

Tema 4: Curvas polinómicas a trozos

Leonardo Fernández Jambrina
E.T.S.I. Navales, Universidad Politécnica de Madrid
Avenida de la Memoria 4
E-28040-Madrid, España
Correo-e: leonardo.fernandez@upm.es

1. Introducción

[Vídeo de Curvas spline](#)

En el tema anterior estudiamos que una manera de superar la rigidez de las curvas de Bézier consistía en recurrir a curvas racionales. Aún así, la flexibilidad que proporcionan los pesos de las curvas racionales es muy limitada, ya que, como vimos, existen infinitas configuraciones de pesos para un polígono de control que corresponden a una misma curva, de modo que siempre podemos fijar dos de ellos. Así pues, una cónica no tiene tres grados de libertad adicionales frente a la parábola, sino tan sólo uno.

Además, como las curvas racionales se corresponden con curvas polinómicas en el espacio, su capacidad de variación no es mucho mayor que la de estas.

Aparte, la posibilidad de aparición de polos en la parametrización y puntos en el infinito, los vectores de control, nos fuerzan a ser cuidadosos en la elección de pesos y a mantenernos en el orden de magnitud de la unidad.

Por todo ello, parece aconsejable seguir buscando generalizaciones alternativas de las curvas de Bézier.

En la introducción del tema anterior apuntábamos una, las curvas polinómicas a trozos. En vez de tratar de cubrir toda la curva con una sola parametrización polinómica o racional, emplearemos varias. De esta manera conseguiremos mayor flexibilidad, ya que, modificando una parametrización,

sólo alteraremos la parte de la gráfica correspondiente a esta y , al disponer de tantos polígonos de control como tramos polinómicos tiene la curva, el número de grados de libertad para el diseño aumenta considerablemente. Además, como también podemos recurrir a parametrizaciones racionales, curvas racionales a trozos, también incorporaremos las ventajas del uso de los pesos.

Sin embargo, no todo es tan sencillo como multiplicar el número de vértices del polígono de control por el número de tramos que queremos que tenga la curva. En general, trabajaremos con curvas continuas y suaves. De hecho, las parametrizaciones polinómicas y racionales son de clase C^∞ , son diferenciables tantas veces como queramos. Pero en el tema segundo vimos que las condiciones de diferenciabilidad imponían una serie de restricciones sobre los vértices de los polígonos de control próximos a las uniones en las curvas compuestas. Si queríamos una parametrización global de clase C^s , los $s + 1$ vértices de cada polígono de control más próximos a la unión estaban ligados entre sí. Por ejemplo, tener una tangente continua requería la alineación de los vectores $\mathbf{c}_{n-1}\mathbf{c}_n$ y $\tilde{\mathbf{c}}_0\tilde{\mathbf{c}}_1$.

Salvo que queramos expresamente que la curva presente puntos angulosos, deberemos implementar las condiciones anteriores de modo que la curva presente siempre la clase de diferenciabilidad que hayamos escogido de antemano, de modo que las modificaciones que realice el diseñador sobre los polígonos de control no afecten a estas propiedades.

Estudiaremos inicialmente como construir parábolas de clase C^1 a trozos y cúbicas de clase C^2 y C^1 a trozos antes de introducir un procedimiento general.

2. Parábolas C^1 a trozos

[Vídeo de Parábolas a trozos](#)

Consideremos una curva plana formada por N tramos parabólicos. En vez de utilizar la notación con tildes del primer tema, numeraremos consecutivamente los vértices de los polígonos de control, teniendo en cuenta que, para que la curva sea continua, el vértice último de un polígono debe coincidir con el vértice primero del siguiente. Así pues, tendremos una colección de vértices $\{c_0, \dots, c_{2N}\}$, en la cual $\{c_0, c_1, c_2\}$ es el polígono de control del primer tramo, $\{c_2, c_3, c_4\}$ es el polígono del segundo, $\{c_{2(i-1)}, c_{2i-1}, c_{2i}\}$, el del

i -ésimo, y $\{c_{2N-2}, c_{2N-1}, c_{2N}\}$, el del último.

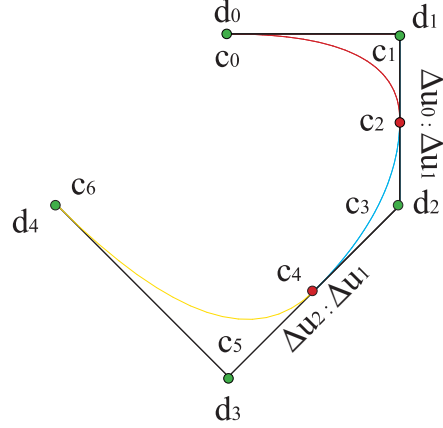


Figura 1: Parábola a trozos

Cada polígono definirá una parametrización en un intervalo. Así el primer tramo estará definido en un intervalo $[u_0, u_1]$, el segundo, en $[u_1, u_2]$, el i -ésimo, en $[u_{i-1}, u_i]$ y el último, en $[u_{N-1}, u_N]$.

Si queremos que la curva compuesta sea de clase C^1 en todo el intervalo $[u_0, u_N]$, tal como se estudió en el tema segundo, tendremos que exigir que lo sea para cada valor $u = u_i$,

$$\frac{\Delta c_{2i-1}}{\Delta u_{i-1}} = \frac{\Delta c_{2i}}{\Delta u_i}, \quad i = 1, \dots, N-1, \quad (1)$$

es decir, que los vectores $\mathbf{c}_{2i-1}\mathbf{c}_{2i}$ y $\mathbf{c}_{2i}\mathbf{c}_{2i+1}$ son paralelos y están en la proporción $\Delta u_{i-1} : \Delta u_i$. Ya sabemos que si los vectores son paralelos, pero no respetan la proporción, la curva tiene tangente continua, aunque la parametrización no sea tenga derivada continua.

Obviamente, las relaciones (1) no se mantienen si permitimos al usuario modificar *todos* los vértices. Por tanto, si queremos mantener la clase de diferenciabilidad, sólo deberemos ofrecerle los vértices que no afecten a dicha condición.

Con esta idea en mente, podemos ver (1) como la definición del vértice c_{2i} , conocidos los vértices c_{2i-1} , c_{2i+1} ,

$$c_{2i} = \frac{\Delta u_i}{\Delta u_{i-1} + \Delta u_i} c_{2i-1} + \frac{\Delta u_{i-1}}{\Delta u_{i-1} + \Delta u_i} c_{2i+1}, \quad (2)$$

que refleja el hecho de que la razón simple de los puntos $c_{2i-1}, c_{2i}, c_{2i+1}$ es precisamente $[c_{2i-1}, c_{2i}, c_{2i+1}] = \Delta u_{i-1} / \Delta u_i$, tal como se refleja en la figura.

Por tanto, conocidos los vértices “interiores”, c_{2i-1} , de cada polígono y los **nudos** de la partición del intervalo, $\{u_0, \dots, u_N\}$, podemos reconstruir los vértices “exteriores”, c_{2i} , de los polígonos por la relación anterior, garantizando de este modo que la parametrización será de clase C^1 .

Por supuesto, a los vértices impares habrá que añadir los vértices que no se obtienen a partir de ninguna condición de diferenciabilidad, c_0, c_{2N} , puesto que no están situados sobre ninguna unión de tramos.

En resumen, los únicos datos de la curva que debemos facilitar al usuario para que los modifique libremente, sin alterar la condición de suavidad, son los nudos de la partición del intervalo, $\{u_0, \dots, u_N\}$ y los vértices de los polígonos de control $\{c_0, c_1, \dots, c_{2i-1}, \dots, c_{2N-1}, c_{2N}\}$, es decir, un total de $N + 2$ vértices, como corresponde al hecho de que tenemos $2N + 1$ vértices y $N - 1$ condiciones de diferenciabilidad en las uniones. [Ejemplo](#).

Vídeo de Ejemplo de tramo parabólico spline

Adelantando notación posterior, denotaremos como $\{d_0, \dots, d_{N+1}\}$ estos vértices,

$$d_0 := c_0, \quad d_i := c_{2i-1}, \quad i = 1, \dots, N, \quad d_{N+1} := c_{2N}. \quad (3)$$

A este conjunto de vértices lo denominaremos **polígono B-spline** de la curva polinómica a trozos.

Obsérvese que con esta representación, usando la forma polar de cada tramo,

$$d_0 = c[u_0, u_0], \quad d_i = c[u_{i-1}, u_i], \quad i = 1, \dots, N, \quad d_{N+1} = c[u_N, u_N].$$

Por tanto, podemos unificar la notación definiendo **nudos auxiliares** $u_{-1} := u_0, u_{N+1} := u_N$, de modo que

$$d_i = c[u_{i-1}, u_i], \quad i = 0, \dots, N + 1,$$

O lo que es lo mismo, los nuevos vértices se evalúan con la forma polar sobre listas correlativas de dos nudos consecutivos extraídas de

$$\{u_0, u_0, u_1, u_2, \dots, u_{N-2}, u_{N-1}, u_N, u_N\}.$$

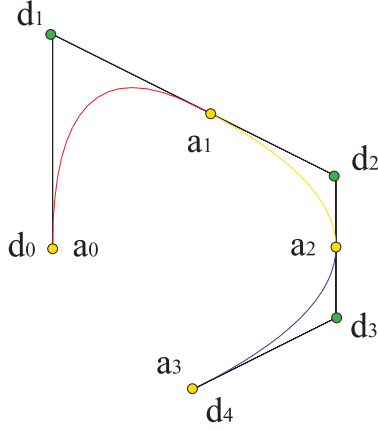


Figura 2: Parábola a trozos que interpola cuatro puntos

Esta construcción de la curva de clase C^1 parabólica a trozos se podría emplear para interpolar una curva de clase C^1 entre $N + 1$ puntos del plano, a_0, \dots, a_N , por los que sabemos que pasa para valores conocidos del parámetro u , $a_i = c(u_i)$. Teniendo en cuenta (1) y (3), el problema se reduce, en función de los vértices $\{d_0, \dots, d_{N+1}\}$, a resolver el sistema de ecuaciones $c_{2i} = a_i$,

$$\begin{aligned}
 a_0 &= d_0 \\
 a_i &= \frac{\Delta u_i}{\Delta u_{i-1} + \Delta u_i} d_i + \frac{\Delta u_{i-1}}{\Delta u_{i-1} + \Delta u_i} d_{i+1}, \quad i = 1, \dots, N-1 \\
 a_N &= d_{N+1},
 \end{aligned} \tag{4}$$

$$\begin{pmatrix} x_1 \\ \vdots \\ x_{N-1} \end{pmatrix} = \begin{pmatrix} \Delta u_1 & \Delta u_0 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \Delta u_{N-1} & \Delta u_{N-2} \end{pmatrix} \begin{pmatrix} d_1 \\ \vdots \\ d_N \end{pmatrix}. \tag{5}$$

que, obviamente es un sistema indeterminado, ya que consta, una vez eliminadas las dos ecuaciones triviales, de $N - 1$ ecuaciones lineales para N incógnitas, los vértices d_1, \dots, d_N . Nos haría falta un dato adicional aparte de los valores $x_i := (\Delta u_{i-1} + \Delta u_i)a_i$.

Esto es lógico, ya que lo que estamos haciendo es trazar parábolas entre cada par de puntos consecutivos a_{i-1} , a_i y una parábola precisa de tres datos para ser determinada. Podemos afirmar que la condición de diferenciabilidad fija un tercer dato, la tangente en uno de los extremos, conocida la tangente del tramo anterior. Lo cual nos lleva a tener que fijar, por ejemplo, una tangente en uno de los tramos. Esta sería una solución un tanto asimétrica del problema.

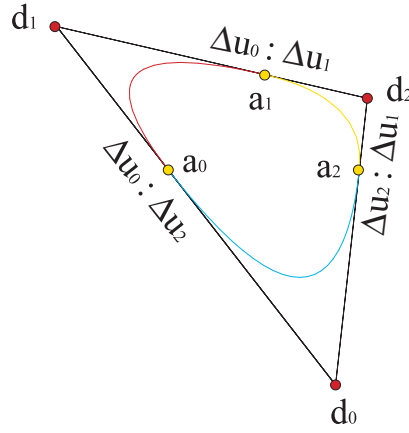


Figura 3: Parábola a trozos cerrada

Obviamente, si la curva es cerrada, $c_0 = c_{2N+2}$, todo vértice par se obtiene a partir de los impares adyacentes por la relación (2) y sólo es preciso facilitar estos últimos, $\{c_1, \dots, c_{2i-1}, \dots, c_{2N+1}\}$. El vértice inicial-final se obtiene como caso particular,

$$\begin{aligned} c_0 &= c_{2N+2} = \frac{\Delta u_0}{\Delta u_N + \Delta u_0} c_{2N+1} + \frac{\Delta u_N}{\Delta u_N + \Delta u_0} c_1 \\ c_{2N} &= \frac{\Delta u_{N-1}}{\Delta u_{N-1} + \Delta u_N} c_{2N+1} + \frac{\Delta u_N}{\Delta u_{N-1} + \Delta u_N} c_{2N-1}. \end{aligned} \quad (6)$$

El polígono B-spline, $\{d_0, \dots, d_N\}$, está formado por los vértices interiores, $d_0 = c_{2N+1}$, $d_i = c_{2i-1}$, $i = 1, \dots, N$. Para el problema de interpolación,

$$c_{2i} = a_i, \quad i = 1, \dots, N,$$

$$\begin{pmatrix} x_0 \\ \vdots \\ x_N \end{pmatrix} = \begin{pmatrix} \Delta u_0 & \Delta u_N & 0 & \cdots & 0 \\ 0 & \Delta u_1 & \Delta u_0 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \ddots & \Delta u_{N-1} & \Delta u_{N-2} \\ \Delta u_{N-1} & 0 & \cdots & 0 & \Delta u_N \end{pmatrix} \begin{pmatrix} d_0 \\ \vdots \\ d_N \end{pmatrix}, \quad (7)$$

donde $x_0 := (\Delta u_N + \Delta u_0)a_0$, $x_N := (\Delta u_{N-1} + \Delta u_N)a_N$.

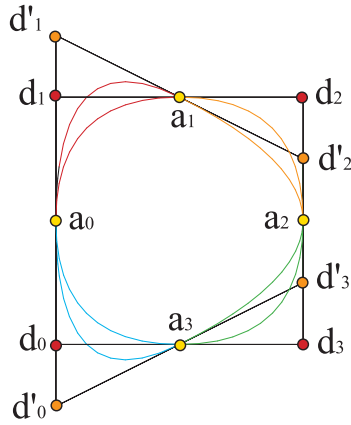


Figura 4: Dos soluciones para el problema de interpolación de cuatro puntos

La matriz de este problema de interpolación presenta un grave problema: es fácil ver que sólo es regular para N par. Luego tenemos garantizada la solución única del problema de interpolación exclusivamente para un número impar de puntos, lo cual no quiere decir que no haya solución en todos los casos. Luego no es un método afortunado de interpolación. [Ejemplo](#). En el caso se dos puntos, por ejemplo, no hay solución, ya que no se pueden enlazar dos parábolas con tangente continua.

Es claro que, como la parábola es una curva plana, esta construcción sólo es válida en el plano. En el espacio lo único que conseguiríamos sería una curva espacial compuesta de sucesivos tramos planos. Por todo ello, pasamos al siguiente ejemplo, que es mucho más interesante.

3. Cúbicas C^2 a trozos

Vídeo de Cúbicas a trozos

Como es conocido, la curva no plana de menor grado es la cúbica. Por ello, para abordar el problema de interpolación en el espacio, debemos recurrir a este tipo de curvas. Buscaremos construir una curva de clase C^2 de N tramos. Tendremos, pues, como vértices de los sucesivos polígonos de control $\{c_0, \dots, c_{3N}\}$, de los cuales $\{c_{3(i-1)}, c_{3i-2}, c_{3i-1}, c_{3i}\}$ corresponden al tramo i -ésimo, definido en el intervalo $[u_{i-1}, u_i]$.

Vamos a imponer que la parametrización sea de clase C^2 en $u = u_i$, unión de los tramos i e $(i+1)$ -ésimo. Como en el apartado anterior, la condición de ser de clase C^1 se traduce en

$$\frac{\Delta c_{3i-1}}{\Delta u_{i-1}} = \frac{\Delta c_{3i}}{\Delta u_i}, \quad i = 1, \dots, N-1, \quad (8)$$

$$c_{3i} = \frac{\Delta u_i}{\Delta u_{i-1} + \Delta u_i} c_{3i-1} + \frac{\Delta u_{i-1}}{\Delta u_{i-1} + \Delta u_i} c_{3i+1}, \quad (9)$$

es decir, los vectores $\mathbf{c}_{3i-1}\mathbf{c}_{3i}$ y $\mathbf{c}_{3i}\mathbf{c}_{3i+1}$ son paralelos y están en proporción $\Delta u_{i-1} : \Delta u_i$.

