Bezier racionales y NURBS

NURBS: Non-uniform rational B-spline

Problema: con curvas polinomiales no se pueden cubrir todo tipo de curvas.

Ejemplo: las circunferencias no admiten respresentación paramétrica polinomial.

Sin embargo,

$$\left(\frac{2t(1-t)}{(1-t)^2+t^2}, \frac{(1-t)^2-t^2}{(1-t)^2+t^2} \right)$$ $t \in [0,1]$ representa media circunferencia

o bien,

$$\left(\frac{2t}{1+t^2}, \frac{1-t^2}{1+t^2} \right)$$ $t \in (-\infty, \infty)$ representa la circunferencia entera

son representaciones paramétricas de la circunferencia.

Ojo: ¡¡son siempre cociente de polinomios!!

Curvas racionales

- Curva de Bezier racional
- NURBS (non-uniform-rational-B-spline)

los puntos de control en **coordenadas homogéneas**

Coordenadas homogéneas: en R^3 los puntos se representan mediante 4 coordenadas

p=(x, y, z, 1) ó, equivalentemente, p=(wx, wy, wz, w), $w \neq 0$
w se llama "peso" de p
para recuperar p en R^3, dividimos sus tres primeras coordenadas por w
los puntos de peso 0 son "puntos del infinito"

Procedimiento:

Construimos curvas con coordenadas homogéneas.
Dividimos por la última coordenada para recuperar la curva en R^3.

Bezier racionales

Dados los puntos $b_0,\ldots, b_n$, considerar los mismos puntos en coordenadas homogéneas: $b_0^w=(w_0b_0,\underbrace{w_0}_{\text{pesos asociados}}), \ldots, b_n^w=(w_nb_n, \underbrace{w_n}_{\text{pesos asociados}})$, con $\underbrace{w_i}_{\text{pesos}}\neq 0$, $i=0,\ldots,n$.

Construir la curva de Bezier de estos puntos:

$$\boxed{B^w(t)=\sum_{i=0}^n B_i^n(t)\, b_i^w = \sum_{i=0}^n B_i^n(t)\,(w_ib_i\, ,\, w_i)= \left(\sum_{i=0}^n B_i^n(t)w_ib_i\, ,\, \sum_{k=0}^n B_k^n(t)\, w_k \right)}$$

Recuperar la curva en coordenadas no homogéneas:

$$\boxed{B(t)= \sum_{i=0}^n \underbrace{\left(\frac{w_i\, B_i^n(t)}{\sum_{k=0}^n w_kB_k^n(t)}\right)}_{\text{nuevas funciones de base}} b_i}$$

nuevas funciones de base

Propiedades de las Bezier racionales

•Las funciones de base suman 1.

•Si se impone que todos los pesos sean positivos, las funciones de base toman valores entre 0 y 1.

•La curva es una combinación convexa de los puntos de control.

•Interpolación en los extremos y tangencia al polígono de control en los extremos.

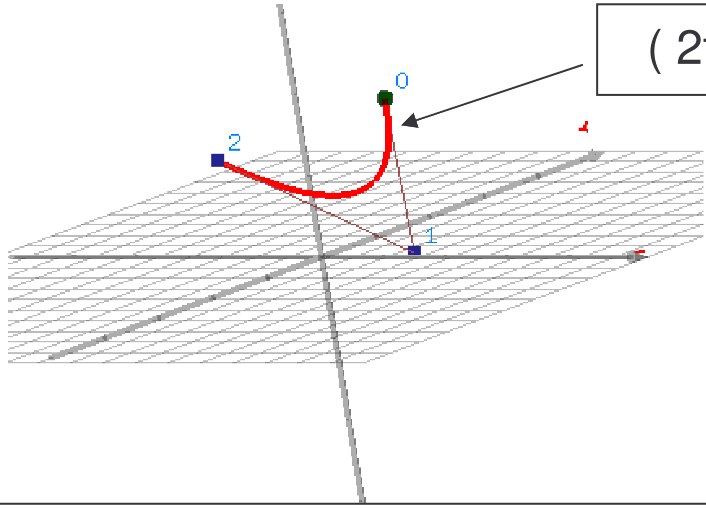
•La última coordenada (homogénea) de cada punto se puede utilizar para dar más o menos peso a un punto. Si se eleva su peso, la curva se dobla hacia el punto.

