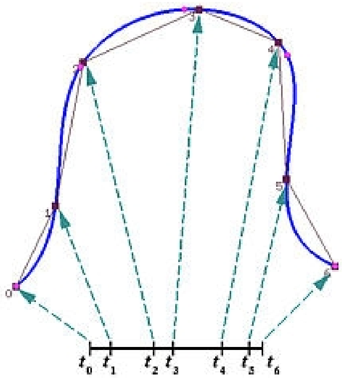
Interpolación con curvas B-spline

Modelado 3D

Introducción. Interpolación con curvas B-spline



- **Objetivo**: Dado un conjunto de $K+1$ puntos, los datos $d_0, \ldots, d_K$, la curva Bspline cúbica debe pasar por todos los puntos, los datos, en el orden dado.

Introducción. Curvas B-spline

- Una curva B-spline de grado p y $(n+1)$ puntos de control viene dada por:

$$p(u) = \sum_{i=0}^{n} p_i \cdot N_{i,p}(u) \qquad u \in [0,1)$$

- Las funciones base se calculan a partir de la fórmula recursiva de Cox de Boor:

$$N_{i,0}(u) = \begin{cases} 1 & u_i \leq u < u_{i+1} \\ 0 & c.c. \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} \cdot N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} \cdot N_{i+1,p-1}(u)$$

Introducción. Curvas B-spline

- Las funciones base son no-negativas.
- Número de nudos: $m = n + p + 1$.
- El cociente $0/0 = 0$
- La función base $N_{i,p}(u)$ es no nula en el intervalo de nudos $[u_i, u_{i+p+1})$. Es decir, $N_{i,p}(u)$ es no nula en $p+1$ intervalos:

$$[u_i, u_{i+1}), [u_{i+1}, u_{i+2}), \ldots, [u_{i+p}, u_{i+p+1})$$

- En el intervalo $[u_i, u_{i+1})$, como mucho, $p+1$ funciones base de grado p son no nulas:

$$N_{i-p,p}(u), N_{i-p+1,p}(u), N_{i-p+2,p}(u), \ldots, N_{i-1,p}(u) \text{ y } N_{i,p}(u)$$

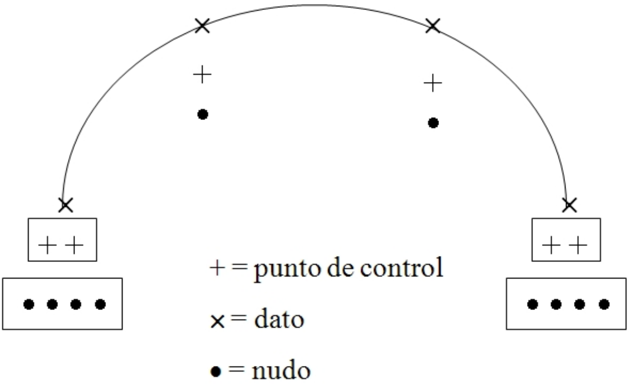
- La curva Bspline interpoladora es una curva Bspline *clamped*[1].
- Por lo tanto, el conjunto de $m+1$ nudos $U = (u_0, \dots, u_m)$ en el dominio de la curva:

$$U = \left(\underbrace{u_0, \dots, u_p}_{p+1}, \underbrace{u_{p+1}, \dots, u_n}_{n-p}, \underbrace{u_{n+1}, \dots, u_{n+p+1}}_{p+1} \right) =$$

$$= \left(\underbrace{0, \dots, 0}_{p+1}, \underbrace{u_{p+1}, \dots, u_n}_{n-p}, \underbrace{1, \dots, 1}_{p+1} \right)$$

con $m = n + p + 1$.

[1] Es una curva Bspline *clamped* ya que la curva debe interpolar el primer y el último dato que haremos corresponder con el primer y último punto de control de la misma.



- Hacemos corresponder a cada nudo con un dato y, por tanto, $n - p + 1 = K$.