Finalmente, la condición de continuidad de las derivadas segundas, desarrollada en el tema segundo, aplicada al nudo u_i nos proporciona como información que los vectores $\mathbf{c}_{3i-2}\mathbf{c}_{3i-1}$ y $\mathbf{c}_{3i-1}\mathbf{d}_i$ están en proporción $\Delta u_{i-1} : \Delta u_i$, igual que los vectores $\mathbf{d}_i\mathbf{c}_{3i+1}$ y $\mathbf{c}_{3i+1}\mathbf{c}_{3i+2}$.

Aplicando esta misma condición al nudo u_{i+1} obtenemos otra relación. Los vectores $\mathbf{c}_{3i+1}\mathbf{c}_{3i+2}$ y $\mathbf{c}_{3i+2}\mathbf{d}_{i+1}$ están en proporción $\Delta u_i : \Delta u_{i+1}$. Y del nudo u_{i-1} extraemos que $\mathbf{d}_{i-1}\mathbf{c}_{3i-2}$ y $\mathbf{c}_{3i-2}\mathbf{c}_{3i-1}$ guardan una proporción igual a $\Delta u_{i-2} : \Delta u_{i-1}$.

Con toda esta información, estamos en condiciones de expresar los vértices c_{3i-1}, c_{3i+1} en términos de los nuevos puntos que hemos introducido,

$$\begin{aligned} c_{3i-1} &= \frac{\Delta u_i}{\Delta u_{i-2} + \Delta u_{i-1} + \Delta u_i} d_{i-1} + \frac{\Delta u_{i-2} + \Delta u_{i-1}}{\Delta u_{i-2} + \Delta u_{i-1} + \Delta u_i} d_i, \\ c_{3i+1} &= \frac{\Delta u_i + \Delta u_{i+1}}{\Delta u_{i-1} + \Delta u_i + \Delta u_{i+1}} d_i + \frac{\Delta u_{i-1}}{\Delta u_{i-1} + \Delta u_i + \Delta u_{i+1}} d_{i+1}. \end{aligned} \quad (10)$$

Este resultado, combinado con (9) nos permite obtener todos los vértices de la curva compuesta a partir de los puntos d_1, \dots, d_{N-1} , garantizando que

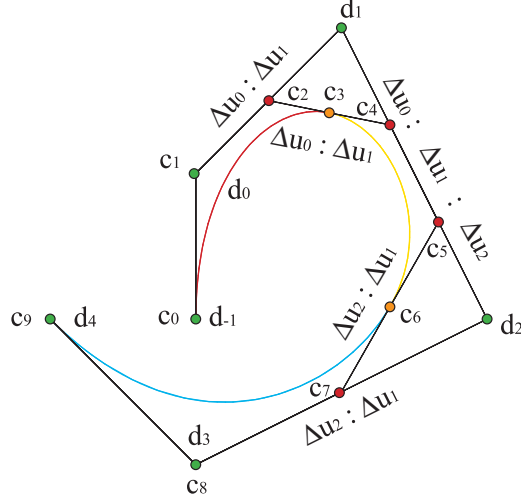


Figura 5: Cúbica a trozos

la parametrización es de clase C^2 . Quedan fuera los vértices que no son adyacentes a ninguna unión, es decir, los vértices iniciales, c_0, c_1 , y finales, c_{3N-1}, c_{3N} , que no pueden obtenerse por estas relaciones.

Denotaremos, como hicimos en el apartado anterior, $d_{-1} := c_0$, $d_0 := c_1$, $d_N := c_{3N-1}$, $d_{N+1} := c_{3N}$, para poder completar la relación de vértices. Los restantes vértices, c_2, c_{3N-2} se obtienen a partir de (10) tomando Δu_{-1} y Δu_N nulos.

Podemos reescribir los nuevos vértices por medio de la forma polar, al igual que hicimos con las parábolas a trozos:

En el nudo u_i , los vértices de la cúbica son

$$c_{3i-2} = c[u_{i-1}, u_{i-1}, \mathbf{u}_i], \quad c_{3i-1} = c[u_{i-1}, u_i, \mathbf{u}_i], \quad c_{3i} = c[u_i, u_i, \mathbf{u}_i],$$

$$c_{3i+1} = c[\mathbf{u}_i, u_i, u_{i+1}], \quad c_{3i+2} = c[\mathbf{u}_i, u_{i+1}, u_{i+1}].$$

Y sobre la parábola auxiliar que hemos usado para la continuidad de las derivadas segundas,

$$c_{3i-2} = b[u_{i-1}, u_{i-1}], \quad c_{3i-1} = b[u_{i-1}, u_i], \quad c_{3i} = b[u_i, u_i],$$

$$c_{3i+1} = b[u_i, u_{i+1}], \quad c_{3i+2} = b[u_{i+1}, u_{i+1}].$$

Por tanto, $d_i = b[u_{i-1}, u_{i+1}] = c[u_{i-1}, \mathbf{u}_i, u_{i+1}]$, $i = 1, \dots, N-1$, puesto que estamos fijando el mismo valor u_i en un paso del algoritmo de De Casteljaun en la cúbica al evaluar todos los vértices implicados, con lo cual se obtienen los mismos resultados que con la parábola.

Como, por su parte,

$$d_{-1} := c_0 = c[u_0, u_0, u_0], \quad d_0 := c_1 = c[u_0, u_0, u_1],$$

$$d_N := c_{3N-1} = c[u_{N-1}, u_N, u_N], \quad d_{N+1} := c_{3N} = c[u_N, u_N, u_N],$$

si introducimos cuatro **nudos auxiliares**

$$u_{-2} := u_0, \quad u_{-1} := u_0, \quad u_{N+1} := u_N, \quad u_{N+2} := u_N,$$

todos los vértices se evaluarán sobre listas de nudos correlativos,

$$d_i = c[u_{i-1}, u_i, u_{i+1}], \quad i = -1, \dots, N+1.$$

O lo que es lo mismo, los nuevos vértices se evalúan con la forma polar sobre listas correlativas de tres nudos consecutivos extraídas de

$$\{u_{-2}, u_{-1}, u_0, u_1, u_2, \dots, u_{N-2}, u_{N-1}, u_N, u_{N+1}, u_{N+2}\}.$$

Realmente, la notación usual es la correlativa de 0 hasta $N+2$, para lo cual deberíamos adelantar los índices en una unidad. Esta será la notación que vamos a seguir.

Una vez más, en el caso de curvas cerradas, no hace falta añadir más puntos y todos los vértices se obtienen a partir de (9) y (10).

Por ello, podemos codificar la curva compuesta por medio del **polígono de control B-spline**, $\{d_0, \dots, d_{N+2}\}$ y la sucesión de nudos $\{u_0, \dots, u_N\}$ de modo que el usuario podrá modificar la forma de la curva sin alterar sus propiedades de diferenciabilidad.

Sustituyendo las expresiones de los vértices c_{3i-1} y c_{3i+1} en (9),

$$\begin{aligned} c_{3i} &= \frac{\alpha_i d_i + \beta_i d_{i+1} + \gamma_i d_{i+2}}{\Delta u_{i-1} + \Delta u_i}, \\ \alpha_i &:= \frac{(\Delta u_i)^2}{\Delta u_{i-2} + \Delta u_{i-1} + \Delta u_i}, \\ \beta_i &:= \left(\Delta u_i \frac{\Delta u_{i-2} + \Delta u_{i-1}}{\Delta u_{i-2} + \Delta u_{i-1} + \Delta u_i} + \Delta u_{i-1} \frac{\Delta u_i + \Delta u_{i+1}}{\Delta u_{i-1} + \Delta u_i + \Delta u_{i+1}} \right), \\ \gamma_i &:= \frac{(\Delta u_{i-1})^2}{\Delta u_{i-1} + \Delta u_i + \Delta u_{i+1}}, \end{aligned} \tag{11}$$

vemos que c_{3i} depende de tres vértices consecutivos del polígono B-spline.

Esta relación es útil para resolver el problema de interpolación: dados $N + 1$ puntos a_0, \dots, a_N y los nudos $\{u_0, \dots, u_N\}$, obtener la curva de clase C^2 cúbica a trozos que verifique $c(u_i) = a_i$. El problema se reduce al sistema lineal (11), donde los datos son $c_0 = d_0 = a_0$, $c_{3i} = a_i$, $i = 1, \dots, N - 1$, $a_N = d_{N+2} = c_{3N}$ y las incógnitas son los vértices del polígono B-spline, $\{d_0, \dots, d_{N+2}\}$. Tenemos, pues, $N + 3$ incógnitas para $N + 1$ datos, así que el sistema tiene dos parámetros libres, por ejemplo, d_1 y d_{N+1} . Se puede demostrar que el sistema tiene rango $N + 1$, así que el problema tiene solución única, una vez introducidos los valores de los vértices libres. El sistema quedaría como,

$$\begin{pmatrix} x_0 \\ \vdots \\ x_N \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \alpha_1 & \beta_1 & \gamma_1 & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \alpha_{N-1} & \beta_{N-1} & \gamma_{N-1} \\ 0 & \cdots & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} d_1 \\ \vdots \\ d_{N+1} \end{pmatrix}, \quad (12)$$

donde estamos usando ya la nueva numeración de vértices y $x_0 := c_1$, $x_i := (\Delta u_{i-1} + \Delta u_i)a_i$, $i = 1, \dots, N - 1$, $x_N := c_{3N-1}$.

Vídeo de Interpolación cúbica

Existen varias maneras de fijar estos vértices, imponiendo condiciones razonables a la curva.

Una posibilidad son las **condiciones naturales**, que imitan el comportamiento físico del junquillo, el cual tiende a ser recto en los extremos. Por tanto, la condición natural es

$$c''(u_0) = 0 = c''(u_N), \quad (13)$$

que podemos desglosar en función de los vértices del polígono de control, ya que la derivada segunda en el origen de la curva es, salvo factores constantes,

$$0 = \Delta^2 c_0 = c_0 - 2c_1 + c_2 = d_0 - 2d_1 + \frac{\Delta u_1}{\Delta u_0 + \Delta u_1} d_1 + \frac{\Delta u_0}{\Delta u_0 + \Delta u_1} d_2,$$

haciendo uso de la relación (10), con la nueva numeración de vértices.

De esta expresión y su análoga para los vértices finales de la curva, obtenemos la condición natural sobre los primeros vértices,

$$\begin{aligned} 0 &= (\Delta u_0 + \Delta u_1)d_0 - (2\Delta u_0 + \Delta u_1)d_1 + \Delta u_0 d_2, \\ 0 &= (\Delta u_{N-1} + \Delta u_{N-2})d_{N+2} - (2\Delta u_{N-1} + \Delta u_{N-2})d_{N+1} + \Delta u_{N-1}d_N. \end{aligned}$$

No obstante, estas condiciones no son muy apreciadas en el diseño, precisamente porque provocan que los extremos de las curvas sean excesivamente rectos.

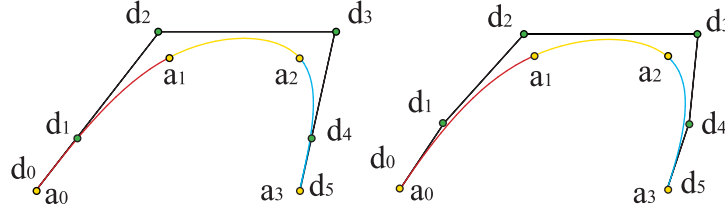


Figura 6: Interpolación con condiciones naturales y con tangentes de Bessel

Otra opción consiste en elegir las tangentes en los extremos, $c'(u_0)$, $c'(u_N)$, que determinan a su vez los vértices d_1 , d_{N+1} ,

$$c'(u_0) = \frac{3}{\Delta u_0}(d_1 - d_0), \quad c'(u_N) = \frac{3}{\Delta u_{N-1}}(d_{N+2} - d_{N+1}),$$

teniendo en cuenta que $d_0 = a_0$, $d_{N+2} = a_N$,

$$d_1 = a_0 + \frac{\Delta u_0}{3}c'(u_0), \quad d_{N+1} = a_N - \frac{\Delta u_{N-1}}{3}c'(u_N), \quad (14)$$

que una vez reemplazadas en el sistema de ecuaciones, lo reducen a un sistema de $N - 1$ ecuaciones para $N - 1$ incógnitas.

Una manera práctica de prescribir las tangentes son las **condiciones de Bessel**, que consisten en tomar como valor de $c'(u_0)$ la derivada en el origen de la parábola que pasa por los puntos a_0 , a_1 , a_2 y, análogamente, $c'(u_N)$ será la derivada en u_N de la parábola que interpola a_{N-2} , a_{N-1} , a_N .

La parametrización de la primera de las parábolas es

$$\hat{c}(u) = \frac{(u_2 - u)^2 \hat{c}_0 + 2(u - u_0)(u_2 - u) \hat{c}_1 + (u - u_0)^2 \hat{c}_2}{(\Delta u_0 + \Delta u_1)^2},$$

donde $\hat{c}_0 = a_0$, $\hat{c}_2 = a_2$ y \hat{c}_1 se determina por la condición $a_1 = \hat{c}(u_1)$

$$\begin{aligned}\hat{c}(u_1) &= \frac{(\Delta u_1)^2 a_0 + 2\Delta u_0 \Delta u_1 \hat{c}_1 + (\Delta u_0)^2 a_2}{(\Delta u_0 + \Delta u_1)^2}, \\ \hat{c}_1 &= \frac{1}{2} \left(\frac{(\Delta u_0 + \Delta u_1)^2}{\Delta u_0 \Delta u_1} a_1 - \frac{\Delta u_1}{\Delta u_0} a_0 - \frac{\Delta u_0}{\Delta u_1} a_2 \right),\end{aligned}$$

con lo cual la derivada en el origen y el vértice d_1 son

$$\begin{aligned}c'(u_0) &:= \hat{c}'(u_0) = \frac{2}{\Delta u_0 + \Delta u_1} (\hat{c}_1 - \hat{c}_0), \\ d_1 &= \frac{2}{3(\Delta u_0 + \Delta u_1)} (\Delta u_1 a_0 + \Delta u_0 \hat{c}_1) + \frac{1}{3} a_0,\end{aligned}$$

y de manera análoga para el extremo final de la curva,

$$\begin{aligned}\hat{c}_{N-1} &= \frac{1}{2} \left(\frac{(\Delta u_{N-2} + \Delta u_{N-1})^2}{\Delta u_{N-1} \Delta u_{N-2}} a_{N-1} - \frac{\Delta u_{N-2}}{\Delta u_{N-1}} a_N - \frac{\Delta u_{N-1}}{\Delta u_{N-2}} a_{N-2} \right), \\ d_{N+1} &= \frac{2}{3(\Delta u_{N-2} + \Delta u_{N-1})} (\Delta u_{N-2} a_N + \Delta u_{N-1} \hat{c}_{N-1}) + \frac{1}{3} a_N.\end{aligned}$$

Las tangentes de Bessel proporcionan resultados mucho mejores que la condición natural. [Ejemplo](#).

En el caso de curvas cerradas, como se ha indicado, no son precisos vértices ni datos adicionales y el sistema se cerraría en forma periódica,

$$\begin{pmatrix} x_0 \\ \vdots \\ x_N \end{pmatrix} = \begin{pmatrix} \alpha_0 & \beta_0 & \gamma_0 & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \alpha_{N-2} & \beta_{N-2} & \gamma_{N-2} \\ \gamma_{N-1} & \ddots & 0 & \alpha_{N-1} & \beta_{N-1} \\ \beta_N & \gamma_N & 0 & \ddots & \alpha_N \end{pmatrix} \begin{pmatrix} d_0 \\ \vdots \\ d_N \end{pmatrix}, \quad (15)$$

donde ahora $x_0 := (\Delta u_N + \Delta u_0) a_0$, $x_N := (\Delta u_{N-1} + \Delta u_N) a_N$.

4. Cúbicas C^1 a trozos

Un problema alternativo al anterior sería la interpolación con cúbicas, conocidos los puntos de la curva $c(u_i)$ y las tangentes a la curva, de clase C^1 al menos en dichos puntos.