•Las curvas de Bezier racionales son estables bajo proyectividades (ej: afinidades, transformaciones de perspectiva, etc)

•Si utilizamos puntos con peso cero podremos construir secciones cónicas: las secciones cónicas son curvas de Bezier racionales de grado 2. Se pueden ver como proyecciones sobre el plano z=1 de parábolas en 3D.

Ejemplo:

<u>curva de Bezier</u> con puntos de control (0,1,1), (1,0,0) y (0,-1,1):



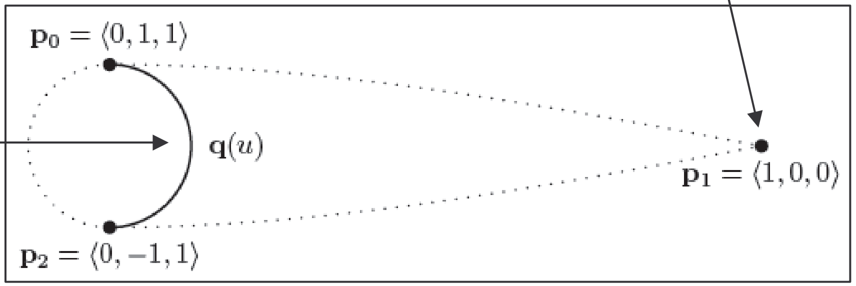
$(2t(1-t) , (1-t)^2-t^2 , (1-t)^2+t^2)$

proyección en z=1

<u>curva de Bezier racional</u>:

se trata como un punto del infinito

$$\left(\frac{2t(1-t)}{(1-t)^2+t^2} , \frac{(1-t)^2-t^2}{(1-t)^2+t^2} \right)$$



NURBS

Dados el grado k, los puntos $b_0,\ldots, b_n$, y los nodos $x_0,\ldots,x_{n+k+1}$, considerar los mismos puntos en coordenadas homogéneas:

$$b_0^w=(w_0 b_0,\underbrace{w_0}_{\text{pesos asociados}}),\ \ldots,\ b_n^w=(w_n b_n,\ \underbrace{w_n}_{\text{pesos asociados}}),\ \text{con } \underbrace{w_i}_{\text{pesos}}\neq 0,\ i=0,\ldots,n.$$

Construir el B-spline de estos puntos:

$$\boxed{P^w(t)=\sum_{i=0}^n N_i^k(t)\, b_i^w = \sum_{i=0}^n N_i^k(t)\,(w_i b_i\,,\, w_i)= \left(\sum_{i=0}^n N_i^k(t) w_i b_i\,,\, \sum_{l=0}^n N_l^k(t)\, w_l \right)}$$

Recuperar la curva en coordenadas no homogéneas:

$$\boxed{P(t)= \sum_{i=0}^n \underbrace{\left(\frac{w_i\, N_i^k(t)}{\sum_{l=0}^n w_l N_l^k(t)}\right)}_{\text{nuevas funciones de base } R_i^k(t)} b_i}$$

Propiedades de las funciones de base

Las nuevas funciones de base son: $$R_i^k(t)= \frac{w_i\, N_i^k(t)}{\sum_{l=0}^{n} w_l N_l^k(t)}$$

Propiedades:

- Las funciones son funciones racionales de grado k.
- Son no negativas (supondremos que los puntos nunca tiene pesos negativos).
- En el intervalo (x_i, x_{i+1}) solo son no nulas las funciones $R_{i-k}^k(t),\ldots, R_i^k(t)$, y suman 1
- En un nodo x_i de multiplicidad m, la función de base $R_i^k(t)$ es de clase C^{k-m}.
- Si el peso w_i de todos los puntos siempre es $c \neq 0$, las funciones de base coinciden con las funciones de base habituales de los B-splines.