Interpolación con curvas B-spline

- La curva Bspline cúbica *clamped* $(p = 3)$ debe pasar por todos los puntos $(K + 1 = n - 1$ datos$)$ en el orden dado:

$$
\begin{aligned}
d_0 &= p(u_3) = p_0 \cdot N_{0,3}(u_3) + p_1 \cdot N_{1,3}(u_3) + \ldots + p_n \cdot N_{n,3}(u_3) \\
d_1 &= p(u_4) = p_0 \cdot N_{0,3}(u_4) + p_1 \cdot N_{1,3}(u_4) + \ldots + p_n \cdot N_{n,3}(u_4) \\
&\ldots \\
d_k &= p(u_{k+3}) = p_0 \cdot N_{0,3}(u_{k+3}) + p_1 \cdot N_{1,3}(u_{k+3}) + \\
&+ \ldots + p_n \cdot N_{n,3}(u_{k+3}) \\
&\ldots \\
d_{n-2} &= p(u_{n+1}) = p_0 \cdot N_{0,3}(u_{n+1}) + p_1 \cdot N_{1,3}(u_{n+1}) + \\
&+ \ldots + p_n \cdot N_{n,3}(u_{n+1})
\end{aligned}
$$

Interpolación con curvas B-spline

- Es decir,

$$d_k = p\left(u_{k+3}\right) = \sum_{i=0}^{n} p_i \cdot N_{i,3}\left(u_{k+3}\right) \qquad 0 \le k \le n-2 = K$$

con

$$u_0, u_1, u_2, u_3 = 0 \qquad u_{n+1}, u_{n+2}, u_{n+3}, u_{n+4} = 1$$

Interpolación con curvas B-spline

- Esta curva B-spline *clamped* tiene $n+1$ puntos de control **desconocidos**.
  - Es decir, tenemos $n-1$ ecuaciones, correspondientes a los $n-1$ datos, y $n+1$ incógnitas, los $n+1$ puntos de control.
- Por consiguiente, debemos definir dos condiciones adicionales para que el sistema de ecuaciones tenga solución: repetimos los puntos iniciales y finales:

$$p_0 = p_1 \qquad p_{n-1} = p_n$$

Interpolación con curvas B-spline

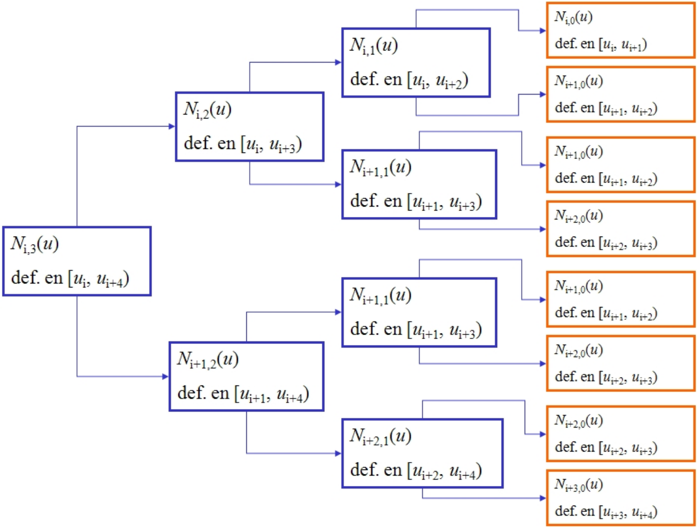
- Hay que evaluar la función base en los valores de los nudos $u_3, \ldots, u_{n+1}$:

$$d_0 = p\left(u_3\right) \quad \ldots \quad d_k = p\left(u_{k+3}\right) \quad \ldots \quad d_{n-2} = p\left(u_{n+1}\right)$$

- En el intervalo $\left[u_{k+3}, u_{k+4}\right)$ sólo se encuentran definidas, como mucho, cuatro funciones base:

$$N_{k,3}\left(u\right) \quad N_{k+1,3}\left(u\right) \quad N_{k+2,3}\left(u\right) \quad N_{k+3,3}\left(u\right)$$

Interpolación con curvas B-spline



Interpolación con curvas B-spline

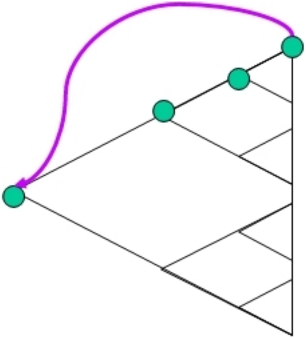
- Debemos encontrar el valor de estas funciones base calculando los valores no nulos de las hojas del árbol y evaluando las expresiones de las funciones base de atrás-adelante en el árbol de recursión de las mismas.