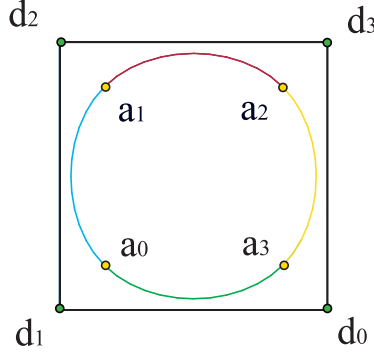


Figura 7: Interpolación con cúbica a trozos cerrada

Supongamos que tenemos $N + 1$ puntos $\{a_0, \dots, a_N\}$ y $N + 1$ vectores $\{v_0, \dots, v_N\}$ y queremos trazar una curva interpolante $c(u)$ que verifique $c(u_i) = a_i$, $c'(u_i) = v_i$, $i = 0, \dots, N$ para una sucesión creciente de valores del parámetro u , $\{u_0, \dots, u_N\}$.

Una solución sencilla para este problema nos la proporciona una cúbica a trozos de clase C^1 con N tramos. Al igual que en el caso anterior, la curva requiere $3N + 1$ vértices $\{c_0, \dots, c_{3N}\}$, de los cuales $N + 1$, los extremos de las cúbicas, están determinados por las condiciones $c_{3i} = a_i$, $i = 0, \dots, N$. El resto están determinados por las tangentes.

Teniendo en cuenta las expresiones para las derivadas en los extremos de cada cúbica, estudiadas previamente,

$$v_i = c'(u_i) = 3 \frac{c_{3i+1} - c_{3i}}{\Delta u_i}, \quad i = 0, \dots, N - 1,$$

$$v_i = c'(u_i) = 3 \frac{c_{3i} - c_{3i-1}}{\Delta u_{i-1}}, \quad i = 1, \dots, N,$$

respectivamente para el extremo inicial y final de cada tramo. De aquí podemos despejar los vértices interiores de cada polígono,

$$c_{3i+1} = c_{3i} + \frac{\Delta u_i}{3} v_i, \quad i = 0, \dots, N - 1,$$

$$c_{3i-1} = c_{3i} - \frac{\Delta u_{i-1}}{3} v_i, \quad i = 1, \dots, N, \quad (16)$$

así que en este caso no hay grados de libertad adicionales y la curva queda determinada unívocamente.

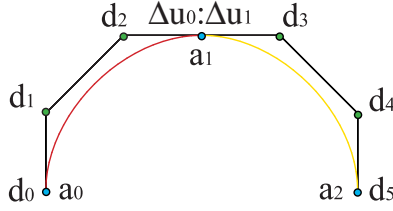


Figura 8: Interpolación C^1 con cúbicas a trozos

Sin embargo, para codificar la información de la curva, no es preciso una vez más dar todos los vértices, sino tan sólo los interiores, ya que los vértices exteriores se despejan de las relaciones anteriores,

$$c_{3i} = \frac{\Delta u_{i-1} c_{3i+1} + \Delta u_i c_{3i-1}}{\Delta u_{i-1} + \Delta u_i}, \quad i = 1, \dots, N-1, \quad (17)$$

salvo los vértices inicial y final, c_0, c_{3N} . Por tanto, podemos caracterizar la curva por la sucesión de nudos $\{u_0, \dots, u_N\}$ y los vértices del **polígono B-spline**, $\{d_0, \dots, d_{2N+1}\}$, dado por $d_0 := c_0, d_{2i-1} = c_{3i-2}, d_{2i} = c_{3i-1}, i = 1, \dots, N, d_{2N+1} = c_{3N}$,

$$c_{3i} = \frac{\Delta u_{i-1} d_{2i+1} + \Delta u_i d_{2i}}{\Delta u_{i-1} + \Delta u_i}, \quad i = 1, \dots, N-1. \quad (18)$$

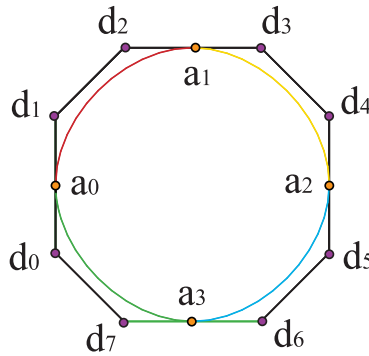


Figura 9: Interpolación C^1 cerrada con cúbicas a trozos

Escribimos los nuevos vértices con la forma polar,

$$d_0 = c[u_0, u_0, u_0], \quad d_1 = c[u_0, u_0, u_1], \quad d_2 = c[u_0, u_1, u_1], \quad d_3 = c[u_1, u_1, u_2],$$

$$d_{2N-1} = c[u_{N-1}, u_{N-1}, u_N], \quad d_{2N} = c[u_{N-1}, u_N, u_N], \quad d_{2N+1} = c[u_N, u_N, u_N],$$

y observamos que se obtienen evaluándola sobre listas correlativas de tres nudos consecutivos tomados de

$$\{u_0, u_0, u_0, u_1, u_1, u_2, u_2, \dots, u_{N-1}, u_{N-1}, u_N, u_N, u_N\}.$$

El caso de curvas cerradas que interpolen los $N + 1$ puntos y tangentes es aún más sencillo: las expresiones (16-18) son válidas también para los casos extremos $i = 0, N$ sin más que tomar $\Delta u_{-1} := \Delta u_N$, para lo cual necesitamos un nudo más, u_{N+1} , el que cierra la curva mediante las condiciones $a_0 = c(u_{N+1})$, $v_0 = c'(u_{N+1})$.

Este problema de interpolación se puede generalizar dando los valores de las derivadas segundas de la parametrización, para lo cual recurriremos a curvas cuánticas de clase C^2 a trozos.

5. Curvas B-spline

Vídeo de Curvas B-spline

De los ejemplos anteriores podemos inferir ciertas propiedades de las curvas de grado n de clase C^{n-1} a trozos con N tramos:

- Se definen con un polígono B-spline de $n + N$ vértices $\{d_0, \dots, d_{n+N-1}\}$ y una lista de $2n + N - 1$ nudos $\{u_0, \dots, u_{2n+N-2}\}$.
- La curva está definida en el intervalo de N tramos $[u_{n-1}, u_{n+N-1}]$.
- Los $n - 1$ primeros nudos y los últimos son nudos auxiliares y se toman respectivamente iguales a u_{n-1} y a u_{n+N-1} .
- Es decir, la lista de nudos comienza con n nudos iguales $u_0 = \dots = u_{n-1}$ y acaba con otros n nudos iguales $u_{n+N-1} = \dots = u_{2n+N-2}$.
- Los vértices se definen a partir de la forma polar evaluada sobre listas de nudos correlativos, $d_i = c[u_i, \dots, u_{i+n-1}]$.

- La clase de diferenciabilidad en un nudo interior u_i baja a C^{n-r} si dicho nudo está repetido r veces. De esta propiedad nos ocuparemos más adelante.

La primera propiedad es sencilla de comprobar, ya que una curva de grado n y N tramos tiene polígonos de control por un total de $1 + nN$ vértices, $\{c_0, \dots, c_{nN}\}$. Si imponemos que la curva sea de clase C^{n-1} , tenemos $n - 1$ condiciones en las $N - 1$ uniones de los tramos. Por tanto, el número de vértices libres resultante es

$$1 + nN - (n - 1)(N - 1) = n + N.$$

La curva a trozos no puede ser de clase C^n o superior, ya que un polinomio de grado n está determinado por sus n derivadas. Por tanto, si estas n derivadas fuesen iguales en todas las uniones, la curva sería en realidad de un único tramo, ya que las funciones polinómicas serían las mismas en todos los tramos. De hecho, la cuenta anterior daría un total de $n + 1$ vértices libres, los que corresponden a una curva de grado n y un único tramo.

El resto de propiedades son consecuencia de definir los vértices como $d_i = c[u_i, \dots, u_{i+n-1}]$, manteniendo los vértices inicial y final,

$$d_0 = c_0 = c[u_{n-1}^{<n>}], \quad d_{n+N-1} = c_{nN} = c[u_{n+N-1}^{<n>}],$$

lo que obliga a introducir los nudos auxiliares.

Queda por ver cómo expresar el algoritmo de De Casteljaou con polígonos B-spline y nudos, ya que resultaría engorroso tener que pasar de nuevo por los polígonos de control de cada tramo, vía forma polar, para poder aplicar el algoritmo.

Para ilustrarlo, comenzamos con un parábola de un sólo tramo, con polígono B-spline $\{d_0, d_1, d_2\}$ y nudos $\{u_0, u_1, u_2, u_3\}$. El algoritmo de De Casteljaou tiene dos iteraciones,

$$\begin{array}{l} d_0 = c[u_0, u_1] \quad d_1 = c[u_1, u_2] \quad d_2 = c[u_2, u_3] \\ d_0^{(1)}(u) \quad \quad \quad d_1^{(1)}(u) \quad \quad \quad , \\ d_0^{(2)}(u) = c[u, u] \end{array}$$

de las cuales desconocemos la expresión, pero sabemos que son afines en u y que deben ser combinaciones baricéntricas de dos vértices consecutivos. Por ejemplo,

$$d_0^{(1)}(u) = \lambda d_0 + (1 - \lambda) d_1 = \lambda c[u_0, u_1] + (1 - \lambda) c[u_1, u_2] = c[u_1, \lambda u_0 + (1 - \lambda) u_2],$$

por la simetría y multiafinidad de la forma polar.

Por otra parte, como en cada iteración el algoritmo sustituye un nudo por una u , la única posibilidad es que $d_0^1(u) = c[u_1, u]$. Comparando ambas expresiones, deducimos que

$$u = \lambda u_0 + (1 - \lambda)u_2.$$

Como a su vez λ tiene que ser afín en u , $\lambda = au + b$, la única solución posible es

$$\lambda = \frac{u_2 - u}{u_2 - u_0} \Rightarrow d_0^1(u) = \frac{u_2 - u}{u_2 - u_0}d_0 + \frac{u - u_0}{u_2 - u_0}d_1,$$

y, del mismo modo, trasladando una unidad los índices,

$$d_1^1(u) = c[u, u_2] = \frac{u_3 - u}{u_3 - u_1}d_1 + \frac{u - u_1}{u_3 - u_1}d_2.$$

De idéntica manera, en la segunda iteración,

$$\begin{aligned} d_0^2(u) &= \mu d_1^1(u) + (1 - \mu)d_1^1(u) = \mu c[u_1, u] + (1 - \mu)c[u, u_2] \\ &= c[u, \mu u_1 + (1 - \mu)u_2], \end{aligned}$$

y, a su vez, $d_0^2(u) = c(u) = c[u, u]$.

Por tanto, razonando igual que en el paso anterior,

$$u = \mu u_1 + (1 - \mu)u_2 = \frac{u_2 - u}{u_2 - u_1}u_1 + \frac{u - u_1}{u_2 - u_1}u_2,$$

$$d_0^2(u) = \frac{u_2 - u}{u_2 - u_1}d_0^1(u) + \frac{u - u_1}{u_2 - u_1}d_1^1(u),$$

$$\begin{aligned} c(u) &= \frac{u_2 - u}{u_2 - u_1} \left(\frac{u_2 - u}{u_2 - u_0}c[u_0, u_1] + \frac{u - u_0}{u_2 - u_0}c[u_1, u_2] \right) \\ &+ \frac{u - u_1}{u_2 - u_1} \left(\frac{u_3 - u}{u_3 - u_1}c[u_1, u_2] + \frac{u - u_1}{u_3 - u_1}c[u_2, u_3] \right). \end{aligned} \quad (19)$$

Resumiendo, los pasos seguidos en el algoritmo han sido:

$$\begin{aligned} r = 0) & \quad c[u_0, u_1], c[u_1, u_2], c[u_2, u_3] \\ r = 1) & \quad c[u_1, u], c[u, u_2] \\ r = 2) & \quad c[u, u], \end{aligned} \quad (20)$$

donde cada paso consiste en la interpolación en u en el intervalo formado por los nudos distintos de los vértices correlativos. [Ejemplo](#). Este es el **algoritmo de De Boor** para una curva parabólica de un único tramo. [Ejemplo](#).

Por ejemplo, para un polígono B-spline $\{(0, 0), (0, 1), (1, 1)\}$ y una sucesión de nudos $\{0, 1, 2, 3\}$, el valor de la parametrización correspondiente a $u = 2$ es

$$c(2) = \frac{1}{2}d_1 + \frac{1}{2}d_2 = (1/2, 1).$$

El algoritmo se generaliza sin dificultad a curvas de cualquier grado, n , y un solo tramo:

En la primera iteración, de cada pareja de vértices consecutivos, $d_i = c[u_i, \dots, u_{i+n-1}]$, $d_{i+1} = c[u_{i+1}, \dots, u_{i+n}]$, con $n - 1$ nudos $\{u_{i+1}, \dots, u_{i+n-1}\}$ comunes desaparecen los nudos no comunes, u_i , u_{i+n} , y aparece u por interpolación,

$$u = \frac{u_{i+n} - u}{u_{i+n} - u_i}u_i + \frac{u - u_i}{u_{i+n} - u_i}u_{i+n},$$

para dar el nuevo vértice $d_i^{(1)}(u)$, $i = 0, \dots, n - 1$,

$$d_i^{(1)}(u) = c[u_{i+1}, \dots, u_{i+n-1}, u] = \frac{u_{i+n} - u}{u_{i+n} - u_i}d_i + \frac{u - u_i}{u_{i+n} - u_i}d_{i+1}.$$

El esquema se repite para el resto de iteraciones, $r = 1, \dots, n$: de la pareja de vértices,

$$d_i^{(r)} = c[u_{i+r}, \dots, u_{i+n-1}, u^{<r>}], \quad d_{i+1}^{(r)} = c[u_{i+r+1}, \dots, u_{i+n}, u^{<r>}],$$

desaparecen los nudos no comunes, u_{i+r} , u_{i+n} , y aparece u por interpolación,

$$u = \frac{u_{i+n} - u}{u_{i+n} - u_{i+r}}u_{i+r} + \frac{u - u_{i+r}}{u_{i+n} - u_{i+r}}u_{i+n},$$

para dar el nuevo vértice $d_i^{(r+1)}(u)$, $i = 0, \dots, n - r - 1$,

$$\begin{aligned} d_i^{(r+1)}(u) &= c[u_{i+r+1}, \dots, u_{i+n-1}, u^{<r+1>}] \\ &= \frac{u_{i+n} - u}{u_{i+n} - u_{i+r}}d_i^{(r)}(u) + \frac{u - u_{i+r}}{u_{i+n} - u_{i+r}}d_{i+1}^{(r)}(u). \end{aligned}$$

Recapitulando, el algoritmo de De Boor consiste en la aplicación reiterada

del algoritmo de De Casteljaou,

$$\begin{aligned}
d_i^1(u) &:= c[u_{i+1}, \dots, u_{i+n-1}, u], \quad i = 0, \dots, n-1, \\
&= \frac{u_{i+n} - u}{u_{i+n} - u_i} d_i + \frac{u - u_i}{u_{i+n} - u_i} d_{i+1}, \\
d_i^r(u) &:= c[u_{i+r}, \dots, u_{i+n-1}, u^{<r>}], \quad i = 0, \dots, n-r, \quad r = 1, \dots, n, \\
&= \frac{u_{i+n} - u}{u_{i+n} - u_{i+r-1}} d_i^{r-1}(u) + \frac{u - u_{i+r-1}}{u_{i+n} - u_{i+r-1}} d_{i+1}^{r-1}(u), \\
d_0^n(u) &:= c[u^{<n>}] = \frac{u_n - u}{u_n - u_{n-1}} d_0^{n-1}(u) + \frac{u - u_{n-1}}{u_n - u_{n-1}} d_1^{n-1}(u). \quad (21)
\end{aligned}$$

Por supuesto, $c(u) = d_0^n(u)$. La parametrización está definida en el intervalo final, $[u_{n-1}, u_n]$. [Ejemplo](#).

