m(m + 1)/2$ combinazioni lineari, rendendolo molto più efficiente rispetto al calcolo diretto mediante definizione con i polinomi di Bernstein.

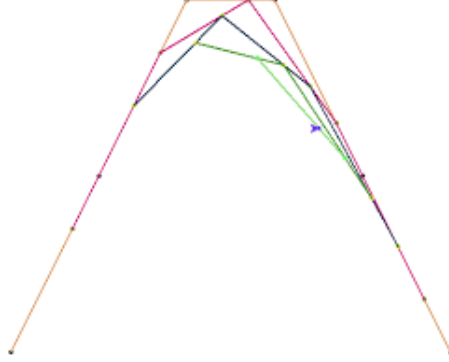


Figura 5: Algoritmo di De Casteljau

2.6 Splitting delle curve di Bézier

L'algoritmo di De Casteljau è utile per risolvere il problema della suddivisione di una curva di Bézier. Consideriamo una curva di Bézier \mathbf{B}_m definita da $m + 1$ punti di controllo P_0, P_1, \dots, P_m e un parametro $\theta \in I = [0, 1]$; sia $P(\theta)$ il punto di \mathbf{B}_m associato a θ mediante l'algoritmo di De Casteljau.

Per suddividere la curva, generiamo i punti P_q^p per $p = 0, 1, \dots, m$ e $q = p, p + 1, \dots, m$ utilizzando l'algoritmo di De Casteljau e poniamo infine $P(\theta) = P_m^m$. Considerando i punti $P_0^0, P_1^1, \dots, P_m^m$ generati dall'algoritmo, costruiamo una seconda curva di Bézier \mathbf{B}'_m , traiettoria del punto $P'(t')$, definito come segue:

$$P'(t') = \sum_{k=0}^m B_m^k(t') P_k^k$$

dove $B_m^k(t')$ sono i polinomi di Bernstein per la nuova curva. In particolare, non solo si ha che $P'(0) = P_0^0 = P_0$ e $P'(1) = P_m^m = P(\theta)$, ma la curva di Bézier generata dalla traiettoria di questo punto per $t' \in I$ è contenuta nella curva di Bézier originale generata dai punti di controllo P_0, P_1, \dots, P_m , e coincide con quest'ultima per i valori di $t \in [0, \theta]$.

2.7 Intersezione fra due curve di Bézier

Siano \mathbf{B}_m e $\mathbf{B}'_{m'}$ due curve di Bézier definite rispettivamente da $m + 1$ punti di controllo P_0, P_1, \dots, P_m e $m' + 1$ punti di controllo $P'_0, P'_1, \dots, P'_{m'}$. Le espressioni delle due curve sono:

$$P(t) = \sum_{k=0}^m B_m^k(t) P_k$$

$$P'(t') = \sum_{k'=0}^{m'} B_{m'}^{k'}(t') P_{k'}$$

Vogliamo determinare - se esistono - due valori t e t' tali che $P(t) = P'(t')$. A livello teorico, vi sono due strade per affrontare il problema: la prima consiste nel dedurre le rappresentazioni implicite delle due curve, $f(x, y)$ e $g(x, y)$, e risolvere il sistema di equazioni implicite per trovare le radici di un polinomio di grado $m + m'$; tuttavia, questo approccio risulta scomodo dal punto di vista computazionale.

Una soluzione alternativa, più pratica, sfrutta gli inviluppi convessi: poiché le due curve di Bézier sono contenute nei rispettivi inviluppi convessi, possiamo concludere che se tali inviluppi non si intersecano allora le curve di Bézier non si intersecano. Al contrario, se gli inviluppi convessi si intersecano, le curve di Bézier potrebbero avere uno o più punti di intersezione.

Pertanto, possiamo procedere con il seguente algoritmo iterativo:

- 1) Finché gli inviluppi convessi si intersecano, suddividiamo le due curve di Bézier nei rispettivi valori medi dei parametri.
- 2) Associamo i nuovi inviluppi convessi alle curve suddivise.
- 3) Continuare l'iterazione fino a quando gli inviluppi convessi non si intersecano più, oppure viene trovata un'intersezione che converge a un punto, il quale sarà un punto di intersezione tra le due curve.