$$N_{k,3}\left(u\right) \quad N_{k+1,3}\left(u\right) \quad N_{k+2,3}\left(u\right) \quad N_{k+3,3}\left(u\right)$$

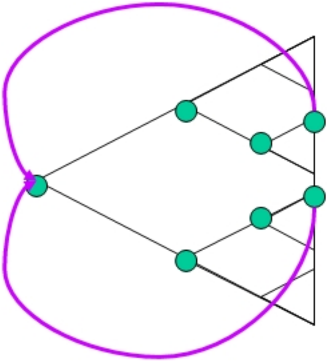
- Como veremos a continuación, solamente las tres primeras funciones base son no nulas en el intervalo $\left[u_{k+3}, u_{k+4}\right)$.

Interpolación con curvas B-spline

$$N_{k,3}\left(u_{k+3}\right) = \frac{\left(u_{k+4} - u_{k+3}\right)^2}{\left(u_{k+4} - u_{k+1}\right)\left(u_{k+4} - u_{k+2}\right)} = \alpha_k$$

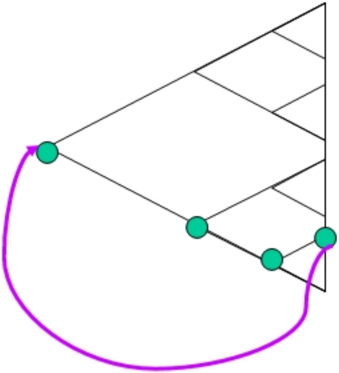


$$N_{k+1,3}\left(u_{k+3}\right)=\frac{\left(u_{k+3}-u_{k+1}\right)\left(u_{k+4}-u_{k+3}\right)}{\left(u_{k+4}-u_{k+1}\right)\left(u_{k+4}-u_{k+2}\right)}+$$
$$+\frac{\left(u_{k+5}-u_{k+3}\right)\left(u_{k+3}-u_{k+2}\right)}{\left(u_{k+5}-u_{k+2}\right)\left(u_{k+4}-u_{k+2}\right)}=\beta_k$$



Interpolación con curvas B-spline

$$N_{k+2,3}\left(u_{k+3}\right) = \frac{\left(u_{k+3} - u_{k+2}\right)^2}{\left(u_{k+4} - u_{k+2}\right)\left(u_{k+5} - u_{k+2}\right)} = \gamma_k$$



Interpolación con curvas B-spline

- Por tanto, los datos pueden especificarse como la suma de tres términos:

$$d_k = p\left(u_{k+3}\right) = N_{k,3}\left(u_{k+3}\right) \cdot p_k + N_{k+1,3}\left(u_{k+3}\right) \cdot$$
$$\cdot p_{k+1} + N_{k+2,3}\left(u_{k+3}\right) \cdot p_{k+2} =$$
$$= \alpha_k \cdot p_k + \beta_k \cdot p_{k+1} + \gamma_k \cdot p_{k+2}$$

Interpolación con curvas B-spline

- En forma matricial:

$$\begin{bmatrix} \beta_0 & \gamma_0 & & & & & \\ \alpha_1 & \beta_1 & \gamma_1 & & & & \\ & & & \cdots & & & \\ & & & & \alpha_{n-3} & \beta_{n-3} & \gamma_{n-3} \\ & & & & & \alpha_{n-2} & \beta_{n-2} \end{bmatrix} \cdot \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_{n-2} \\ p_{n-1} \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_{n-3} \\ d_{n-2} \end{bmatrix}$$

Interpolación con curvas B-spline

- Este tipo de matrices se denominan matrices de los **sistemas tridiagonales** (sólo los elementos de la diagonal y sus dos vecinos son no nulos):

$$\begin{bmatrix} \beta_0 & \gamma_0 & & & & & \\ \alpha_1 & \beta_1 & \gamma_1 & & & & \\ & & & \cdots & & & \\ & & & & \alpha_{n-3} & \beta_{n-3} & \gamma_{n-3} \\ & & & & & \alpha_{n-2} & \beta_{n-2} \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{n-3} \\ X_{n-2} \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_{n-3} \\ d_{n-2} \end{bmatrix}$$

Interpolación con curvas B-spline

- Resolución eficaz del sistema de ecuaciones:
  - Cualquier fila de la matriz representa una ecuación lineal del sistema.
  - Podemos calcular múltiplos de estas ecuaciones y operar entre ellos sin afectar al resultado.
- Realizaremos los cálculos en dos pasos:
  - Paso *Forward*
  - Paso *Backward*