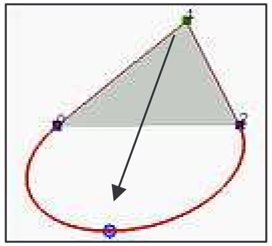
Propiedades de las NURBS

1. Control local: las $R_i^k(t)$ tienen soporte local.

2. Invarianza proyectiva.

3. Envolvente convexa fuerte. ¡Ojo, es necesario que los pesos sean ≥ 0!
   (si se utilizan pesos negativos, esta propiedad no se cumple)

   Ejemplo: clamped knots y peso -0.5 para el punto central

4. La curva está formada a trozos por curvas de Bezier racionales de grado k.

5. Si los k+1 primeros nodos coinciden, la curva empieza en b_0
   Si los k+1 últimos nodos coinciden, la curva termina en b_n

6. La curva es de clase C^{k-m} en nodos de multiplicidad m.

7. Las curvas de Bezier y los B-splines son casos particulares de NURBS.

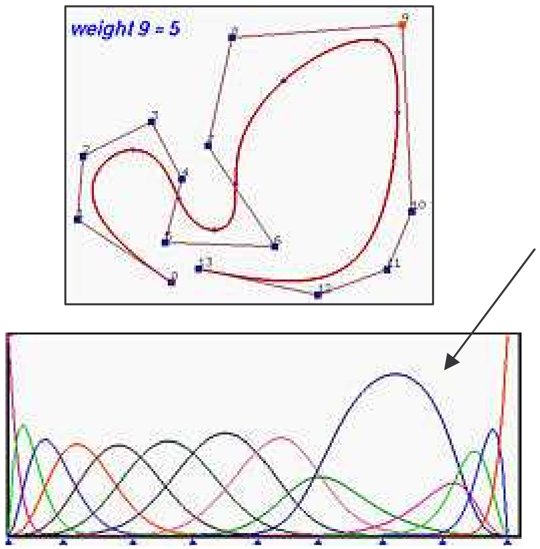
NURBS: modificación de pesos

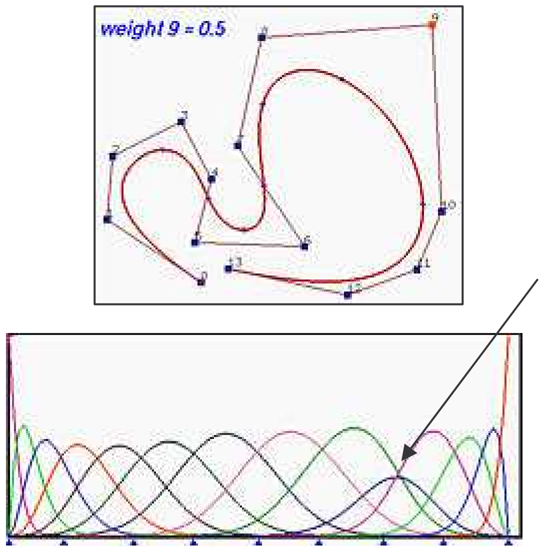
Las nuevas funciones de base son: $\boxed{R_i^k(t)= \dfrac{w_i\, N_i^k(t)}{\sum_{l=0}^{n} w_l N_l^k(t)}}$

(los pesos intervienen directamente en la definición de las func. de base)

•Si aumentamos el valor de un peso w_i, la función de base $R_i^k(t)$ aumenta, lo que implica que la NURBS "se curva" hacia el punto b_i. Si disminuimos el valor del peso, la curva "e aleja" del punto b_i.

•Si hacemos un peso $w_i=0$, la función de base $R_i^k(t)=0$ y el punto b_i no interviene en la fórmula de la NURBS.





NURBS: inserción de nodos

•Una NURBS en 3-dim. se puede ver como una B-spline en 4-dim., sin mas que considerar los puntos $b_0,\dots, b_n$ con pesos asociados $w_0,\dots, w_n$ y los nuevos puntos de control $(w_0b_0, w_0),\dots, (w_nb_n, b_n)$.

•Para insertar nodos en una NURBS nos pasamos al B-spline asociado y utilizamos los algoritmos de inserción de nodos en B-splines. Recalculamos los nuevos puntos de control con los algoritmos de B-splines y "volvemos" al espacio de 3-dim. dividiendo cada punto por su última coordenada, que hará de peso del nuevo punto.