A livello pratico, l'algoritmo può essere implementato sostituendo gli inviluppi convessi con i rettangoli definiti dalle coordinate dei punti di controllo "estremi". Fissato un valore di tolleranza $\sigma > 0$, iteriamo su tali rettangoli finché la loro altezza o base non scende sotto σ ; a questo punto, l'intersezione S viene definita come il punto di intersezione delle diagonali M_1M_2 e $M'_1M'_2$ del rettangolo. I valori t e t' si ottengono tramite un'approssimazione lineare:

$$t \approx t(M_2) \frac{x(S) - x(M_1)}{x(M_2) - x(M_1)} + t(M_1) \frac{x(M_2) - x(S)}{x(M_2) - x(M_1)}$$

e

$$t' \approx t'(M'_2) \frac{x(S) - x(M'_1)}{x(M'_2) - x(M'_1)} + t'(M'_1) \frac{x(M'_2) - x(S)}{x(M'_2) - x(M'_1)}$$

Oss 2.2: se $x(M_1) = x(M_2)$, useremo le ordinate $y(M_1), y(M_2)$.

Intersezione di due curve : vista globale

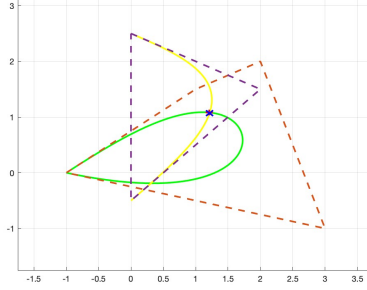


Figura 6: Intersezione trovata dall'algoritmo

Intersezione di due curve: algoritmo

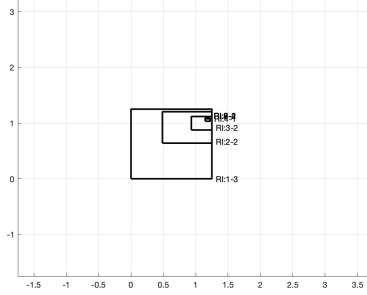


Figura 7: Rettangoli generati dall'algoritmo

3 Superfici di Bézier

Il discorso fatto sulle curve di Bézier si estende alle superfici di Bézier tramite prodotto tensoriale:

Def 3.1 (Superficie di Bézier): Siano $m_1 > 0$ e $m_2 > 0$ due interi, consideriamo un insieme di $(m_1 + 1) \times (m_2 + 1)$ punti di controllo $P_{k_1, k_2} \in \mathbb{R}^3$. Definiamo il punto $P(t_1, t_2)$ mediante la relazione:

$$P(t_1, t_2) = \sum_{k_1=0}^{m_1} \sum_{k_2=0}^{m_2} B_{m_1}^{k_1}(t_1) B_{m_2}^{k_2}(t_2) P_{k_1, k_2}$$

dove $B_{m_1}^{k_1}(t_1)$ e $B_{m_2}^{k_2}(t_2)$ sono i polinomi di Bernstein di parametri m_1, k_1 e m_2, k_2 per le variabili parametriche $t_1, t_2 \in [0, 1] \times [0, 1]$.

Una patch di Bézier è quindi la traiettoria del punto $P(t_1, t_2)$ al variare di t_1, t_2 nel quadrato unitario $[0, 1] \times [0, 1]$; una superficie di Bézier non è nient'altro che l'unione di tali patches.

Si noti che questa definizione è formulata utilizzando le *patch rettangolari*, ma potrebbe essere data una definizione equivalente utilizzando *patch triangolari*, che sono comunemente usate in altre applicazioni, come la modellazione di superfici più complesse; in questo definiremo il punto $P(t_1, t_2, t_3)$ come

$$P(t_1, t_2, t_3) = \sum_{k_1+k_2+k_3=m} \frac{m!}{k_1!k_2!k_3!} t_1^{k_1} t_2^{k_2} t_3^{k_3} P_{k_1, k_2, k_3}.$$