Interpolación con curvas B-spline. Paso Forward

$$\begin{bmatrix} \beta_0 & \gamma_0 & & & & & \\ \alpha_1 & \beta_1 & \gamma_1 & & & & \\ & & & \cdots & & & \\ & & & & \alpha_{n-3} & \beta_{n-3} & \gamma_{n-3} \\ & & & & & \alpha_{n-2} & \beta_{n-2} \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{n-3} \\ X_{n-2} \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_{n-3} \\ d_{n-2} \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} 1 & \lambda_0 & & & & \\ & 1 & \lambda_1 & & & \\ & & & \cdots & & \\ & & & & 1 & \lambda_{n-3} \\ & & & & & 1 \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{n-3} \\ X_{n-2} \end{bmatrix} = \begin{bmatrix} \delta_0 \\ \delta_1 \\ \vdots \\ \delta_{n-3} \\ \delta_{n-2} \end{bmatrix}$$

Pasos Forward y Backward

- Paso *Forward*

$$\lambda_0 = \frac{\gamma_0}{\beta_0} \qquad \lambda_i = \frac{\gamma_i}{\beta_i - \alpha_i \lambda_{i-1}} \qquad i = 1, \ldots, n-3$$

$$\delta_0 = \frac{d_0}{\beta_0} \qquad \delta_i = \frac{d_i - \alpha_i \delta_{i-1}}{\beta_i - \alpha_i \lambda_{i-1}} \qquad i = 1, \ldots, n-2$$

- Paso *Backward*

$$X_{n-2} = \delta_{n-2}$$

$$X_i = \delta_i - \lambda_i \cdot X_{i+1} \quad i = n-3, \ldots, 0$$

Interpolación con curvas B-spline

- Finalmente, nuestro conjunto de puntos de control es el conjunto de variables X_i resultantes de la ejecución del paso *backward*.

$$p_0 = p_1 \qquad \underset{i=0,\ldots,n-2}{p_{i+1} = X_i} \qquad p_{n-1} = p_n$$

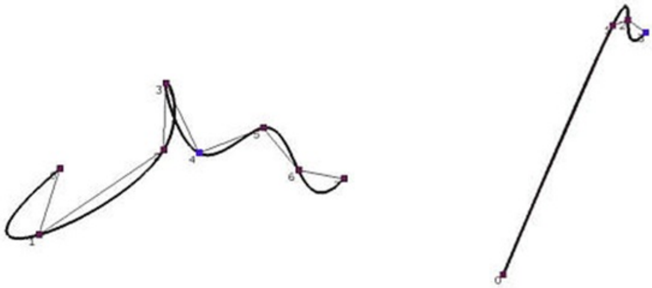
Vector de nudos: Método uniformemente espaciado

- La elección del conjunto de nudos para una curva B-spline cúbica $(p = 3)$ con $(n+1)$ puntos de control y $(n+5)$ nudos en este caso no tiene en cuenta la situación de los puntos (los datos):

$$\left\{ \begin{array}{c} u_0 = u_1 = u_2 = u_3 = 0 \\ u_{k+3} = \frac{k}{n-2} \\ u_{n+1} = u_{n+2} = u_{n+3} = u_{n+4} = 1 \end{array} \right. \qquad 0 < k < n-2$$

Vector de nudos: Método uniformemente espaciado

- Cuando los puntos no se encuentran uniformemente espaciados, este método puede crear formas extrañas tales como picos, protuberancias y bucles.



- La B-spline uniforme oscila porque el vector de nudos uniforme no tiene en cuenta la geometría de los datos.