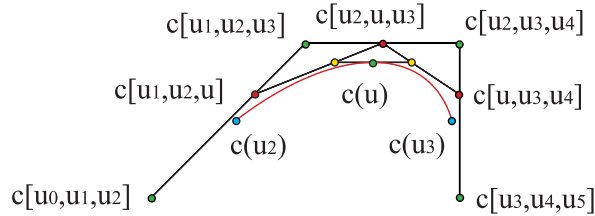
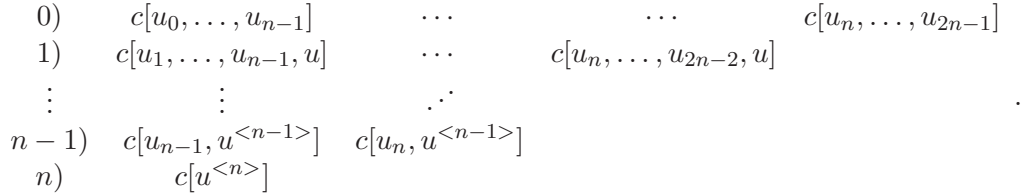


Figura 10: Algoritmo de De Boor para una cúbica

A pesar de su aparente complejidad, este algoritmo es recursivo. En el primer paso interpolamos en el intervalo formado por los nudos no comunes de dos vértices consecutivos para el valor u del parámetro y eliminamos los nudos primero y último para quedarnos con $\{u_1, \dots, u_{2n-2}\}$ y n vértices, $\{d_0^1(u), \dots, d_{n-1}^1(u)\}$. Sobre estos volvemos a aplicar el mismo procedimiento, hasta un total de n iteraciones. Tras la iteración r -ésima, nos quedamos con los nudos $\{u_r, \dots, u_{2n-r-1}\}$ y $n-r$ vértices, $\{d_0^r(u), \dots, d_{n-r}^r(u)\}$.

6. Forma polar

La forma polar correspondiente al algoritmo de De Boor se construye de manera trivial. De hecho, hemos hecho uso de ella para deducir el propio algoritmo. Al igual que sucedía con el algoritmo de De Casteljau, consiste en interpolar en cada iteración del algoritmo para un valor, en general, distinto del parámetro u . Para el polígono correspondiente a una curva de grado n , $\{d_0, \dots, d_n\}$, y una sucesión de nudos, $\{u_0, \dots, u_{2n-1}\}$, la forma polar se evalúa en los valores v_1, \dots, v_n del parámetro,

$$\begin{aligned}
d_i^{(1)}[v_1] &:= c[u_{i+1}, \dots, u_{i+n-1}, v_1], \\
&= \frac{u_{i+n} - v_1}{u_{i+n} - u_i} d_i + \frac{v_1 - u_i}{u_{i+n} - u_i} d_{i+1}, \quad i = 0, \dots, n-1, \\
d_i^{(r)}[v_1, \dots, v_r] &:= c[u_{i+r}, \dots, u_{i+n-1}, v_1, \dots, v_r] \\
&= \frac{u_{i+n} - v_r}{u_{i+n} - u_{i+r-1}} d_i^{(r-1)}[v_1, \dots, v_{r-1}] \\
&\quad + \frac{v_r - u_{i+r-1}}{u_{i+n} - u_{i+r-1}} d_{i+1}^{(r-1)}[v_1, \dots, v_{r-1}], \\
i &= 0, \dots, n-r, \quad r = 1, \dots, n, \\
d[v_1, \dots, v_n] &:= d_0^{(n)}[v_1, \dots, v_n] = \frac{u_n - v_n}{u_n - u_{n-1}} d_0^{(n-1)}[v_1, \dots, v_{n-1}] \\
&\quad + \frac{v_n - u_{n-1}}{u_n - u_{n-1}} d_1^{(n-1)}[v_1, \dots, v_{n-1}] = c[v_1, \dots, v_n]. \quad (22)
\end{aligned}$$

A pesar de su aparente asimetría, la forma polar sigue siendo simétrica, como se puede comprobar fácilmente para $n = 2$,

$$\begin{aligned}
d[v_1, v_2] &= c[v_1, v_2] = \frac{u_2 - v_2}{u_2 - u_1} c[v_1, u_1] + \frac{v_2 - u_1}{u_2 - u_1} c[v_1, u_2] \\
&= \frac{u_2 - v_2}{u_2 - u_1} \left(\frac{u_2 - v_1}{u_2 - u_0} c[u_0, u_1] + \frac{v_1 - u_0}{u_2 - u_0} c[u_1, u_2] \right) \\
&\quad + \frac{v_2 - u_1}{u_2 - u_1} \left(\frac{u_3 - v_1}{u_3 - u_1} c[u_1, u_2] + \frac{v_1 - u_1}{u_3 - u_1} c[u_2, u_3] \right),
\end{aligned}$$

donde los términos de $c[u_0, u_1]$, $c[u_2, u_3]$ son claramente simétricos en el intercambio $v_1 \leftrightarrow v_2$. El término menos aparente, $c[u_1, u_2]$ es también simétrico, aunque requiere un pequeño cálculo.

Como ejemplo, calculamos la forma polar para $\{d_0, d_1, d_2, d_3\}$ y sucesión de nudos $\{u_0, u_1, u_2, u_3, u_4, u_5\}$, evaluándola en $\{v_1, v_2, v_3\}$. Obviamente,

$$d_0 = c[u_0, u_1, u_2], \quad d_1 = c[u_1, u_2, u_3], \quad d_2 = c[u_2, u_3, u_4], \quad d_3 = c[u_3, u_4, u_5],$$

$$\begin{aligned}
d[v_1, v_2, v_3] &= c[v_1, v_2, v_3] = \frac{u_3 - v_3}{u_3 - u_2} c[v_1, v_2, u_2] + \frac{v_3 - u_2}{u_3 - u_2} c[v_1, v_2, u_3] \\
&= \frac{u_3 - v_3}{u_3 - u_2} \left(\frac{u_3 - v_2}{u_3 - u_1} c[v_1, u_1, u_2] + \frac{v_2 - u_1}{u_3 - u_1} c[v_1, u_2, u_3] \right) \\
&+ \frac{v_3 - u_2}{u_3 - u_2} \left(\frac{u_4 - v_2}{u_4 - u_2} c[v_1, u_2, u_3] + \frac{v_2 - u_2}{u_4 - u_2} c[v_1, u_3, u_4] \right) \\
&= \frac{u_3 - v_3}{u_3 - u_2} \left(\frac{u_3 - v_2}{u_3 - u_1} \left(\frac{u_3 - v_1}{u_3 - u_0} d_0 + \frac{v_1 - u_0}{u_3 - u_0} d_1 \right) \right. \\
&+ \left. \frac{v_2 - u_1}{u_3 - u_1} \left(\frac{u_4 - v_1}{u_4 - u_1} d_1 + \frac{v_1 - u_1}{u_4 - u_1} d_2 \right) \right) \\
&+ \frac{v_3 - u_2}{u_3 - u_2} \left(\frac{u_4 - v_2}{u_4 - u_2} \left(\frac{u_4 - v_1}{u_4 - u_1} d_1 + \frac{v_1 - u_1}{u_4 - u_1} d_2 \right) \right. \\
&+ \left. \frac{v_2 - u_2}{u_4 - u_2} \left(\frac{u_5 - v_1}{u_5 - u_2} d_2 + \frac{v_1 - u_2}{u_5 - u_2} d_3 \right) \right).
\end{aligned}$$

Obviamente, la forma polar devuelve los valores correctos de los vértices del polígono de control. [Ejemplo.](#) [Ejemplo.](#)

$$d[u_i, \dots, u_{i+n-1}] = c[u_i, \dots, u_{i+n-1}] = d_i.$$

En el fondo, lo único que hemos hecho al introducir el algoritmo de De Boor para curvas polinómicas de un sólo tramo es cambiar el conjunto de puntos con los cuales calculábamos la forma polar. Pasamos de $\{c_0, \dots, c_n\}$ para el algoritmo de De Casteljaou a $\{d_0, \dots, d_n\}$ con el algoritmo de De Boor. Pero la forma polar sigue siendo la misma.

Por tanto, dado que $d_i = c[u_i, \dots, u_{i+n-1}]$ y $c_i = c[u_{n-1}^{<n-i>}, u_n^{<i>}]$, vemos que la forma polar nos proporciona un sencillo mecanismo para pasar de un esquema a otro. Por ejemplo, si tenemos el polígono B-spline de una curva de grado n , $\{d_0, \dots, d_n\}$, su polígono de control, como curva de Bézier vendrá dado simplemente por

$$c_i = c[u_{n-1}^{<n-i>}, u_n^{<i>}] = d[u_{n-1}^{<n-i>}, u_n^{<i>}].$$

Por ejemplo, la curva parabólica de la figura tiene polígono B-spline $\{(0, 0), (1, 1), (2, 0)\}$ y nudos $\{0, 1, 2, 3\}$. En la representación de Bézier le corresponde un polígono de control $\{(1/2, 1/2), (1, 1), (3/2, 1/2)\}$. Partimos de la expresión, teniendo en cuenta que la curva está definida en el intervalo $[1, 2]$, $u_0 = 0$, $u_1 = 1$, $u_2 = 2$, $u_3 = 3$,

$$d[v_1, v_2] = (2 - v_2) \left(\frac{2 - v_1}{2} d_0 + \frac{v_1}{2} d_1 \right) + (v_2 - 1) \left(\frac{3 - v_1}{2} d_1 + \frac{v_1 - 1}{2} d_2 \right),$$

$$\begin{aligned}
c_0 &= d[1, 1] = \frac{1}{2}d_0 + \frac{1}{2}d_1 = (1/2, 1/2), \\
c_1 &= d[1, 2] = d_1 = (1, 1), \\
c_2 &= d[2, 2] = \frac{1}{2}d_1 + \frac{1}{2}d_2 = (3/2, 1/2).
\end{aligned}$$

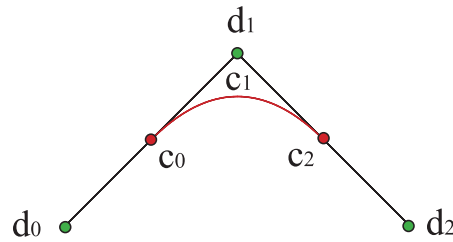


Figura 11: Polígonos B-spline y de control para una curva parabólica

A este nivel puede parecer un esfuerzo innecesario el que estamos realizando al cambiar de representación, pero el apartado siguiente nos permitirá avanzar hacia las curvas de varios tramos.

7. Algoritmo de De Boor

[Vídeo de Algoritmo de De Boor](#)

El algoritmo de De Boor se puede extender a curvas compuestas de N tramos polinómicos de grado n , que es para lo que estaba ideado inicialmente. Consideremos una colección de vértices $\{d_0, \dots, d_{n+N-1}\}$ y una sucesión de nudos $\{u_0, \dots, u_{2n+N-2}\}$. El primer tramo de curva está parametrizado en el intervalo $[u_{n-1}, u_n]$, de acuerdo con el algoritmo ya estudiado, por los vértices $\{d_0, \dots, d_n\}$ y los nudos $\{u_0, \dots, u_{2n-1}\}$. El segundo tramo corresponde a los vértices $\{d_1, \dots, d_{n+1}\}$ y a los nudos $\{u_1, \dots, u_{2n}\}$ y está definido en $[u_n, u_{n+1}]$. Y así sucesivamente, el tramo i -ésimo tiene por polígono $\{d_{i-1}, \dots, d_{n+i-1}\}$, sucesión de nudos $\{u_{i-1}, \dots, u_{2n+i-2}\}$ y su intervalo de definición es $[u_{n+i-2}, u_{n+i-1}]$. Finalmente, el último tramo de curva tiene por polígono $\{d_{N-1}, \dots, d_{n+N-1}\}$, nudos $\{u_{N-1}, \dots, u_{2n+N-2}\}$ y está definido en $[u_{n+N-2}, u_{n+N-1}]$. **Ejemplo.**

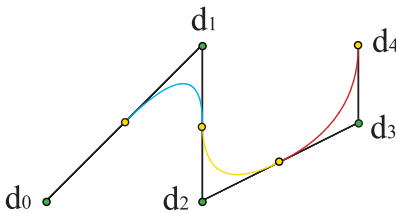


Figura 12: Curva spline parabólica de tres tramos. El nudo final está repetido

Así pues, una curva de N tramos y grado n tiene $n + N$ vértices y $2n + N - 1$ nudos y está definida en el intervalo $[u_{n-1}, u_{n+N-1}]$. Los $n - 1$ nudos iniciales y finales son auxiliares. Lo normal es tomarlos iguales al nudo inicial o final, respectivamente, $u_0 = \dots = u_{n-1}$, $u_{n+N-1} = \dots = u_{2n+N-2}$, tal como hicimos, sin indicarlo, en los apartados precedentes, donde no aparecían tales nudos auxiliares. El motivo no es otro que forzar a la curva a pasar por los vértices inicial y final, $d_0 = c(u_{n-1})$, $d_{n+N-1} = c(u_{n+N-1})$. Así pues, lo común es que la sucesión de nudos comience, y acabe, con un nudo repetido n veces.

Como es fácil de imaginar, la repetición de un nudo no auxiliar, por ejemplo, $u_I = u_{I+1}$, supone la desaparición del tramo de curva correspondiente, en este caso el $(I - n + 1)$ -ésimo, que queda reducido a un punto. También supone, como veremos, rebajar en una unidad la clase de diferenciabilidad de la curva en dicha unión. Por ello, no tiene mucho sentido repetir más de n veces el nudo inicial o el final, ya que simplemente estaríamos acortando la curva por uno de los extremos, a la vez que almacenando información inútil en la lista de vértices. Además, el algoritmo (21) fracasaría, ya que tendríamos un denominador nulo.

Sólo resta, pues, saber cómo aplicar el algoritmo de De Boor a cada uno de los tramos para construir la curva compuesta.

Supongamos que queremos evaluar la curva en un valor u del parámetro. Tendremos que averiguar en qué intervalo $[u_I, u_{I+1})$ está contenido y aplicar el algoritmo de De Boor al polígono correspondiente a ese tramo, $\{\tilde{d}_0 := d_{I-n+1}, \dots, \tilde{d}_n := d_{I+1}\}$ y a los nudos $\tilde{u}_0 := u_{I-n+1}, \dots, \tilde{u}_{2n-1} := u_{I+n}$, tal como se explica en (21).

Por ejemplo, si queremos evaluar el spline cúbico de dos tramos dados por el polígono $\{(0, 0), (0, 1), (1, 1), (2, 0), (1, 0)\}$ y la sucesión de nudos $\{0, 0, 0, 1, 2, 2, 2\}$ para $u = 1/2$, tendremos que tener en cuenta que el valor de u está en el primer tramo, $[0, 1]$ y, por tanto, tenemos que aplicar el al-

goritmo a los vértices d_0, d_1, d_2, d_3 y a los seis primeros nudos, 0, 0, 0, 1, 2, 2,

$$\begin{aligned}
c(1/2) &= \frac{1-1/2}{1-0} \left(\frac{1-1/2}{1-0} \left(\frac{1-1/2}{1-0} d_0 + \frac{1/2-0}{1-0} d_1 \right) \right. \\
&+ \left. \frac{1/2-0}{1-0} \left(\frac{2-1/2}{2-0} d_1 + \frac{1/2-0}{2-0} d_2 \right) \right) \\
&+ \frac{1/2-0}{1-0} \left(\frac{2-1/2}{2-0} \left(\frac{2-1/2}{2-0} d_1 + \frac{1/2-0}{2-0} d_2 \right) \right. \\
&+ \left. \frac{1/2-0}{2-0} \left(\frac{2-1/2}{2-0} d_2 + \frac{1/2-0}{2-0} d_3 \right) \right) \\
&= \frac{1}{8} d_0 + \frac{19}{32} d_1 + \frac{1}{4} d_2 + \frac{1}{32} d_3 = (5/16, 27/32).
\end{aligned}$$

En cambio, si queremos evaluar el spline en $u = 3/2$, estaremos en el segundo tramo, $[1, 2]$, y tendremos que emplear los vértices d_1, d_2, d_3, d_4 y los seis últimos nudos, 0, 0, 1, 2, 2, 2,

$$\begin{aligned}
c(3/2) &= \frac{2-3/2}{2-1} \left(\frac{2-3/2}{2-0} \left(\frac{2-3/2}{2-0} d_0 + \frac{3/2-0}{2-0} d_1 \right) \right. \\
&+ \left. \frac{3/2-0}{2-0} \left(\frac{2-3/2}{2-0} d_1 + \frac{3/2-0}{2-0} d_2 \right) \right) \\
&+ \frac{3/2-1}{2-1} \left(\frac{2-3/2}{2-1} \left(\frac{2-3/2}{2-0} d_1 + \frac{3/2-0}{2-0} d_2 \right) \right. \\
&+ \left. \frac{3/2-1}{2-1} \left(\frac{2-3/2}{2-1} d_2 + \frac{3/2-1}{2-1} d_3 \right) \right) \\
&= \frac{1}{32} d_0 + \frac{1}{4} d_1 + \frac{19}{32} d_2 + \frac{1}{8} d_3 = (25/16, 9/32).
\end{aligned}$$

Y si queremos evaluar la forma polar, por ejemplo, para calcular el polígono de control de uno de los tramos, el procedimiento es el mismo. Sólo hay que establecer cuál es el tramo que nos interesa (aquel cuyo polígono queremos determinar, por ejemplo) y aplicar el algoritmo del apartado anterior.

