

Bézier

Descrizione

L'applicazione deve permettere all'utente di inserire punti in una finestra 2D al click del mouse. I punti sono disegnati e connessi con una poligonale

Obiettivi

1. Disegnare la curva di Bézier utilizzando l'evaluator di OpenGL
2. Disegnare la curva di Bézier mediante algoritmo di de Casteljau
3. Disegnare la curva di Bézier mediante algoritmo ottimizzato basato sulla suddivisione adattiva
4. Permettere la modifica della posizione dei punti di controllo tramite trascinamento con il mouse

Risultati

1. Disegnare la curva con l'evaluator di OpenGL

Nella funzione `drawBezier()` viene definito l'evaluator di OpenGL usando `glMap1f()`, successivamente viene disegnata la curva richiamando la funzione `glEvalCoord()` 100 volte con parametro valori nell'intervallo $[0, 1]$.

Questo metodo ha la limitazione di essere in grado di gestire solo 10 control points.

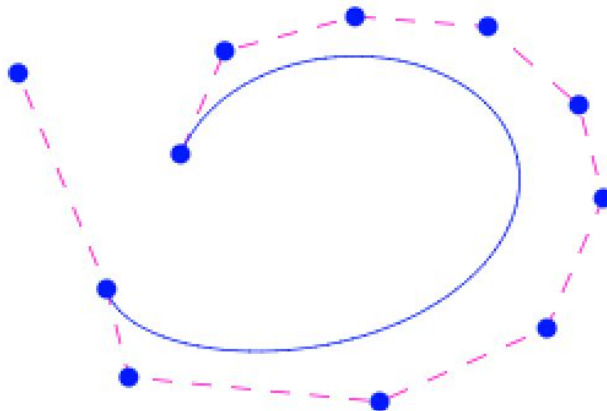


Fig 1. Curva di Bézier disegnata con l'evaluator di OpenGL

2. Disegnare la curva con De Casteljau

Per poter gestire più di 10 punti di controllo usiamo l'algoritmo di De Casteljau.

Nella funzione `drawBezier()` è stata sostituita `glEvalCoord()` con la funzione `decasteljau()`, la quale prende come parametro un valore t in $[0, 1]$ valutando la curva in quel punto.