Vector de nudos: Método Chord Length

- Si la curva interpoladora se acerca mucho al polígono formado por los puntos dados por el usuario (los datos), la longitud del segmento de curva entre dos datos adyacentes es igual a la longitud de la cuerda entre estos dos puntos.
- Dados los datos $d_0, d_1, \ldots, d_{n-2}$. La longitud entre los datos d_{i-1} y d_i es:

  $$|d_i - d_{i-1}|$$

  y la longitud del polígono de datos es la suma de las longitudes de estas cuerdas:

  $$L = \sum_{i=1}^{n-2} |d_i - d_{i-1}|$$

Vector de nudos: Método Chord Length

- La relación entre la longitud de la cuerda del dato d_0 a d_k con respecto a la longitud total del polígono es:

$$L_k = \frac{\sum_{i=1}^{k} |d_i - d_{i-1}|}{L}$$

- Si el dominio de los nudos es $[0, 1]$:

$$\begin{cases} u_0 = u_1 = u_2 = u_3 = 0 \\ u_{k+3} = L_k = \frac{1}{L} \sum_{i=1}^{k} |d_i - d_{i-1}| & 0 < k < n-2 \\ u_{n+1} = u_{n+2} = u_{n+3} = u_{n+4} = 1 \end{cases}$$

Vector de nudos: Método Chord Length

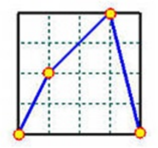
- El espaciado de los nudos es proporcional a la distancia entre los puntos de los datos lo cual da lugar a B-splines no uniformes.
- Se utiliza para el suavizado de las curvas.

Vector de nudos: Método Chord Length

Example

Sean $d_0 = (0,0)$, $d_1 = (1,2)$, $d_2 = (3,4)$ y $d_3 = (4,0)$.

$$|d_1 - d_0| = \sqrt{5} = 2.236$$
$$|d_2 - d_1| = 2\sqrt{2} = 2.828$$
$$|d_3 - d_2| = \sqrt{17} = 4.123$$
$$L = \sqrt{5} + 2\sqrt{2} + \sqrt{17} = 9.8176$$



Vector de nudos: Método Chord Length

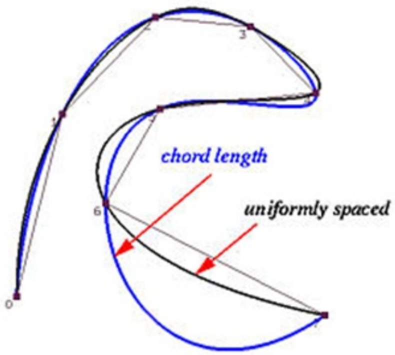
Example

[cont.] El valor de los nudos es:

$$\begin{cases} u_0 = u_1 = u_2 = u_3 = 0 \\ u_4 = \frac{|d_1 - d_0|}{L} = \frac{2.236}{9.8176} = 0.227 \\ u_5 = \frac{|d_1 - d_0| + |d_2 - d_1|}{L} = \frac{2.236 + 2.828}{9.8176} = 0.515 \\ u_{n+1} = u_{n+2} = u_{n+3} = u_{n+4} = 1 \end{cases}$$

Vector de nudos: Método Chord Length

- Este método suele utilizarse frecuentemente aunque, en alguna ocasión, la existencia de una cuerda de gran longitud puede dar lugar a una mala aproximación ya que puede producirse una "protuberancia" innecesaria en la curva.



Vector de nudos: Método Centrípeta

- Dados los datos $d_0, d_1, \ldots, d_{n-2}$. La longitud entre los datos d_{i-1} y d_i es:

$$\left|d_i - d_{i-1}\right|^a$$

y la longitud del polígono de datos es la suma de las longitudes de estas cuerdas:

$$L = \sum_{i=1}^{n-2} \left|d_i - d_{i-1}\right|^a$$

Vector de nudos: Método Centrípeta

- La relación entre la longitud de la cuerda del dato d_0 a d_k con respecto a la longitud total del polígono es:

$$L_k = \frac{\sum_{i=1}^{k} |d_i - d_{i-1}|^a}{L}$$

- Si el dominio de los nudos es $[0, 1]$:

$$\begin{cases} u_0 = u_1 = u_2 = u_3 = 0 \\ u_{k+3} = L_k = \frac{1}{L} \sum_{i=1}^{k} |d_i - d_{i-1}|^a & 0 < k < n-2 \\ u_{n+1} = u_{n+2} = u_{n+3} = u_{n+4} = 1 \end{cases}$$

Vector de nudos: Método Centrípeta

- Si $a = 1$, el método de la fuerza centrípeta es el método de la longitud de la cuerda.
- Si $a < 1$ (por ejemplo, $a = 1/2$), se cumple que:

$$|d_i - d_{i-1}|^a < |d_i - d_{i-1}|$$

- Es decir:
  - el impacto de una cuerda larga (es decir, longitud > 1) sobre la longitud del polígono de datos se reduce.
  - el impacto de una cuerda corta (es decir, longitud < 1) sobre la longitud del polígono de datos aumenta.