Por ejemplo, si queremos obtener los polígonos de control del spline cúbico de dos tramos dado por el polígono $\{(0, 0), (0, 1), (1, 1), (2, 0), (1, 0)\}$ y la sucesión de nudos $\{0, 0, 0, 1, 2, 2, 2\}$, para el primer tramo, $[0, 1]$, aplicaremos (23) a los vértices d_0, d_1, d_2, d_3 y a los nudos 0, 0, 0, 1, 2, 2,

$$\begin{aligned}
c_0 &= d[0, 0, 0] = (0, 0), & c_1 &= d[0, 0, 1] = (0, 1), \\
c_2 &= d[0, 1, 1] = (1/2, 1), & c_3 &= d[1, 1, 1] = (1, 3/4),
\end{aligned}$$

mientras que para el segundo tramo, $[1, 2]$, aplicaremos la expresión de la forma polar a los vértices d_1, d_2, d_3, d_4 y a los nudos $0, 0, 1, 2, 2, 2$, es decir, desplazados los índices una unidad,

$$\begin{aligned} d[v_1, v_2, v_3] &= c[v_1, v_2, v_3] = \frac{u_4 - v_3}{u_4 - u_3} \left(\frac{u_4 - v_2}{u_4 - u_2} \left(\frac{u_4 - v_1}{u_4 - u_1} d_1 + \frac{v_1 - u_1}{u_4 - u_1} d_2 \right) \right. \\ &+ \frac{v_2 - u_2}{u_4 - u_2} \left(\frac{u_5 - v_1}{u_5 - u_2} d_2 + \frac{v_1 - u_2}{u_5 - u_2} d_3 \right) \\ &+ \frac{v_3 - u_3}{u_4 - u_3} \left(\frac{u_5 - v_2}{u_5 - u_3} \left(\frac{u_5 - v_1}{u_5 - u_2} d_2 + \frac{v_1 - u_2}{u_5 - u_2} d_3 \right) \right. \\ &\left. \left. + \frac{v_2 - u_3}{u_5 - u_3} \left(\frac{u_6 - v_1}{u_6 - u_3} d_3 + \frac{v_1 - u_3}{u_6 - u_3} d_4 \right) \right) \right), \end{aligned}$$

$$\begin{aligned} \hat{c}_0 &= d[1, 1, 1] = (1, 3/4), & \hat{c}_1 &= d[1, 1, 2] = (3/2, 1/2), \\ \hat{c}_2 &= d[1, 2, 2] = (2, 0), & \hat{c}_3 &= d[2, 2, 2] = (1, 0). \end{aligned}$$

Obviamente, el vértice común a ambos tramos, $c_3 = \hat{c}_0$, lo podemos calcular con ambas formas polares. El resto, no.

La extensión a curvas racionales a trozos es bastante sencilla, pues, como vimos en el tema anterior, las parametrizaciones racionales en \mathbb{A}^n se obtienen a partir de parametrizaciones polinómicas en \mathbb{R}^{n+1} , donde la componente adicional corresponde al denominador de la parametrización. Así pues, introduciendo una componente más, el algoritmo de De Boor es aplicable a curvas racionales, sin más que dividir por la coordenada adicional.

8. Propiedades de las curvas polinómicas a trozos

[Vídeo de Propiedades de Curvas spline](#)

El algoritmo de De Boor expresa que las parametrizaciones de las curvas polinómicas a trozos son combinaciones baricéntricas de los vértices del polígono B-spline, así que disfrutan de las mismas propiedades que las curvas de Bézier, tales como la invariancia afín, disminución de la variación, envolvente convexa... En el caso de las curvas racionales, se mantiene la invariancia proyectiva. Otras, como pasar por los vértices inicial y final, dependen de la elección de los nudos auxiliares, aunque ya hemos dicho que no

es común tomarlos distintos de los nudos inicial y final de la parametrización. [Ejemplo.](#)

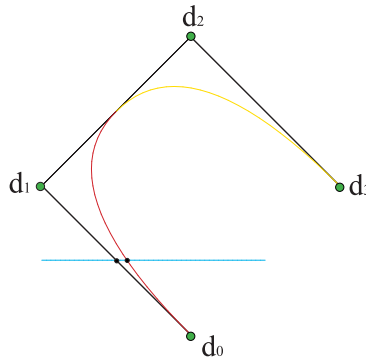


Figura 13: Disminución de la variación: una recta corta al tramo de una curva en a lo sumo tantos puntos como a su tramo de polígono

Como en el algoritmo de De Boor sólo intervienen cocientes de diferencias de valores del parámetro u , la parametrización es insensible a cambios afines de parámetro $\tilde{u} = au + b$, siempre que transformemos los nudos de idéntica manera, $\tilde{u}_i = au_i + b$, $i = 0, \dots, 2n + N - 2$.

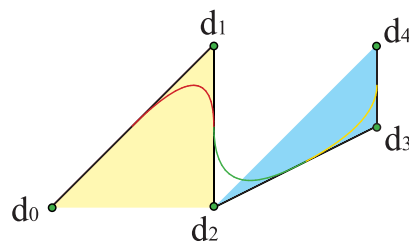


Figura 14: Cada tramo está contenido en la envolvente convexa de su polígono

Más aún, como en cada intervalo las curvas spline son curvas polinómicas, gozan de las propiedades que ya estudiamos para estas, no sólo globalmente, sino en cada tramo concreto. Así cada tramo de curva, por ejemplo el i -ésimo, está contenido en la envolvente convexa de su polígono de control, $\{d_{i-1}, \dots, d_{i+n-1}\}$. Lo mismo sucede con la propiedad de la disminución de

la variación. Una recta corta al tramo i -ésimo de la curva en a lo sumo tantos puntos como al polígono de control de dicho tramo. [Ejemplo](#).

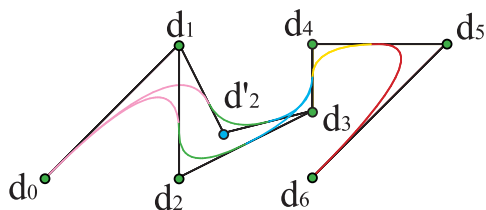


Figura 15: Control local: un vértice de un spline parabólico controla tres tramos de curva

Finalmente, hay propiedades que presentan una mejora sustancial. Cada tramo de la curva depende exclusivamente de los $n + 1$ vértices de su polígono de control. Por tanto, no se ve afectado por los cambios de los demás vértices. Visto de otro modo, un vértice genérico d_i pertenece a los polígonos de control de todos los tramos desde el $(i - n + 1)$ -ésimo, $\{d_{i-n}, \dots, d_i\}$, hasta el $(i + 1)$ -ésimo, $\{d_i, \dots, d_{i+n}\}$. Así pues, un desplazamiento del vértice d_i modifica tan sólo a un total de $n + 1$ tramos de la curva. Obviamente, si el vértice está próximo al comienzo o al final del polígono, controlará un número inferior de tramos. Por ejemplo, el vértice inicial, d_0 , sólo afecta al primer tramo. A esta propiedad se la denomina **control local** y no existía para las curvas de Bézier, donde la modificación de un vértice afectaba a toda la curva. [Ejemplo](#).

9. Algoritmo de inserción

[Vídeo de Algoritmo de inserción y elevación del grado](#)

Íntimamente ligado al algoritmo de De Boor está el algoritmo de inserción de nudos. Esencialmente es el equivalente del algoritmo de subdivisión estudiado para el algoritmo de De Casteljaou.

Consideremos un polígono de control B-spline para una curva de grado n de N tramos, $\{d_0, \dots, d_{n+N-1}\}$ y una sucesión de nudos $\{u_0, \dots, u_{2n+N-2}\}$, en la cual queremos introducir un nudo adicional \tilde{u} , distinto o igual a alguno de los originales, de modo que la gráfica de la curva permanezca invariable.

Como el número de vértices, $L + 1$ y el número de nudos, $K + 1$, están ligados entre sí, $K = L + n - 1$, la inclusión del nuevo nudo supone la aparición de un tramo nuevo de curva, es decir, una subdivisión, y, por tanto, un nuevo vértice.

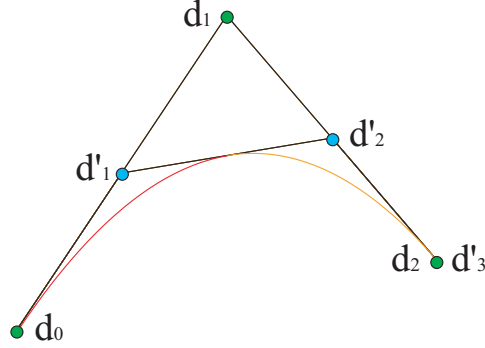


Figura 16: Inserción de un nudo en una parábola de un tramo

Si en la nueva sucesión ordenada de nudos, $\{\tilde{u}_0, \dots, \tilde{u}_{2n+N-1}\}$, el nuevo nudo ocupa la posición i -ésima,

$$\begin{aligned}\tilde{u}_j &= u_j, \quad j = 0, \dots, i-1, \quad \tilde{u}_i = \tilde{u}, \\ \tilde{u}_j &= u_{j-1}, \quad j = i+1, \dots, 2n+N-1,\end{aligned}$$

como ya sabemos que la forma polar nos proporciona los vértices del polígono de control, tenemos ya todos los ingredientes para construirlo. [Ejemplo](#).

$$\tilde{d}_j = d[\tilde{u}_j, \dots, \tilde{u}_{j+n-1}], \quad j = 0, \dots, n+N \quad (23)$$

Por tanto, los únicos vértices que se modifican con el algoritmo son los n vértices en los que interviene el nuevo nudo, $\tilde{d}_{i-n+1}, \dots, \tilde{d}_i$. Los vértices iniciales, $\tilde{d}_0, \dots, \tilde{d}_{i-n}$, son iguales a los originales y los finales, $\tilde{d}_{i+1}, \dots, \tilde{d}_{n+N}$ sólo sufren el incremento de su ordinal, d_i, \dots, d_{n+N-1} .

Los nuevos tramos en los que se excinde el original corresponden a los intervalos $[\tilde{u}_{i-1}, \tilde{u}_i]$, $[\tilde{u}_i, \tilde{u}_{i+1}]$, que tienen cada uno, respectivamente, por polígonos $\{\tilde{d}_{i-n}, \dots, \tilde{d}_i\}$, $\{\tilde{d}_{i-n+1}, \dots, \tilde{d}_{i+1}\}$, que engloban la totalidad de los vértices nuevos.

La aplicación reiterada del algoritmo de inserción de los nudos u_i, u_{i+1} hasta que alcancen ambos la multiplicidad n es una manera equivalente de

obtener el polígono de control de la curva en el intervalo $[u_i, u_{i+1}]$. **Ejemplo.** Esto es obvio, ya que la sucesión de nudos de dicho intervalo, después de la inserción será $\{u_i^{<n>}, u_{i+1}^{<n>}\}$ que corresponde, como sabemos, a una curva de Bézier, con vértices $c_i = d[u_i^{<n-i>}, u_{i+1}^{<i>}]$.

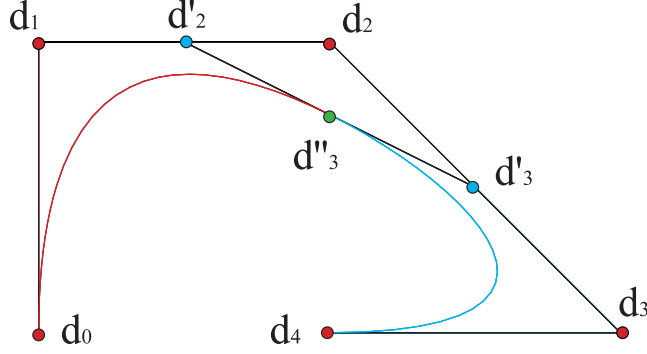


Figura 17: Spline cúbico de dos tramos: la inserción del nudo central dos veces proporciona los polígonos de control de ambos tramos

Lo vemos con un ejemplo. Consideremos un spline cuadrático de dos tramos determinado por el polígono $\{(0, 0), (0, 1), (1, 1), (2, 1)\}$ y la sucesión de nudos $\{0, 0, 1, 2, 2\}$. La inserción del nudo central proporciona unos nuevos nudos, $\{0, 0, 1, 1, 2, 2\}$, y un polígono, $\{(0, 0), (0, 1), (1/2, 1), (1, 1), (2, 1)\}$,

$$\begin{aligned}
 \tilde{d}_0 &= d[\tilde{u}_0, \tilde{u}_1] = d[0, 0] = d[u_0, u_1] = d_0 = (0, 0), \\
 \tilde{d}_1 &= d[\tilde{u}_1, \tilde{u}_2] = d[0, 1] = d[u_1, u_2] = d_1 = (0, 1), \\
 \tilde{d}_2 &= d[\tilde{u}_2, \tilde{u}_3] = d[1, 1] = d[u_2, u_2] = \frac{u_2 - 1}{u_2 - u_1} \left(\frac{u_2 - 1}{u_2 - u_0} d_0 + \frac{1 - u_0}{u_2 - u_0} d_1 \right) \\
 &\quad + \frac{1 - u_1}{u_2 - u_1} \left(\frac{u_3 - 1}{u_3 - u_1} d_1 + \frac{1 - u_1}{u_3 - u_1} d_2 \right) = \frac{d_1 + d_2}{2} = (1/2, 1), \\
 \tilde{d}_3 &= d[\tilde{u}_3, \tilde{u}_4] = d[1, 2] = d[u_2, u_3] = d_2 = (1, 1), \\
 \tilde{d}_4 &= d[\tilde{u}_4, \tilde{u}_5] = d[2, 2] = d[u_3, u_4] = d_3 = (2, 1).
 \end{aligned}$$

Obsérvese que, al estar repetidos todos los nudos dos veces para una curva de grado dos, hemos obtenido los polígonos de control de ambos tramos de la curva, $\{(0, 0), (0, 1), (1/2, 1)\}$ para el primer tramo, el definido en $[0, 1]$, y $\{(1/2, 1), (1, 1), (2, 1)\}$ para el segundo tramo, el definido inicialmente en $[1, 2]$.

Probamos ahora a insertar un nudo que no esté en la sucesión inicial. Por ejemplo, $u = 3/2$, lo que proporciona la nueva sucesión $\{0, 0, 1, 3/2, 2, 2\}$. Como $3/2 \in [1, 2]$, emplearemos para los cálculos, cuando sea necesario, la forma polar del segundo segmento, la definida por los vértices $\{d_1, d_2, d_3\}$ y los nudos $\{u_1, u_2, u_3, u_4\}$, es decir, desplazando los índices una unidad en (23),

$$\begin{aligned}
\tilde{d}_0 &= d[\tilde{u}_0, \tilde{u}_1] = d[0, 0] = d[u_0, u_1] = d_0 = (0, 0), \\
\tilde{d}_1 &= d[\tilde{u}_1, \tilde{u}_2] = d[0, 1] = d[u_1, u_2] = d_1 = (0, 1), \\
\tilde{d}_2 &= d[\tilde{u}_2, \tilde{u}_3] = d[1, 3/2] = \frac{u_3 - 1}{u_3 - u_2} \left(\frac{u_3 - 3/2}{u_3 - u_1} d_1 + \frac{3/2 - u_1}{u_3 - u_1} d_2 \right) \\
&\quad + \frac{1 - u_2}{u_3 - u_2} \left(\frac{u_4 - 3/2}{u_4 - u_2} d_2 + \frac{3/2 - u_2}{u_4 - u_2} d_3 \right) = \frac{d_1 + 3d_2}{4} = (3/4, 1), \\
\tilde{d}_3 &= d[\tilde{u}_3, \tilde{u}_4] = d[3/2, 2] = \frac{u_3 - 3/2}{u_3 - u_2} \left(\frac{u_3 - 2}{u_3 - u_1} d_1 + \frac{2 - u_1}{u_3 - u_1} d_2 \right) \\
&\quad + \frac{3/2 - u_2}{u_3 - u_2} \left(\frac{u_4 - 2}{u_4 - u_2} d_2 + \frac{2 - u_2}{u_4 - u_2} d_3 \right) = \frac{d_2 + d_3}{2} = (3/2, 1), \\
\tilde{d}_4 &= d[\tilde{u}_4, \tilde{u}_5] = d[2, 2] = d[u_3, u_4] = d_3 = (2, 1),
\end{aligned}$$

es decir, el nuevo polígono es $\{(0, 0), (0, 1), (3/4, 1), (3/2, 1), (2, 1)\}$. La curva queda dividida en tres tramos, definidos en los intervalos $[0, 1]$, $[1, 3/2]$, $[3/2, 2]$.

