Trazo de líneas curvas

Valdivia Gomez Victor Rafael

Numero de control: 18290936

Curvas spline

-¿Qué se conoce como Spline?

Un spline es una curva diferenciable definida en porciones mediante polinomios.

-Tanto en dibujo como en computación gráfica, ¿qué es una curva spline?

Una curva spline es una secuencia de segmentos de curva que se conectan entre sí para formar una sola curva continua.

-¿Qué es una superficie spline?

Es una función spline que tiene el mínimo soporte con respecto a un determinado grado, suavidad y partición del dominio

-¿Qué usos tienen las curvas y superficies splines o para qué se utilizan?

Las curvas de Bézier han sido ampliamente usadas en los gráficos generados por ordenador para modelado de curvas suaves. Como la curva está completamente contenida en la envolvente convexa de los puntos de control, dichos puntos pueden ser visualizados gráficamente sobre el área de trabajo y usados para manipular la curva de una forma muy intuitiva. Las transformaciones afines tales como traslaciones y rotaciones pueden ser aplicadas, con gran facilidad, a las curvas, aplicando las transformaciones respectivas sobre los puntos de control.

-Tipos de splines, dar una breve descripción de estos

- M-spline:
  -
- I-spline:
  -
- B-spline:
  - Una B-spline o Basis spline, es una función spline que tiene el mínimo soporte con respecto a un determinado grado, suavidad y partición del dominio. Un teorema fundamental establece que cada función spline de un determinado grado, suavidad y partición del dominio, se puede representar como una combinación lineal de B-splines del mismo grado y suavidad, y sobre la misma partición.

-¿Cómo se especifica una curva spline?

Una curva de spline se especifica a partir de un conjunto de posiciones de coordenadas, que se conocen como puntos de control, los cuales indican la forma general de la curva. Dado un conjunto de puntos de control, los métodos de interpolación generan una curva que pasa por todos los puntos de control.

-¿Qué se entiende por punto de control?

Un punto en el cual al ser modificado se modifica los demás puntos, seria algo asi como el punto de referencia.

-¿A qué se refiere el control interpolado y a que se refiere el control aproximado? y ¿qué diferencias hay ente éstos?

El interpolado hace referencia a un cálculo matemático más complejo que el de aproximación.

-¿Cómo se manipula una curva spline y cómo una superficie spline?

Mediante un punto de control en la suprerfice xd.

-¿Qué grado tiene una curva spline?

Existen Lineales, Cuadraticas y de orden superior

-¿Qué elementos matemáticos definen una curva spline?

Una curva spline es una secuencia de segmentos de curva que se conectan entre sí para formar una sola curva continua. Por ejemplo, una colección de trozos de curvas, en las que los extremos inicial y final están conectados, puede ser llamada una curva spline.

-¿Qué se entiende por armazón convexo o capsula convexa? y ¿qué representa dentro de la curva spline?

Puede ser el como se conforma la curva del spline.

-¿Qué es un grafo de control y cuál es su uso o que nos representa?

Una gráfica de control es un diagrama que sirve para examinar si un proceso se encuentra en una condición estable, o para asegurar que se mantenga en esa condición.

-¿Qué es una condición de continuidad?

La continuidad es la existencia de un camino completo para el flujo de la corriente. Un interruptor cerrado que está en funcionamiento, por ejemplo, tiene continuidad.

-¿Qué es una continuidad paramétrica?

Permite representar una curva o superficie en el plano o en el espacio, mediante valores que recorren un intervalo de números reales, mediante una variable, llamada parámetro, considerando cada coordenada de un punto como una función dependiente del parámetro.

-¿Qué es una continuidad de cuarto orden?

Lo mismo de la continuidad normal, pero de orden cuatro

-¿Qué es una continuidad geométrica?

No sé, derivadas geométricas tal vez.

-¿Qué es una continuidad geométrica de cuarto orden?

No sé, derivadas geométricas tal vez pero de cuatro órdenes.