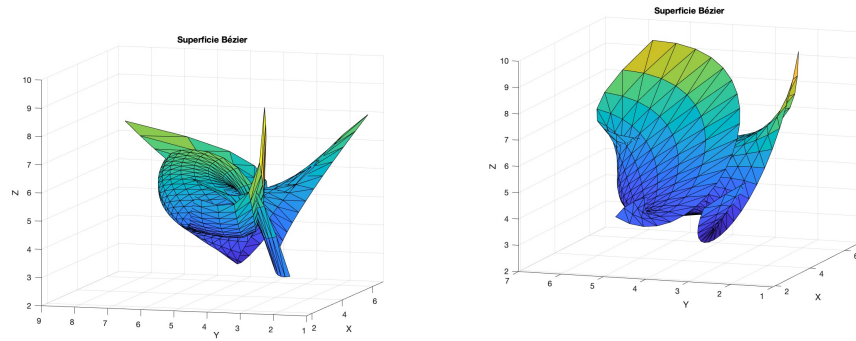


Figura 8: Esempi di Superficie di Bézier

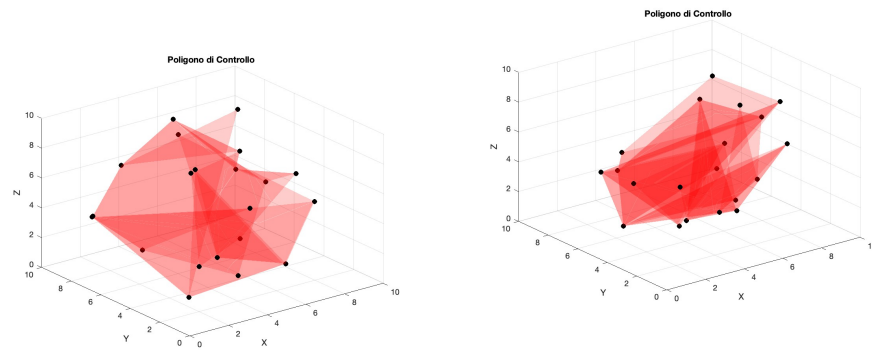


Figura 9: Rispettivi poligoni di controllo

3.1 Proprietà delle superfici di Bézier

Analizziamo alcune delle proprietà principali delle superfici di Bézier.

3.1.1 Inviluppo convesso

Analogamente alle curve di Bézier, la superficie di Bézier giace interamente nell'inviluppo convesso dei punti di controllo. In altre parole, la superficie generata dai punti di controllo è contenuta nell'inviluppo convesso definito dai vertici del poligono formato dai punti di controllo. Questo significa che la superficie non può mai uscire da un volume convesso che contiene tutti i punti di controllo.

3.1.2 Piano Tangente

In modo simile a quanto accade per le curve di Bézier, è possibile definire il piano tangente alla superficie di Bézier in un punto $P(t_1, t_2)$ della superficie. Questo piano tangente può essere ottenuto calcolando le derivate parziali della superficie rispetto ai parametri t_1 e t_2 , ossia:

$$\tau(t) = \left(\frac{d}{dt_1} P(t_1, t_2), \frac{d}{dt_2} P(t_1, t_2) \right)$$

Pertanto, se $P_{0,0} \neq P_{0,1}$, la superficie di Bézier è tangente al triangolo $P_{1,0}, P_{0,0}, P_{0,1}$ in $P_{0,0}$. Proprietà analoga vale per i vertici $P_{m_1,0}, P_{m_1,m_2}, P_{0,m_2}$.

3.1.3 Unione di due patch di Bézier

Sia S_{m_1,m_2} una patch di Bézier definita da $(m_1 + 1) \times (m_2 + 1)$ punti di controllo P_{k_1,k_2} , dove $1 \leq k_1 \leq m_1$ e $1 \leq k_2 \leq m_2$, e $S_{m'_1,m'_2}$ una patch di Bézier definita da $(m'_1 + 1) \times (m'_2 + 1)$ punti di controllo $P_{k'_1,k'_2}$, dove $1 \leq k'_1 \leq m'_1$ e $1 \leq k'_2 \leq m'_2$. Il problema che affrontiamo ora è come connettere queste due patch, con i tre tipi di connessione gi menzionati per le curve di Bézier. Per la connessione C^0 , supponendo che esista una curva comune sui bordi delle due patch, tale curva corrisponde ad un estremo del parametro t_1 o t_2 (ossia $t_1 = 0$ o $t_1 = 1$, oppure $t_2 = 0$ o $t_2 = 1$). La continuità C^0 è soddisfatta quando le due curve di Bézier sui bordi delle patch coincidono, ovvero i punti di controllo sulle due patch devono essere identici sui bordi comuni.