A su vez, si volvemos a insertar el nudo $u = 3/2$, la sucesión de nudos será $\{0, 0, 1, 3/2, 3/2, 2, 2\}$ y el polígono B-spline, usando la forma polar del segundo tramo, el definido en $[1, 3/2]$, cuando sea necesario,

$$\begin{aligned}
\hat{d}_0 &= d[\hat{u}_0, \hat{u}_1] = d[0, 0] = d[\tilde{u}_0, \tilde{u}_1] = \tilde{d}_0 = (0, 0), \\
\hat{d}_1 &= d[\hat{u}_1, \hat{u}_2] = d[0, 1] = d[\tilde{u}_1, \tilde{u}_2] = \tilde{d}_1 = (0, 1), \\
\hat{d}_2 &= d[\hat{u}_2, \hat{u}_3] = d[1, 3/2] = d[\tilde{u}_2, \tilde{u}_3] = \tilde{d}_2 = (3/4, 1), \\
\hat{d}_3 &= d[\hat{u}_3, \hat{u}_4] = d[3/2, 3/2] = \frac{\tilde{u}_3 - 3/2}{\tilde{u}_3 - \tilde{u}_2} \left(\frac{\tilde{u}_3 - 3/2}{\tilde{u}_3 - \tilde{u}_1} \tilde{d}_1 + \frac{3/2 - \tilde{u}_1}{\tilde{u}_3 - \tilde{u}_1} \tilde{d}_2 \right) \\
&\quad + \frac{3/2 - \tilde{u}_2}{\tilde{u}_3 - \tilde{u}_2} \left(\frac{\tilde{u}_4 - 3/2}{\tilde{u}_4 - \tilde{u}_2} \tilde{d}_2 + \frac{3/2 - \tilde{u}_2}{\tilde{u}_4 - \tilde{u}_2} \tilde{d}_3 \right) = \frac{\tilde{d}_2 + \tilde{d}_3}{2} = (9/8, 1), \\
\hat{d}_4 &= d[\hat{u}_4, \hat{u}_5] = d[3/2, 2] = d[\tilde{u}_3, \tilde{u}_4] = \tilde{d}_3 = (3/2, 1), \\
\hat{d}_5 &= d[\hat{u}_5, \hat{u}_6] = d[2, 2] = d[\tilde{u}_4, \tilde{u}_5] = \tilde{d}_4 = (2, 1),
\end{aligned}$$

es $\{0, 0), (0, 1), (3/4, 1), (9/8, 1), (3/2, 1), (2, 1)\}$, de donde podemos leer, por

ejemplo, el polígono de control del último tramo, definido en $[3/2, 2]$, que está formado por los tres últimos vértices, $\{(9/8, 1), (3/2, 1), (2, 1)\}$.

10. Diferenciabilidad

Vídeo de Derivadas de curvas spline

En el primer tema demostramos que las derivadas de una curva de Bézier se podían expresar por medio de la forma polar,

$$\frac{d^r c(t)}{dt^r} = \frac{n!}{(n-r)!} c[t^{\langle n-r \rangle}, \mathbf{e}^{\langle r \rangle}]. \quad (24)$$

usando interpolaciones vectoriales con el vector \mathbf{e} , que une sobre la recta los puntos 0 y 1.

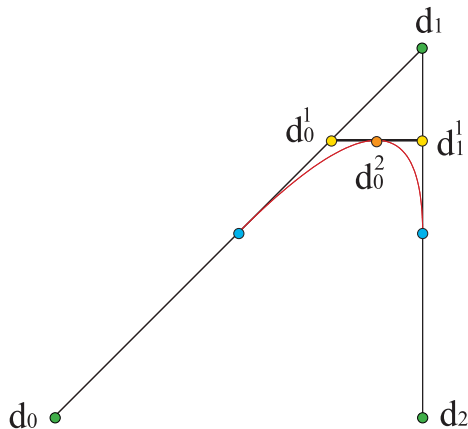


Figura 18: Los dos vértices del penúltimo paso del algoritmo de De Boor nos dan la tangente en un punto de la curva

Esta expresión sigue siendo válida para cada tramo de una curva spline, ya que está basada en la forma polar. Por ejemplo, para la derivada primera en $u \in [u_i, u_{i+1}]$,

$$\frac{dc(u)}{du} = \frac{n}{u_{i+1} - u_i} d[u^{\langle n-1 \rangle}, \mathbf{e}] = \frac{n}{u_{i+1} - u_i} (d_1^{n-1}(u) - d_0^{n-1}(u)), \quad (25)$$

es decir, los vértices del último paso del algoritmo determinan la tangente a la curva en $c(u)$. [Ejemplo](#). Para derivadas superiores,

$$\frac{d^r c(u)}{d^r u} = \frac{n!}{(n-r)!} \frac{1}{(u_{i+1} - u_i)^r} d[u^{\langle n-r \rangle}, \mathbf{e}^{\langle r \rangle}]. \quad (26)$$

En este apartado vamos a volver sobre el tema ya mencionado varias veces de la diferenciabilidad de curvas polinómicas de grado n a trozos. Consideraremos el caso más sencillo, en el que todos los nudos interiores, desde u_n hasta u_{n+N-2} son simples, es decir, no están repetidos.

Nos vamos a fijar en la unión de dos tramos de la curva, parametrizados en los intervalos $[u_{i-1}, u_i]$, $[u_i, u_{i+1}]$. El primero de los tramos tiene sucesión de nudos $\{u_{i-n}, \dots, u_{i+n-1}\}$ y polígono $\{d_{i-n}, \dots, d_i\}$, mientras que el segundo tiene nudos $\{u_{i-n+1}, \dots, u_{i+n}\}$ y vértices $\{d_{i-n+1}, \dots, d_{i+1}\}$.

Consideremos sobre el intervalo $[u_{i-1}, u_i]$ la curva de grado $n-1$ que tiene por forma polar \hat{d} , de modo que,

$$\hat{d}[v_1, \dots, v_{n-1}] := d[u_i, v_1, \dots, v_{n-1}],$$

es decir, \hat{d} es el resultado de fijar el nudo u_i en la forma polar d .

Esta nueva forma polar corresponde a una curva de vértices,

$$\begin{aligned} d_{i-n+1} &= d[u_{i-n+1}, \dots, u_i] = \hat{d}[u_{i-n+1}, \dots, u_{i-1}], \dots, \\ d_i &= d[u_i, \dots, u_{i+n-1}] = \hat{d}[u_{i+1}, \dots, u_{i+n-1}], \end{aligned}$$

con lo cual su sucesión de nudos es $\{u_{i-n+1}, \dots, u_{i-1}, u_{i+1}, \dots, u_{i+n-1}\}$. Es decir, hemos perdido los nudos u_{i-n} , u_i y el intervalo en el que está definida es $[u_{i-1}, u_{i+1}]$.

Al mismo resultado habríamos llegado si hubiéramos procedido con la forma polar de la curva en el intervalo $[u_i, u_{i+1}]$.

Tenemos, pues, una curva, $\hat{c}(u)$ que posee $n-1$ derivadas no nulas en u_i , que además coinciden con las derivadas de nuestra curva, $c(u)$, en u_i hasta orden $n-1$, salvo un factor dependiente del grado, puesto que, por la simetría de la forma polar, como $c(u) = d[u^{\langle n \rangle}]$,

$$\left. \frac{dc(u)}{du} \right|_{u=u_i} = \sum_{j=1}^n \left. \frac{dd[u_i^{\langle j-1 \rangle}, v_j, u_i^{\langle n-j \rangle}]}{dv_j} \right|_{v_j=u_i} = n \left. \frac{dd[u_i^{\langle n-1 \rangle}, v_n]}{dv_n} \right|_{v_n=u_i},$$

ya la derivada de $\hat{c}(u)$ es idéntica, pero con un factor $n-1$ en vez de n , ya que tiene una variable menos. Un razonamiento análogo se sigue para las

derivadas superiores, hasta la $n - 1$, ya que la derivada n -ésima de $\hat{c}(u)$ es nula.

Por tanto, como, por ser polinómica, $\hat{c}(u)$ es de clase C^∞ en $[u_{i-1}, u_{i+1}]$ y las $n - 1$ derivadas comunes de $c(u)$ en u_i son continuas. Así pues, hemos demostrado el resultado que anunciábamos, que $c(u)$ es de clase C^{n-1} en u_i , si este nudo es simple. \square

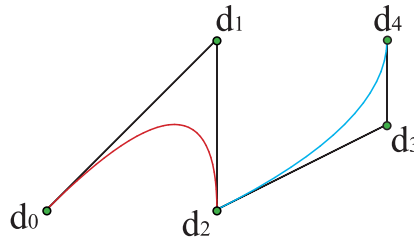


Figura 19: Spline cuadrático de tres tramos: al tomar el nudo tercero igual al cuarto, desaparece el segundo tramo y la tangente no es continua

El caso de nudos múltiples no es mucho más complicado. Consideremos ahora la unión de dos tramos en u_i , de modo que u_i es doble. Es decir, los intervalos de los tramos son $[u_{i-1}, u_i]$, $[u_{i+1}, u_{i+2}]$. Entre medias hay un intervalo vacío $[u_i = u_{i+1}]$. El razonamiento es enteramente similar, sólo que debemos fijar dos nudos, u_i , u_{i+1} para definir \hat{d} y la curva auxiliar, $\hat{c}(u)$, es de grado $n - 2$. Con lo cual sólo podemos concluir que $c(u)$ sea de clase C^{n-2} en u_i .

Por tanto, en una unión de tramos en la que el nudo común de los intervalos tiene multiplicidad r , la curva es de clase al menos C^{n-r} , pudiendo ser de clase superior, aunque genéricamente no lo sea. \square [Ejemplo](#).

Este resultado es una muestra más de la versatilidad de las curvas polinómicas a trozos, que nos permiten combinar tramos con diferentes clases de diferenciabilidad sin más que repetir los nudos de las uniones.

11. Elevación del grado

[Vídeo de Algoritmo de inserción y elevación del grado](#)

En el tema segundo demostramos la expresión de la elevación del grado,

$$c^1[t_1, \dots, t_{n+1}] = \frac{1}{n+1} \sum_{i=1}^{n+1} c[t_1, \dots, t_{i-1}, t_{i+1}, \dots, t_{n+1}], \quad (27)$$

que relaciona la forma polar de una curva de grado n con la forma polar de grado $n+1$ para la misma curva. Esta misma fórmula es válida asimismo para elevar el grado de un tramo de una curva polinómica de grado n a trozos, con algunas precisiones.

La fórmula (27) está pensada para curvas de Bézier y, por tanto, no hace referencia a ninguna sucesión de nudos. Pero es claro que, como la sucesión de nudos para estas curvas es de la forma $\{u_0^{<n>}, u_1^{<n>}\}$ y la curva elevada sigue siendo de Bézier, pero de grado $n+1$, su sucesión de nudos será $\{u_0^{<n+1>}, u_1^{<n+1>}\}$. Por tanto, para elevar el grado, hay que usar la fórmula (27), pero teniendo en cuenta que hay que incrementar en una unidad la multiplicidad de todos los nudos interiores, los que definen los tramos de la curva, desde u_{n-1} hasta u_{n+N-1} . **Ejemplo. Ejemplo.**

Que hay que añadir dos nudos a cada tramo es lógico, ya que, si queremos aumentar el grado en una unidad, el número de nudos de cada tramo deberá pasar de $2n$ a $2n+2$. Además, deben ser los de los extremos del intervalo, porque, como hemos anunciado, las curvas de grado n son genéricamente de clase C^{n-1} y no de clase C^n , como corresponde a una curva de grado $n+1$. Esto se solventa introduciendo tramos ficticios, es decir, intervalos vacíos de la forma $[u_i, u_i]$, lo que supone duplicar los nudos de los extremos.

Por tanto, para elevar el grado de una curva polinómica de grado a n a trozos en la que los nudos interiores desde u_n hasta u_{n+N-2} son simples (no están repetidos), basta emplear la fórmula (27), por ejemplo para calcular el polígono B-spline, con la nueva sucesión de nudos,

$$\{u_0, \dots, u_{n-1}, u_{n-1}, \dots, u_{n+N-1}, u_{n+N-1}, u_{n+N}, \dots, u_{2n+2N-2}\}, \quad (28)$$

con lo cual acabamos con $2n+2N$ nudos y $n+2N$ vértices en este caso.

Por ejemplo, si tenemos un spline cuadrático de tres tramos determinado por el polígono $\{(0, 0), (0, 1), (1, 1), (2, 0), (1, -1)\}$ y los nudos $\{0, 0, 1, 2, 3, 3\}$, podemos ver la misma curva como un spline cúbico de sucesión de nudos $\{0, 0, 0, 1, 1, 2, 2, 3, 3, 3\}$. Los vértices los obtenemos por la fórmula habitual, combinada con la de elevación de grado, pero primero calcularemos algunos

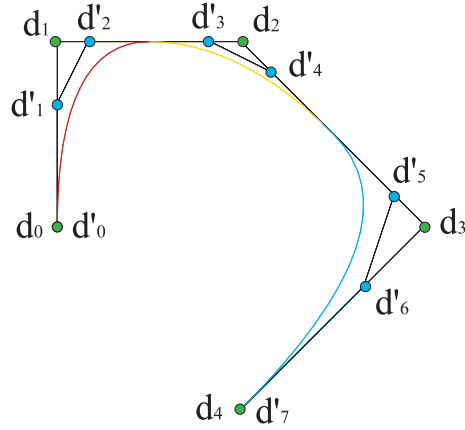


Figura 20: Elevación del grado de un spline cuadrático de tres tramos

puntos necesarios,

$$\begin{aligned}
 d[0, 0] &= d[u_0, u_1] = d_0 = (0, 0), \\
 d[1, 1] &= d[u_2, u_2] = \frac{u_3 - 1}{u_3 - u_2} \left(\frac{u_3 - 1}{u_3 - u_1} d_1 + \frac{1 - u_1}{u_3 - u_1} d_2 \right) \\
 &\quad + \frac{1 - u_2}{u_3 - u_2} \left(\frac{u_4 - 1}{u_4 - u_2} d_2 + \frac{1 - u_2}{u_4 - u_2} d_3 \right) = \frac{1}{2} d_1 + \frac{1}{2} d_2 = (1/2, 1), \\
 d[2, 2] &= d[u_3, u_3] = \frac{u_3 - 2}{u_3 - u_2} \left(\frac{u_3 - 2}{u_3 - u_1} d_1 + \frac{2 - u_1}{u_3 - u_1} d_2 \right) \\
 &\quad + \frac{2 - u_2}{u_3 - u_2} \left(\frac{u_4 - 2}{u_4 - u_2} d_2 + \frac{2 - u_2}{u_4 - u_2} d_3 \right) = \frac{d_2 + d_3}{2} = (3/2, 1/2), \\
 d[3, 3] &= d[u_4, u_5] = d_4 = (1, -1),
 \end{aligned}$$

donde para calcular $c(1) = d[1, 1]$, $c(2) = d[2, 2]$, hemos usado la forma polar de la curva en el segundo tramo, el definido en $[1, 2]$, es decir, desplazando

los índices una unidad respecto a (23).

$$\begin{aligned}
d_0^1 &= d^1[\tilde{u}_0, \tilde{u}_1, \tilde{u}_2] = d^1[0, 0, 0] = d[0, 0] = d[u_0, u_1] = d_0 = (0, 0), \\
d_1^1 &= d^1[\tilde{u}_1, \tilde{u}_2, \tilde{u}_3] = d^1[0, 0, 1] = \frac{d[0, 0] + 2d[0, 1]}{3} = \frac{d[0, 0] + 2d_1}{3} \\
&= (0, 2/3), \\
d_2^1 &= d^1[\tilde{u}_2, \tilde{u}_3, \tilde{u}_4] = d^1[0, 1, 1] = \frac{2d[0, 1] + d[1, 1]}{3} = \frac{2d_1 + d[1, 1]}{3} \\
&= (1/6, 1), \\
d_3^1 &= d^1[\tilde{u}_3, \tilde{u}_4, \tilde{u}_5] = d^1[1, 1, 2] = \frac{d[1, 1] + 2d[1, 2]}{3} = \frac{d[1, 1] + 2d_2}{3} \\
&= (5/6, 1), \\
d_4^1 &= d^1[\tilde{u}_4, \tilde{u}_5, \tilde{u}_6] = d^1[1, 2, 2] = \frac{2d[1, 2] + d[2, 2]}{3} = \frac{2d_2 + d[2, 2]}{3} \\
&= (7/6, 5/6), \\
d_5^1 &= d^1[\tilde{u}_5, \tilde{u}_6, \tilde{u}_7] = d^1[2, 2, 3] = \frac{d[2, 2] + 2d[2, 3]}{3} = \frac{d[2, 2] + 2d_3}{3} \\
&= (11/6, 1/6), \\
d_6^1 &= d^1[\tilde{u}_6, \tilde{u}_7, \tilde{u}_8] = d^1[2, 3, 3] = \frac{2d[2, 3] + d[3, 3]}{3} = \frac{2d_3 + d[3, 3]}{3} \\
&= (5/3, -1/3), \\
d_7^1 &= d^1[\tilde{u}_7, \tilde{u}_8, \tilde{u}_9] = d^1[3, 3, 3] = d[3, 3] = d[u_4, u_5] = d_4 = (1, -1),
\end{aligned}$$

$\{(0, 0), (0, 2/3), (1/6, 1), (5/6, 1), (7/6, 5/6), (11/6, 1/6), (5/3, -1/3), (1, -1)\}$ es el polígono B-spline de grado tres.