-Describir los métodos utilizados para especificar una representación de un spline

Existen varias clases de especificaciones de spline que se usan las aplicaciones gráficas. Cada especificación difiere del resto en el polinomio particular que utiliza junto a las condiciones de frontera o condiciones que se requiere. Recordemos que una función polinómica de n-esimo se define como:

$$y = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1} + a_n x^n$$

Donde n es un entero no negativo y ak son constantes. Siendo an ≠ 0 tienes, n=2 la función es cuadráticas, si n=3 la función es un polinomio, si n=1 tenemos una línea recta.

-¿Qué es una curva B-Spline?

Una B-spline o Basis spline, es una función spline que tiene el mínimo soporte con respecto a un determinado grado, suavidad y partición del dominio. Un teorema fundamental establece que cada función spline de un determinado grado, suavidad y partición del dominio, se puede representar como una combinación lineal de B-splines del mismo grado y suavidad, y sobre la misma partición.

-¿Cómo se genera una curva B-Spline?

Dado m valores reales t_i, llamamos nodos, con

$$t_0 \leq t_1 \leq \cdots \leq t_{m\text{-}1}$$

Un B-spline de grado n es una curva paramétrica

$$S: [t_0, t_{m\text{-}1}] \rightarrow \mathbb{R}^2$$

Compuesta por una combinación línea de B-splines básicas $b_{i,n}$ de grado n

$$S(t) = \sum_{i=0}^{m\text{-}n\text{-}2} P_i b_{i,n}(t), t \in [t_{n\text{-}1}, t_{m\text{-}n}]$$

-En las B-Spline ¿qué es un nudo y qué es un vector de nudos?

Los nudos son puntos que son equidistantes entre si.

El vector de nudos es una secuencia de valores de parámetros que determina dónde y cómo afectan los puntos de control a la curva B-spline

-Propiedades y características de una curva B-Spline

La función B-spline es una combinación de bandas flexibles que atraviesa el número de puntos que se denominan puntos de control y crea curvas suaves. Estas funciones permiten la creación y gestión de formas y superficies complejas utilizando una serie de puntos. La función B-spline y las funciones Bézier se aplican ampliamente en los métodos de optimización de formas.

Un B-spline de orden es una función polinomial a trozos de grado en una variable . Se define sobre ubicaciones , llamadas nudos o puntos de ruptura, que deben estar en orden no descendente . El B-spline contribuye solo en el rango entre el primero y el último de estos nudos y es cero en otros lugares. Si cada nudo está separado por la misma distancia (dónde) de su predecesor, el vector de nudos y las B-splines correspondientes se denominan "uniformes" (consulte el B-spline cardinal a continuación).

$$\mathrm{nn\text{-}1x1} + \mathrm{nt_j t_j} \leq \mathrm{t_{j+1} hh} = \mathrm{t_{j+1}\text{-}t_j}$$

-Describir brevemente los diferentes tipos de curvas B-Spline que hay

B-spline uniforme: Cuando la B-spline es uniforme, las B-splines básicas para un determinado grado n son solo copias cambiadas de una a otra.

B-spline cardinal: Un B-spline cardinal tiene una separación constante, h , entre nudos. Los B-splines cardinales para un orden dado n son solo copias desplazadas entre sí.

-¿Qué elementos matemáticos definen una curva B-spline?

Una función polinomial pr partes de grado variable.

-En una curva B-spline, explicar las diferencias entre control global y control local

No lo sé, espero haberte ayudado, pero si me puedo inventar algo, diría que el control local es para unos puntos de la curva y el control global para toda la curva?

-¿Por qué una curva B-spline tiene control local?

Para determina la continuidad de la curva, creo

-¿Por qué las curvas B-spline tienen la propiedad de cápsula convexa (o armazón convexo)? ¿Qué implicación tiene esto?

Para poder hacer la curva, que se yo, si no le respondi la misma pregunta antes, para que volverla a hacer.