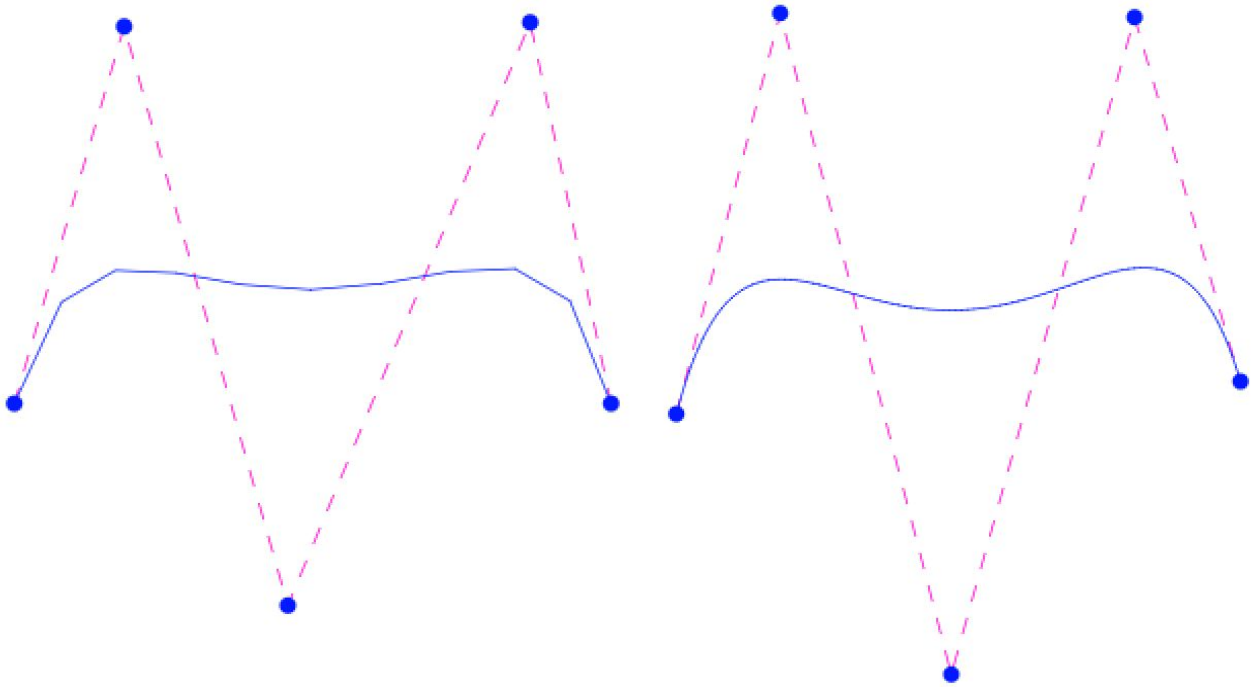


Fig 2. Algoritmo di De Casteljau valutato in 10 punti (a) e in 100 (b)

3. Disegnare la curva con suddivisione adattiva

Per ottimizzare il disegno della curva usiamo la suddivisione adattiva che consente di valutare la curva in un numero di punti sufficiente ad approssimarla con una determinata precisione.

La funzione `drawBezier()` chiama `adaptiveSubdivision()` che implementa l'algoritmo ottimizzato di suddivisione adattiva con una tolleranza di $0.00005f$

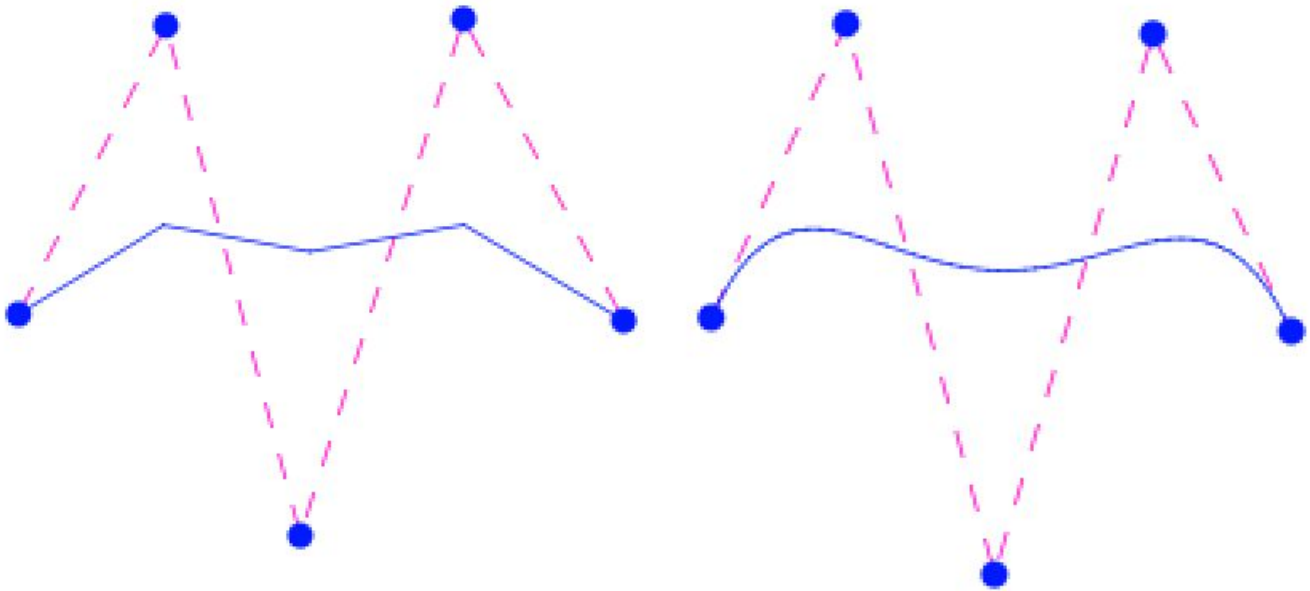


Fig 3. Suddivisione adattiva con tolleranza $0.05f$ (sinistra) e $0.00005f$ (destra)

4. Spostare i punti di controllo

Lo spostamento dei punti di controllo è stato implementato usando due funzioni

- `mouse()`
È stata impostata come handler dei click del mouse usando `glutMouseFunc(mouse)`, reagisce ai click aggiungendo un nuovo punto o selezionandone uno esistente se si trova ad una distanza di $0.05f$ dal punto cliccato
- `mousemove()`
È stata impostata come handler dei drag del mouse usando `glutMotionFunc(mousemove)`, reagisce ai drag spostando il punto selezionato, se esiste