Per ottenere una connessione G^1 , è necessario che i punti di controllo sui bordi delle due patch siano allineati, ossia $P_{m_1,k_2} = P'_{0,k'_2}$ per $k'_2 = 0, 1, \dots, m'_2$. In questo caso, la continuità G^1 si ottiene quando i vettori tangenti $\tau_1(1, t_2) = \frac{d}{dt_1} P(1, t_2)$ e $\tau'_1(0, t'_2) = \frac{d}{dt'_1} P'(0, t'_2)$ sono collineari in ogni punto comune tra le due patch. Questa condizione implica che il punto P_{m_1,k_2} deve essere il punto medio tra i punti P_{m_1-1,k_2} e P'_{1,k_2} per $k_2 = 0, 1, \dots, m_2$.

La connessione C^1 si ottiene quando i vettori tangenti $\tau_2(1, t_2) = \frac{d}{dt_2} P(1, t_2)$ e $\tau'_2(0, t'_2) = \frac{d}{dt'_2} P'(0, t'_2)$ sono identici in ogni punto comune tra le due patch. Questo garantisce che non solo la posizione, ma anche la derivata prima della curva siano continue, il che implica una transizione liscia senza discontinuità nelle tangenti.

3.1.4 De Boor-Coox

Giungiamo così all'ultimo punto di questo elaborato: l'algoritmo di De Boor-Coox. Analogamente a quanto succede per le curve, la generazione del punto $P(t_1, t_2)$ mediante definizione non è conveniente e non è un processo stabile numericamente. Tuttavia si può generare una patch di Bézier come unione di tutte le facce rettangolari generate dai vertici $P(t_1, t_2), P(t_1 + \Delta t_1, t_2), P(t_1 + \Delta t_1, t_2 + \Delta t_2)$ e $P(t_1, t_2 + \Delta t_2)$ generalizzando l'algoritmo di De Casteljau. Tale algoritmo prende il nome di De Boor-Coox:

```

if  $p_1 = 0$ ,
 $\forall q_1 = 0, 1, \dots, m_1, P_{q_1}$  costruito tramite De Casteljau.
if  $p_2 = 0$ ,
 $\forall q_2 = 0, 1, \dots, m_2, P_{q_1, q_2}^0 = P_{q_1, q_2}$ 
 $\forall p_2 = 1, 2, \dots, m_2$ 
 $\forall q_2 = p_2, p_2 + 1, \dots, m_2$ 
 $P_{q_1, q_2}^{p_2} = (1 - t_2)P_{q_1, q_2 - 1}^{p_2 - 1} + t_2 P_{q_1, q_2}^{p_2 - 1}$ 
set  $P_{q_1} = P_{q_1, m_2}$ 
set  $P_{q_1}^0 = P_{q_1}$ 
 $\forall p_1 = 1, 2, \dots, m_1$ ,
 $\forall q_1 = p_1, p_1 + 1, \dots, m_1$ 
do
 $P_{q_1}^{p_1} = (1 - t_1)P_{q_1 - 1}^{p_1 - 1} + t_1 P_{q_1}^{p_1 - 1}$ 
set  $P(t_1, t_2) = P_{m_1}^{m_1}$ 

```

Questo algoritmo conclude l'elaborato; come già menzionato, si invita il lettore ad approfondire l'argomento consultando la bibliografia.

Riferimenti bibliografici

Ionut Danaila, Pascal Joly, Sidi Mahmoud Kaber, and Marie Postel. *An Introduction to Scientific Computing: Fifteen Computational Projects Solved with MATLAB*. Springer Nature, 2023.