Vector de nudos: Método Centrípeta

Example

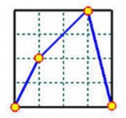
Sea $a = \frac{1}{2}$ y los datos $d_0 = (0,0)$, $d_1 = (1,2)$, $d_2 = (3,4)$ y $d_3 = (4,0)$.

$$|d_1 - d_0|^{1/2} = \sqrt{\sqrt{5}} = 1.495$$

$$|d_2 - d_1| = \sqrt{2\sqrt{2}} = 1.682$$

$$|d_3 - d_2| = \sqrt{\sqrt{17}} = 2.031$$

$$L = \sqrt{\sqrt{5}} + \sqrt{2\sqrt{2}} + \sqrt{\sqrt{17}} = 5.208$$



Vector de nudos: Método Centrípeta

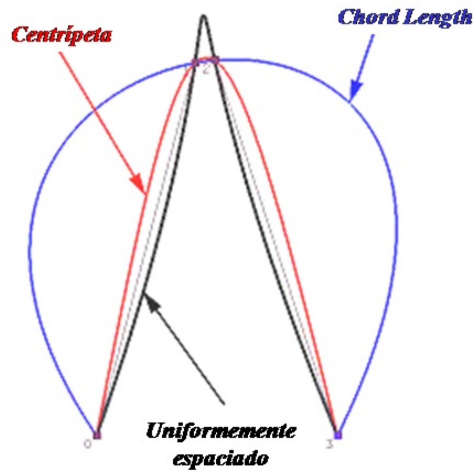
Example

[cont.] El valor de los nudos es:

$$\left\{ \begin{array}{c} u_0 = u_1 = u_2 = u_3 = 0 \\ u_4 = \frac{|d_1 - d_0|}{L} = \frac{1.495}{5.208} = 0.2871 \\ u_5 = \frac{|d_1 - d_0| + |d_2 - d_1|}{L} = \frac{1.495+1.682}{5.208} = 0.6101 \\ u_{n+1} = u_{n+2} = u_{n+3} = u_{n+4} = 1 \end{array} \right.$$

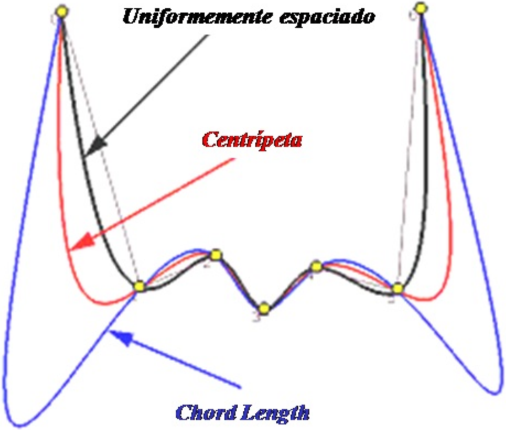
Vector de nudos: Método Centrípeta

- En este caso puede apreciarse el buen comportamiento de la interpolación según el método "centrípeta".



Vector de nudos: Método Centrípeta

- Sin embargo, no podemos hablar de un método eficiente en concreto. En este caso, puede observarse que la mejor aproximación la da el método de "uniformemente espaciado".



Bibliografía I

John C. Beatty, Richard H. Bartels, and Brian A. Barsky.
An Introduction to Splines for Use in Computer Graphics and Geometric Modeling.
Morgan Kaufmann, 1 edition, 1989.

Sambhunath Biswas and Brian C. Lovell.
Bézier and Splines in Image Processing and Machine Vision.
Springer, 1 edition, December 2008.

Les A. Piegl and Wayne Tiller.
The NURBS Book.
Springer, 2nd edition, November 1997.

David Salomon.
Curves and Surfaces for Computer Graphics.
Springer, 1 edition, September 2006.

Bibliografía II

M. Sarfraz.
Interactive Curve Modeling: With Applications to Computer Graphics, Vision and Image Processing.
Springer, 1 edition, October 2007.

C.K. Shene.
Cs3621. introduction to computing with geometry notes.

Alan Watt and Mark Watt.
Advanced Animation and Rendering Techniques.
AMC Press, Addison-Wesley, New York, 1992.