•<u>Algoritmo de de Boor</u>:

se basa en la inserción de nodos hasta tener multiplicidad k

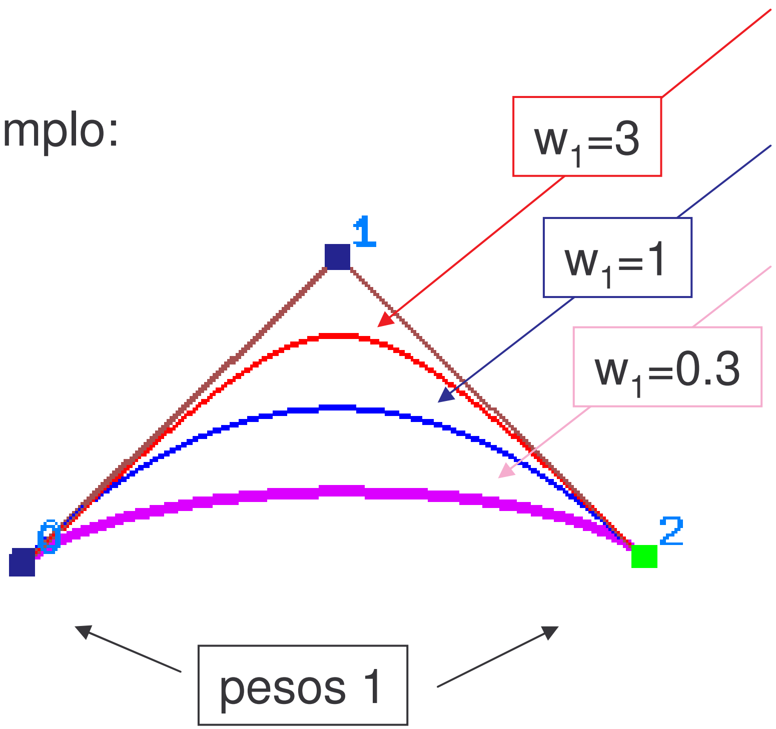
(se trabaja en dimensión 4 con los algoritmos de B-spline)

Trazo de secciones cónicas

Para dibujar una sección cónica se pueden utilizar curvas de Bezier racionales. En concreto, una curva de Bezier racional con 3 puntos de control b_0, b_1 y b_2, de pesos 1, w y 1, respectivamente, define:

- una hipérbola si w>1
- una parábola si w=1
- una elipse si w<1

Un ejemplo:



¿cómo hallar el peso w_1?

lo veremos para circunferencias

Arcos circulares

Según la transparencia anterior, para trazar un arco circular (elipse), se utilizan curvas de Bezier racionales. Habrá que hallar w_1. Con desarrollos geométricos (ver Notas de C.-K. Shene), se llega a:

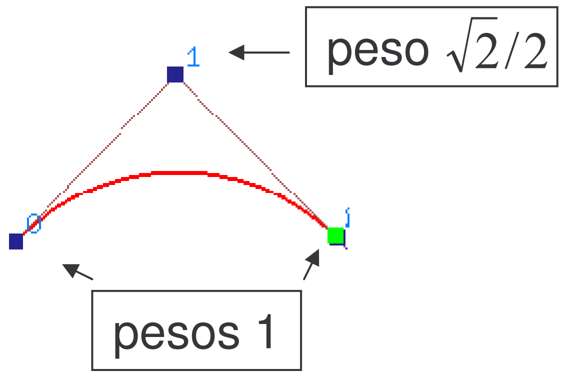
- el módulo de b_0b_1 tiene que coincidir con el de b_1b_2.
- el peso del punto b_1 tiene que coincidir con el seno de **a**/2, siendo **a** el ángulo formado por los vectores b_0b_1 y b_1b_2.
- los otros dos puntos tienen peso 1.

Casos particulares

Caso 1:

si $a=\pi/2$ entonces $w=\text{sen}(\pi/4)=\sqrt{2}/2$

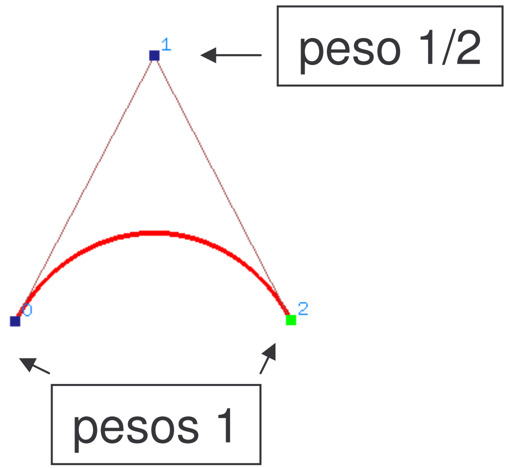
(se obtiene un cuarto de circunferencia)