12. Funciones B-spline o nodales

Vídeo de Funciones nodales

Del mismo modo que la aplicación del algoritmo de De Casteljaou conduce a la aparición de los polinomios de Bernstein, el algoritmo de De Boor introduce funciones polinómicas en cada tramo.

Sea una curva de grado n a trozos con polígono B-spline $\{d_0, \dots, d_{n+N-1}\}$ y nudos $\{u_0, \dots, u_{2n+N-2}\}$. Podemos almacenar todos los pasos del algoritmo de De Boor en unas funciones $\{N_0^n(u), \dots, N_{n+N-1}^n(u)\}$ polinómicas a trozos

denominadas **funciones nodales o B-spline**,

$$c(u) = \sum_{i=0}^{n+N-1} d_i N_i^n(u). \quad (29)$$

Centrémosnos en el primer tramo, $[u_{n-1}, u_n]$, de polígono $\{d_0, \dots, d_n\}$ y nudos $\{u_0, \dots, u_{2n-1}\}$,

$$c(u) = \sum_{i=0}^n d_i N_i^n(u) = \sum_{i=0}^{n-1} d_i^1(u) N_{i+1}^{n-1}(u) = \dots = \sum_{i=0}^1 d_i^{n-1}(u) N_{i+n-1}^1(u).$$

El ordinal de la función que acompaña a $d_i^1(u)$ se adelanta una unidad a $i + 1$, ya que para los splines de grado $n - 1$ el primer intervalo no es $[u_{n-1}, u_n]$, sino $[u_{n-2}, u_{n-1}]$. Así pues, en el intervalo considerado intervienen las funciones $N_1^{n-1}(u), \dots, N_n^{n-1}(u)$.

Por idéntico motivo, a d_i^{n-1} le acompaña $N_{i+n-1}^1(u)$. En cada iteración avanza una unidad el ordinal de la función.

Esta expresión, combinada con el primer paso del algoritmo de De Boor,

$$d_i^1(u) = \frac{u_{i+n} - u}{u_{i+n} - u_i} d_i + \frac{u - u_i}{u_{i+n} - u_i} d_{i+1}, \quad (30)$$

cuyos vértices también deben proporcionar el punto $c(u)$ tras $n-1$ iteraciones,

$$\begin{aligned} c(u) &= \sum_{i=0}^{n-1} \left(\frac{u_{i+n} - u}{u_{i+n} - u_i} d_i + \frac{u - u_i}{u_{i+n} - u_i} d_{i+1} \right) N_{i+1}^{n-1}(u) \\ &= \sum_{i=0}^{n-1} d_i \frac{u_{i+n} - u}{u_{i+n} - u_i} N_{i+1}^{n-1}(u) + \sum_{i=1}^n d_i \frac{u - u_{i-1}}{u_{i+n-1} - u_{i-1}} N_i^{n-1}(u), \end{aligned}$$

nos facilita la regla de recursión de las funciones nodales, tras comparar con (29),

$$N_i^n(u) = \frac{u_{i+n} - u}{u_{i+n} - u_i} N_{i+1}^{n-1}(u) + \frac{u - u_{i-1}}{u_{i+n-1} - u_{i-1}} N_i^{n-1}(u), \quad u \in [u_{n-1}, u_n], \quad (31)$$

teniendo en cuenta que se deben anular los sumandos con nudos u_{-1} o u_{2n} , para lo cual las funciones $N_0^{n-1}(u)$, $N_{n+1}^{n-1}(u)$ se tienen que anular en el tramo $[u_{n-1}, u_n]$.

Esta notación es peligrosa, ya que n hace referencia tanto al intervalo como al grado. Se aclara obteniendo la iteración r -ésima,

$$c(u) = \sum_{i=0}^{n-r-1} d_i^{r+1}(u) N_{i+r+1}^{n-r-1}(u) = \sum_{i=0}^{n-r} d_i^r(u) N_{i+r}^{n-r}(u),$$

a partir del algoritmo de De Boor (21),

$$d_i^{r+1}(u) = \frac{u_{i+n} - u}{u_{i+n} - u_{i+r}} d_i^r(u) + \frac{u - u_{i+r}}{u_{i+n} - u_{i+r}} d_{i+1}^r(u),$$

$$N_{i+r}^{n-r}(u) = \frac{u_{i+n} - u}{u_{i+n} - u_{i+r}} N_{i+r+1}^{n-r-1}(u) + \frac{u - u_{i+r-1}}{u_{i+n-1} - u_{i+r-1}} N_{i+r}^{n-r-1}(u), \quad (32)$$

de modo que (31) es el caso $r = 0$ de esta expresión.

Para completar la cadena de recursión, sólo hace falta el paso inicial, los polinomios $N_{n-1}^1(u)$, $N_n^1(u)$, que se despejan del algoritmo de De Boor de grado uno,

$$c(u) = \frac{u_n - u}{u_n - u_{n-1}} d_0^{n-1}(u) + \frac{u - u_{n-1}}{u_n - u_{n-1}} d_1^{n-1}(u) = \sum_{i=0}^1 d_i^{n-1}(u) N_{i+n-1}^1(u),$$

$$N_{n-1}^1(u) = \frac{u_n - u}{u_n - u_{n-1}}, \quad N_n^1(u) = \frac{u - u_{n-1}}{u_n - u_{n-1}}.$$

Incluso podemos ir un paso más allá, proponiendo un polinomio de grado cero, constante, que satisfaga la recursión,

$$c(u) = d_0^n(u) N_n^0(u),$$

para $r = n - 1$ en (32),

$$N_{i+n-1}^1(u) = \frac{u_{i+n} - u}{u_{i+n} - u_{i+n-1}} N_{i+n}^0(u) + \frac{u - u_{i+n-2}}{u_{i+n-1} - u_{i+n-2}} N_{i+n-1}^0(u),$$

$$N_n^0(u) = 1, \quad u \in [u_{n-1}, u_n], \quad (33)$$

anulándose $N_{n-1}^0(u)$, $N_{n+1}^0(u)$ en el tramo $[u_{n-1}, u_n]$.

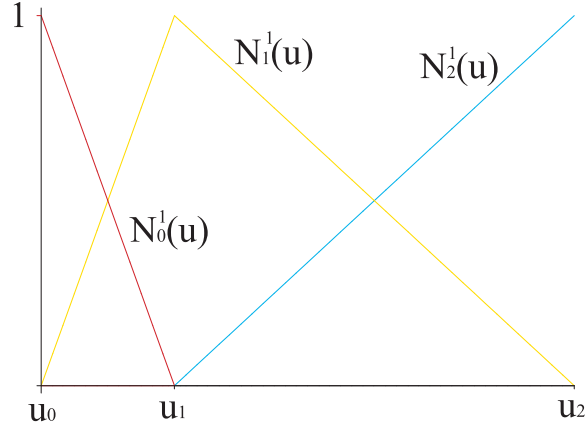


Figura 21: Funciones nodales de grado uno

Para el resto de tramos la expresión es la misma, ya que el índice de las funciones nodales avanza una unidad con cada tramo, igual que el índice de los vértices que intervienen en el mismo. \square

Recapitulando, las funciones B-spline gozan de una propiedad de recursividad, que permite construirlas de forma iterativa a partir de las de menor grado,

$$\begin{aligned}
 N_i^n(u) &= \frac{u - u_{i-1}}{u_{i+n-1} - u_{i-1}} N_i^{n-1}(u) + \frac{u_{i+n} - u}{u_{i+n} - u_i} N_{i+1}^{n-1}(u), \\
 N_i^0(u) &= \begin{cases} 1 & u \in [u_{i-1}, u_i] \\ 0 & u \notin [u_{i-1}, u_i] \end{cases}, \quad (34)
 \end{aligned}$$

para una sucesión de índices $\{u_0, \dots, u_{2n+N-2}\}$. **Ejemplo.**

La función nodal $N_i^n(u)$ tiene, pues, soporte en $n+1$ intervalos, $[u_{i-1}, u_{i+n}]$, ya que el vértice d_i controla todos los tramos desde el $i - n + 1$ hasta el $i + 1$, es decir, desde el intervalo $[u_{i-1}, u_i]$ hasta el intervalo $[u_{i+n-1}, u_{i+n}]$. Fuera de esos intervalos la función nodal correspondiente debe anularse.

Por ello, la fórmula de recursión requiere nudos fuera de la sucesión, u_{-1} , para el cálculo de $N_0^n(u)$, y u_{2n+N-1} , para $N_{n+N-1}^n(u)$, como corresponde al hecho de que los intervalos de definición de estas funciones son, respectivamente, $[u_{-1}, u_n]$ y $[u_{n+N-2}, u_{2n+N-1}]$.

Estos nudos no afectan al intervalo de definición, $[u_{n-1}, u_{n+N-1}]$, de la curva, ya que aparecen en factores que multiplican a $N_0^{n-1}(u)$, cuyo intervalo

es $[u_{-1}, u_{n-1}]$ y a $N_{n+N}^{n-1}(u)$, cuyo intervalo es $[u_{n+N-1}, u_{2n+N-1}]$. Por tanto, podemos tomarlos arbitrarios, tal como se hace en el formato IGES, aunque suponga un exceso de información. Otra opción sería eliminar estos términos o tomar $u_{-1} = -\infty$, $u_{2n+N-1} = \infty$, con lo cual la partición cubriría toda la recta real, y los coeficientes indeterminados tomarían el valor de la unidad.

Por ejemplo, consideremos las funciones nodales que se pueden definir sobre la sucesión de nudos $\{0, 1, 3, 4, 5, 7\}$. Las funciones de grado cero, constantes, son

$$\begin{aligned}
N_0^0(u) &= \begin{cases} 1 & u < 0 \\ 0 & u \geq 0 \end{cases}, & N_1^0(u) &= \begin{cases} 1 & u \in [0, 1) \\ 0 & u \notin [0, 1) \end{cases}, \\
N_2^0(u) &= \begin{cases} 1 & u \in [1, 3) \\ 0 & u \notin [1, 3) \end{cases}, & N_3^0(u) &= \begin{cases} 1 & u \in [3, 4) \\ 0 & u \notin [3, 4) \end{cases}, \\
N_4^0(u) &= \begin{cases} 1 & u \in [4, 5) \\ 0 & u \notin [4, 5) \end{cases}, & N_5^0(u) &= \begin{cases} 1 & u \in [5, 7) \\ 0 & u \notin [5, 7) \end{cases}, \\
N_6^0(u) &= \begin{cases} 1 & u \geq 7 \\ 0 & u < 7 \end{cases},
\end{aligned}$$

Las funciones de grado uno sobre la misma sucesión,

$$\begin{aligned}
N_0^1(u) &= \begin{cases} 1 & u < 0 \\ 1 - u & u \in [0, 1) \\ 0 & u \geq 1 \end{cases}, & N_1^1(u) &= \begin{cases} 0 & u < 0 \\ u & u \in [0, 1) \\ (3 - u)/2 & u \in [1, 3) \\ 0 & u \geq 3 \end{cases}, \\
N_2^1(u) &= \begin{cases} 0 & u < 1 \\ (u - 1)/2 & u \in [1, 3) \\ 4 - u & u \in [3, 4) \\ 0 & u \geq 4 \end{cases}, & N_3^1(u) &= \begin{cases} 0 & u < 3 \\ u - 3 & u \in [3, 4) \\ 5 - u & u \in [4, 5) \\ 0 & u \geq 5 \end{cases}, \\
N_4^1(u) &= \begin{cases} 0 & u < 4 \\ u - 4 & u \in [4, 5) \\ (7 - u)/2 & u \in [5, 7) \\ 0 & u \geq 7 \end{cases}, & N_5^1(u) &= \begin{cases} 0 & u < 5 \\ (u - 5)/2 & u \in [5, 7) \\ 1 & u \geq 7 \end{cases}.
\end{aligned}$$

Las funciones de grado dos,

$$\begin{aligned}
 N_0^2(u) &= \begin{cases} 1 & u < 0 \\ 1 - u^2/3 & u \in [0, 1) \\ (u - 3)^2/6 & u \in [1, 3) \\ 0 & u \geq 3 \end{cases}, \\
 N_1^2(u) &= \begin{cases} 0 & u < 0 \\ u^2/3 & u \in [0, 1) \\ -u^2/3 + 4u/3 - 2/3 & u \in [1, 3) \\ (u - 4)^2/3 & u \in [3, 4) \\ 0 & u \geq 4 \end{cases}, \\
 N_2^2(u) &= \begin{cases} 0 & u < 1 \\ (u - 1)^2/6 & u \in [1, 3) \\ -5u^2/6 + 17u/3 - 53/6 & u \in [3, 4) \\ (u - 5)^2/2 & u \in [4, 5) \\ 0 & u \geq 5 \end{cases}, \\
 N_3^2(u) &= \begin{cases} 0 & u < 3 \\ (u - 3)^2/2 & u \in [3, 4) \\ -5u^2/6 + 23u/3 - 101/6 & u \in [4, 5) \\ (u - 7)^2/6 & u \in [5, 7) \\ 0 & u \geq 7 \end{cases}, \\
 N_4^2(u) &= \begin{cases} 0 & u < 4 \\ (u - 4)^2/3 & u \in [4, 5) \\ -u^2/6 + 7u/3 - 43/6 & u \in [5, 7) \\ 1 & u \geq 7 \end{cases}.
 \end{aligned}$$

Y finalmente las de grado tres, que son las más comunes,

$$\begin{aligned}
N_0^3(u) &= \begin{cases} 1 & u < 0 \\ 1 - u^3/12 & u \in [0, 1) \\ u^3/12 - u^2/2 + u/2 + 5/6 & u \in [1, 3) \\ (4 - u)^3/12 & u \in [3, 4) \\ 0 & u \geq 4 \end{cases} , \\
N_1^3(u) &= \begin{cases} 0 & u < 0 \\ u^3/12 & u \in [0, 1) \\ -u^3/8 + 5u^2/8 - 5u/8 + 5/24 & u \in [1, 3) \\ 7u^3/24 - 25u^2/8 + 85u/8 - 265/24 & u \in [3, 4) \\ (5 - u)^3/8 & u \in [4, 5) \\ 0 & u \geq 5 \end{cases} , \\
N_2^3(u) &= \begin{cases} 0 & u < 1 \\ (u - 1)^3/24 & u \in [1, 3) \\ 13u^2/4 - u^3/3 - 10u - 121/12 & u \in [3, 4) \\ u^3/3 - 19u^2/4 + 22u - 391/12 & u \in [4, 5) \\ (7 - u)^3/24 & u \in [5, 7) \\ 0 & u \geq 7 \end{cases} , \\
N_3^3(u) &= \begin{cases} 0 & u < 3 \\ (u - 3)^3/8 & u \in [3, 4) \\ -5u^3/24 + 23u^2/8 - 101u/8 + 431/24 & u \in [4, 5) \\ u^3/24 - 7u^2/8 + 49u/8 - 319/24 & u \in [5, 7) \\ 1 & u \geq 7 \end{cases} .
\end{aligned}$$

13. Propiedades de las funciones B-spline

Consideremos funciones polinómicas a trozos definidas en un intervalo $[u_0, u_m]$, con partición $\{u_0, \dots, u_m\}$, de modo que en cada intervalo $[u_i, u_{i+1}]$ las funciones son polinomios de grado n y en cada nudo u_i son de clase C^{n-r_i} . Estas funciones forman un espacio lineal. Vamos a calcular su dimensión.