Caso 2:

si $a=\pi/3$ entonces $w=\text{sen}(\pi/6)=\ 1/2$

(se obtiene un tercio de circunferencia)



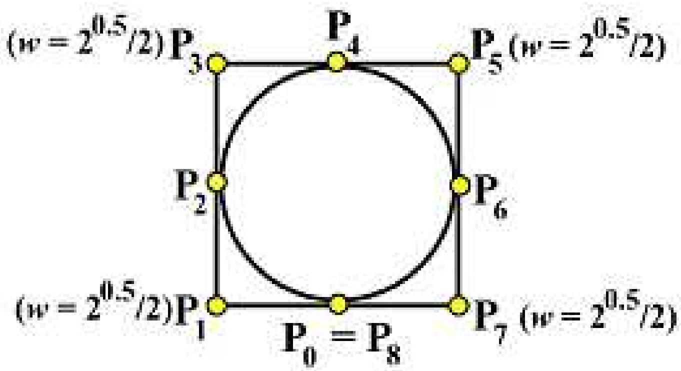
Ejemplo: cómo hacer circunferencias

Los B-splines incluyen las curvas de Bezier a trozos, así que los NURBS han de incluir a las curvas de Bezier racionales a trozos.

Para dibujar una circunferencia, podemos unir los cuatro cuartos de circunferencia vistos en el Ejemplo 1, o bien tres tercios de circunferencia como en el Ejemplo 2.

Caso 1: los puntos de control son $P_0,\ldots,P_8=P_0$. El grado k=2. Se necesitan los nodos $[x_0,x_1,\ldots,x_{n+k+1}]=[x_0,\ldots,x_{11}]$

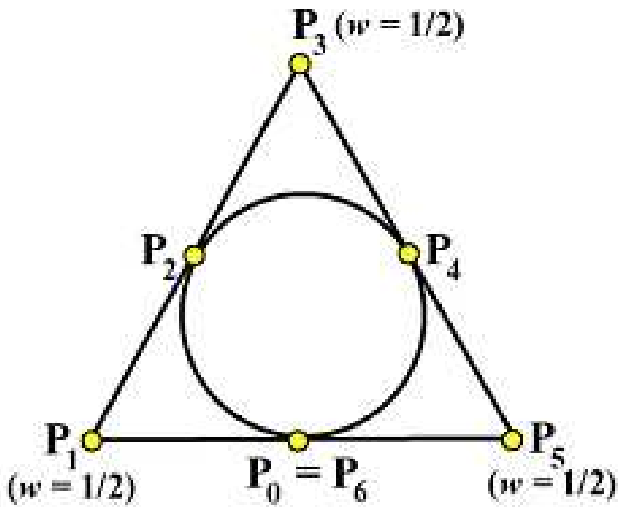
Vamos a calcularlos:



- $x_0=x_1=x_2=0$, $x_9=x_{10}=x_{11}=1$
- La curva se dibuja en 4 tiempos iguales así que habrá nodos interiores ¼, ½, ¾.
- La curva pasa por P_2, P_4 y P_6 en esos tiempos. Por el algoritmo de De Boor, son nodos que aparecen 2 veces cada uno, es decir, $x_3=x_4=¼$, $x_5=x_6=½$, $x_7=x_8=¾$.

Ejemplo: cómo hacer circunferencias

Caso 2: los puntos de control son $P_0,\ldots,P_6=P_0$. El grado k=2. Se necesitan los nodos $[x_0,x_1,\ldots,x_{n+k+1}]=[x_0,\ldots,x_9]$



Vamos a calcularlos:

- $x_0=x_1=x_2=0$, $x_7=x_8=x_9=1$
- La curva se dibuja en 3 tiempos iguales así que habrá nodos interiores 1/3, 2/3.
- La curva pasa por P_2 y P_4 en esos tiempos. Por el algoritmo de De Boor, son nodos que aparecen 2 veces cada uno, es decir, $x_3=x_4=1/3$, $x_5=x_6=2/3$.

<u>Ejercicio</u>: dibuja tus iniciales con B-splines racionales y con NURBS.
Escoge los grados, los vectores de nodos (equiespaciados, "clamped" o no uniformes), y los pesos de los puntos de control que consideres.
Indícalo en los dibujos que entregues.