La dimensión del espacio de polinomios de grado n en el primer intervalo, $[u_0, u_1]$, es $n + 1$. El intervalo $[u_1, u_2]$ aportaría también $n + 1$ a la dimensión, pero hay que restarle las $n - r_1 + 1$ condiciones de continuidad y diferenciabilidad en u_i . Repitiendo el proceso hasta llegar al intervalo final, $[u_{m-1}, u_m]$,

obtenemos

$$\dim = n + 1 + \sum_{i=1}^{m-1} r_i. \quad (35)$$

En el caso que nos interesa, N tramos, reales o no, grado n , tenemos una sucesión de nudos $\{u_0, \dots, u_{2n+N-2}\}$ y las funciones spline están definidas sobre el intervalo $[u_{n-1}, u_{n+N-1}]$. Los nudos pueden estar repetidos. Como sabemos, si un nudo interior, u_i , está repetido r_i veces, los splines serán de clase C^{n-r_i} en dicho nudo. Por tanto, en la expresión (35) la suma será el número de nudos interiores, estén repetidos o no, desde u_n hasta u_{n+N-2} , es decir, el número de tramos menos uno, $N - 1$.

Así pues, la dimensión del espacio de funciones spline de grado n y N tramos es $\dim = n + N = 1 + L$, es decir, coincide con el número de vértices de los polígonos B-spline.

Esto está de acuerdo con la idea de que podemos codificar el algoritmo de De Boor en unas funciones $\{N_0^n(u), \dots, N_L^n(u)\}$, base de dicho espacio, de modo que, si el polígono de una curva es $\{d_0, \dots, d_L\}$,

$$c(u) = \sum_{i=0}^L d_i N_i^n(u). \quad (36)$$

Vamos a obtener los polígonos B-spline de las funciones nodales. Para conseguir esto, debemos saber primero cómo construir funciones en vez de curvas paramétricas. Volvemos, pues, al asunto de la precisión lineal, cómo expresar el monomio lineal u en nuestro formalismo. Para ello, hacemos uso de la forma polar. Si queremos expresarla con polinomios de grado uno, la respuesta es bien sencilla,

$$u = c(u) = d[u],$$

y para grados superiores, sólo tenemos que elevar el grado de la forma polar hasta llegar a n . En el tema segundo se dedujo una fórmula de elevación del grado reiterada,

$$d^{n-1}[v_1, \dots, v_n] = \frac{(n-1)!}{n!} \sum_{i=1}^n d[v_i] = \frac{1}{n} \sum_{i=1}^n v_i, \quad (37)$$

que aquí particularizamos para $n - 1$ elevaciones a partir de grado uno.

Por tanto, si queremos conocer el polígono B-spline, $\{\xi_0, \dots, \xi_{n+N-1}\}$, del monomio lineal con una sucesión de nudos $\{u_0, \dots, u_{2n+N-2}\}$, grado n y N tramos, sólo tenemos que evaluar la forma polar,

$$\xi_i = d^{n-1}[u_i, \dots, u_{i+n-1}] = \frac{1}{n} \sum_{j=i}^{i+n-1} u_j. \quad (38)$$

A estos valores medios de los nudos se los conoce como **abscisas de Greville**.

Por tanto, si expresamos una función polinómica a trozos, $f(u)$, como una gráfica, $(u, f(u))$, los vértices de su polígono B-spline serán de la forma (ξ_i, \hat{d}_i) , donde \hat{d}_i son los vértices de la “curva” paramétrica $f(u)$.

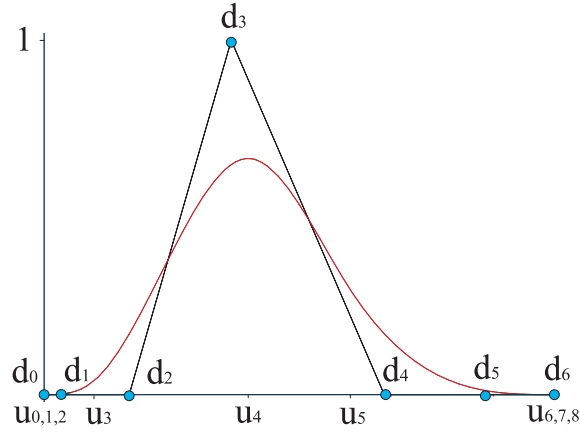


Figura 22: Relación entre una función nodal de grado tres y su polígono B-spline

Con esta información, construimos las funciones nodales o B-spline, $N_i^n(u)$, de grado sobre la sucesión de nudos $\{u_0, \dots, u_{2n+N-2}\}$ como aquellas cuya gráfica tiene polinomio B-spline $\{d_0^i, \dots, d_{n+N-1}^i\}$, donde $d_j^i = (\xi_j, \delta_j^i)$, es decir, la ordenada de todos los vértices es nula, salvo la correspondiente al índice i .

Así pues, $\{\delta_0^i, \dots, \delta_{n+N-1}^i\}$ es el polígono de la función nodal $N_i^n(u)$. Por tanto $\{\sum d_i \delta_0^i, \dots, \sum d_i \delta_{n+N-1}^i\} = \{d_0, \dots, d_{n+N-1}\}$ es a la vez el polígono de $c(u)$ y de $\sum d_i N_i^n(u)$, luego hemos llegado a la expresión (36).

Las funciones B-spline son polinómicas a trozos. Consideremos la gráfica de la función $N_i^n(u)$ sobre una sucesión de nudos $\{u_0, \dots, u_{2n+N-2}\}$. Los

vértices de su polígono B-spline son de la forma (ξ_j, δ_j^i) , todos con ordenada nula, salvo el vértice i -ésimo $(\xi_i, 1)$, así que la función será nula salvo en los tramos controlados por dicho vértice.

Como ya se ha visto, el vértice d_i controla $n + 1$ tramos, los que van desde el $(i - n + 1)$ -ésimo hasta el $(i + 1)$ -ésimo, que están definidos sobre los intervalos $[u_{i-1}, u_i]$ hasta el $[u_{i+n-1}, u_{i+n}]$. Por tanto el soporte, el intervalo cerrado fuera del cual la función $N_i^n(u)$ se anula, es $[u_{i-1}, u_{i+n}]$.

Una propiedad importante de las funciones nodales es que **el soporte es mínimo**. Cualquier función con soporte más corto que el de una función B-spline es necesariamente nulo. Vamos a razonarlo.

Consideremos el espacio de las funciones polinómicas a trozos sobre el intervalo $[u_{i-1}, u_{i+n}]$. De acuerdo con la fórmula (35), es un espacio de dimensión $2n + 1$. Pero hay que descontar las condiciones de anulación de las derivadas en u_{i-1}, u_{i+n} . Si fuesen simples, las funciones serían de clase C^{n-1} en los extremos y habría que restar $2n$ grados de libertad, con lo cual el espacio de funciones spline considerado tendría dimensión uno, es decir, todas las funciones son múltiplos de una dada, la $N_i^n(u)$ y, por tanto, todas ellas, salvo la nula, tienen el mismo soporte. Si consideráramos un soporte un intervalo menor, tendríamos $2n - 2n$ grados de libertad y las funciones serían idénticamente nulas, por lo que no hay funciones spline no nulas con soporte más pequeño que el de las funciones nodales.

El razonamiento es válido asimismo en el caso de que u_{i-1} o u_{i+n} no sean simples. Ciertamente, al aumentar la multiplicidad de uno de ellos, por ejemplo $u_{i-1} = u_i$, en una unidad, tendremos una condición menos de anulación de derivadas y, por tanto, un grado de libertad adicional. Pero también supondría que el soporte sería un tramo más corto, $[u_i, u_{i+n}]$, y también perderíamos un grado de libertad. Por tanto, la cuenta de dimensiones final no se modifica. \square

Estamos ahora en condiciones de demostrar que las funciones B-spline son **linealmente independientes**. Supongamos que tenemos una combinación lineal de funciones nodales idénticamente nula,

$$\sum_{j=0}^{n+N-1} d_j N_j^n(u) \equiv 0,$$

con coeficientes reales, d_i .

Lo demostramos restringiéndonos al intervalo $[u_i, u_{i+1}]$. Considerando únicamente las funciones nodales cuyo soporte contiene dicho intervalo, la con-

dición se reduce a

$$\sum_{j=i-n+1}^{i+1} d_j N_j^n(u) = 0, \quad u \in [u_i, u_{i+1}].$$

Esta suma define una función spline en el intervalo $[u_{i-n}, u_{i+n+1}]$ que se anula en $[u_i, u_{i+1}]$. Pero dicha función habrá de ser nula en todo su dominio, ya que en caso contrario debería ser, o bien no nula en $[u_{i+1}, u_{i+n+1}]$ o bien en $[u_{i-n}, u_i]$ y ambos son demasiado cortos, n intervalos en vez de $n+1$, para soportar un spline no nulo.

Ahora bien, la única función nodal en la suma que tiene soporte en el intervalo $[u_{i-n}, u_{i-n+1}]$ es $N_{i-n+1}^n(u)$, luego d_{i-n+1} debe anularse. Iterando este proceso llegamos a la conclusión de que todos los d_j son nulos y las funciones nodales son linealmente independientes.

En el caso de que un nudo sea múltiple y sean varias las funciones nodales que no se anulen en el intervalo, el razonamiento es el mismo, ya que las funciones serán de diferente clase de diferenciabilidad y no se puede obtener una función de una clase como combinación lineal de funciones de clase superior. \square

Por tanto, como tenemos tantas funciones nodales linealmente independientes como dimensión tiene el espacio de funciones polinómicas a trozos, concluimos que las funciones nodales forman **base** de dicho espacio.

Una propiedad importante de los polinomios de Bernstein, fuente de numerosas propiedades de la curvas de Bézier era que formaban una **partición de la unidad**. Esta propiedad se mantiene en la base de funciones nodales,

$$\sum_{i=0}^{n+N-1} N_i^n(u) = 1, \quad u \in [u_{n-1}, u_{n+N-1}], \quad (39)$$

como se comprueba trivialmente describiendo una recta horizontal mediante un polígono de De Boor con todas las d_i iguales a uno. \square

14. Splines racionales

[Vídeo de splines racionales](#)

Como ya se ha comentado, la aplicación del algoritmo de De Boor a las curvas racionales sólo supone la introducción de una coordenada auxiliar, que da lugar al denominador de la parametrización.

Del mismo modo, una curva racional a trozos se puede expresar en términos de las funciones nodales sin más que introducir un denominador. Así pues, una curva racional de grado n a trozos y N tramos está determinada por los vértices de su polígono B-spline, $\{d_0, \dots, d_{n+N-1}\}$, sus correspondientes pesos, $\{w_0, \dots, w_{n+N-1}\}$, y una sucesión de nudos, $\{u_0, \dots, u_{2n+N-2}\}$,

$$c(u) = \frac{\sum_{i=0}^{n+N-1} w_i d_i N_i^n(u)}{\sum_{i=0}^{n+N-1} w_i N_i^n(u)}.$$

Todos los procesos que hemos descrito para curvas polinómicas a trozos, tales como inserción de nudos, elevación del grado u obtención de polígonos de control, se trasladan automáticamente a curvas racionales, sin más que introducir la coordenada auxiliar y luego proyectar.

Dada la propiedad de localidad de las curvas spline, la modificación de un peso altera tan sólo a $n + 1$ tramos de la curva. **Ejemplo.** Se mantienen las propiedades de las curvas racionales y, por supuesto, las de las curvas polinómicas a trozos.

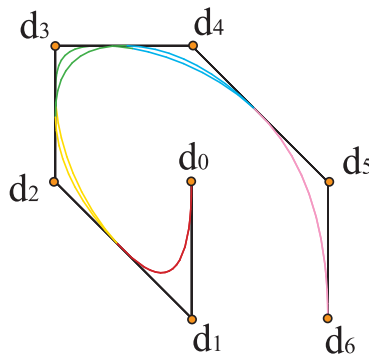


Figura 23: Spline racional cuadrático: al modificar el peso w_3 , se alteran los tramos del segundo al cuarto

Como curiosidad, no podemos representar una circunferencia completa como una cónica B-spline, ya que los vértices del polígono serían aquellos que determinan las tangentes y, por simetría de la curva, deberían ser iguales todos sus pesos, por lo cual la curva no sería una circunferencia, sino una parábola a trozos. El mismo argumento se aplica a las cúbicas a trozos.

15. Teorema de Holladay

El nombre inglés *spline* que se asigna a las curvas polinómicas a trozos proviene de la palabra junquillo, que designa el objeto flexible que tradicionalmente se empleaba en el diseño para trazar curvas. La relación entre el junquillo y las curvas polinómicas nos la proporciona el Teorema de Holladay.

La posición del junquillo, descrita por una curva $c(s)$, viene determinada por el principio de mínima energía elástica, la cual es proporcional al cuadrado de su curvatura, κ ,

$$E[c] = \int_a^b ds \kappa^2(s) = \int_a^b ds \|\ddot{c}(s)\|^2, \quad (40)$$

en la parametrización por longitud de arco.

Las ecuaciones variacionales que describen los mínimos de esta energía son bastante complicadas, por lo cual probaremos una aproximación,

$$\tilde{E}[c] = \int_a^b du \|c''(u)\|^2, \quad (41)$$

en nuestra parametrización polinómica a trozos, que solamente será apropiada cuando se aparte poco de la parametrización por longitud de arco, ya que en general la curvatura de una curva parametrizada viene descrita por

$$\kappa(u) = \frac{\|c'(u) \times c''(u)\|}{\|c'(u)\|^3}.$$

Sólo cuando $\|c'(u)\|$ tome valores próximos a la unidad, nuestra aproximación será válida.

No obstante, vamos a comprobar que un spline cúbico de clase C^2 , $c(u)$, minimiza la aproximación de la energía, \tilde{E} .

Consideremos que $c(u)$ resuelve el problema de interpolación $c(u_i) = a_i$, $i = 0, \dots, N$, $c'(u_0) = m_0$, $c'(u_N) = m_N$. Podemos escribir cualquier otra curva parametrizada $x(u)$ que verifique estas condiciones,

$$x(u) = c(u) + (x(u) - c(u)),$$

de manera práctica para nuestros fines. La aproximación a la energía elástica de $x(u)$ será

$$\begin{aligned} \tilde{E}[x] &= \int_{u_0}^{u_N} du \|c''(u)\|^2 + \int_{u_0}^{u_N} du \|x''(u) - c''(u)\|^2 \\ &+ 2 \int_{u_0}^{u_N} du c''(u) \cdot (x''(u) - c''(u)), \end{aligned}$$

con lo cual habremos probado que el spline cúbico minimiza la energía, $\tilde{E}[x] \geq \tilde{E}[c]$ si demostramos que el término del producto escalar se anula, ya que el segundo término es positivo. Integrando por partes,

$$\int_{u_0}^{u_N} du c''(u) \cdot (x''(u) - c''(u)) = [c''(u) \cdot (x'(u) - c'(u))]_{u_0}^{u_N} - \int_{u_0}^{u_N} du c'''(u) \cdot (x'(u) - c'(u)),$$

el primer término se anula por la condición sobre las derivadas en los extremos, $c'(u_0) = m_0 = x'(u_0)$, $c'(u_N) = m_N = x'(u_N)$. En cuanto al segundo, lo podemos desglosar en los intervalos $[u_i, u_{i+1}]$,

$$\int_{u_0}^{u_N} du c'''(u) \cdot (x'(u) - c'(u)) = \sum_{i=0}^{N-1} c'''(u) \cdot \int_{u_i}^{u_{i+1}} du (x'(u) - c'(u)),$$

en cada uno de los cuales $c'''(u)$ es constante, por ser cúbica, y sale fuera de la integral,

$$\int_{u_i}^{u_{i+1}} du (x'(u) - c'(u)) = [x(u) - c(u)]_{u_i}^{u_{i+1}} = 0,$$

que se anula puesto que $c(u_i) = a_i = x(u_i)$. \square