-¿Cuándo una curva spline se convierte en curva de Bézier?

Cuando esta cumple las condiciones adecuadas.

Curvas de Bézier

-¿Qué es un spline de Bezier o una curva de Bezier?

Se denomina curvas de Bézier a un sistema que se desarrolló hacia los años 1960 para el trazado de dibujos técnicos, en el diseño aeronáutico y en el de automóviles. Su denominación es en honor a Pierre Bézier, quien ideó un método de descripción matemática de las curvas que se comenzó a utilizar con éxito en los programas de CAD.

-¿Qué elementos matemáticos definen una curva de Bezier?

El polinomio de Bernstein

$$\mathbf{B}(t) = \sum_{i=0}^{n} \binom{n}{i} \mathbf{P}_i (1-t)^{n-i} t^i = \mathbf{P}_0 (1-t)^n + \binom{n}{1} \mathbf{P}_1 (1-t)^{n-1} t + \cdots + \mathbf{P}_n t^n \text{ , } t \in [0,1].$$

-¿Qué características tiene una curva de Bezier?

Es flexibe, manipulable fácil de usar, creo que a eso se refiere.

-¿Dónde o para qué se utilizan las curvas de Bezier?

En los graficos vectoriales para la animación, dibujo gráfico, computación grafica.

-¿Qué especificaciones sigue para su trazado?

$$\mathbf{B}(t) = \sum_{i=0}^{n} \binom{n}{i} \mathbf{P}_i (1-t)^{n-i} t^i = \mathbf{P}_0 (1-t)^n + \binom{n}{1} \mathbf{P}_1 (1-t)^{n-1} t + \cdots + \mathbf{P}_n t^n \ , t \in [0,1].$$

-Propiedades de una curva de Bezier

- La curva comienza en P_0 y termina en P_n ; esta es la denominada propiedad de interpolación de punto final .
- La curva es una línea recta si y solo si todos los puntos de control son colineales .
- El inicio y el final de la curva es tangente a la primera y última sección del polígono de Bézier, respectivamente.
- Una curva se puede dividir en cualquier punto en dos subcurvas, o en muchas subcurvas arbitrariamente, cada una de las cuales es también una curva de Bézier. Algunas curvas que parecen simples, como el círculo , no pueden describirse exactamente mediante una curva Bézier o una curva Bézier a trozos ; aunque una curva de Bézier cúbica de cuatro piezas puede aproximarse a un círculo, con un error radial máximo de menos de una parte en mil, cuando cada punto de control interno (o punto fuera de línea) es la distancia horizontal o vertical desde un punto de control exterior en un círculo unitario. De manera más general, una curva de Bézier cúbica de n piezas puede aproximarse a un círculo, cuando cada punto de control interno es la distancia desde un punto de control externo en un círculo unitario, donde t es 360 / n grados y n > 2 .
- Cada curva de Bézier cuadrática es también una curva de Bézier cúbica y, de manera más general, cada curva de grado n de Bézier es también una curva de grado m para cualquier m > n . En detalle, una curva de grado n con puntos de control P_0 , ..., P_n es equivalente (incluida la parametrización) a la curva de grado n + 1 con puntos de control P'_0 , ..., P'_{n+1} , donde .

$$\mathbf{P}'_k = \tfrac{k}{n+1}\mathbf{P}_{k-1} + \left(1 - \tfrac{k}{n+1}\right)\mathbf{P}_k$$

- Las curvas de Bézier tienen la propiedad de disminuir la variación . Lo que esto significa en términos intuitivos es que una curva de Bézier no "ondula" más que el polígono de sus puntos de control, y en realidad puede "ondular" menos que eso.

- No hay control local en las curvas de grado n Bézier, lo que significa que cualquier cambio en un punto de control requiere un nuevo cálculo y, por lo tanto, afecta el aspecto de toda la curva, "aunque cuanto más lejos esté uno del punto de control que se cambió, menor es el cambio en la curva ".

- Una curva de Bézier de orden superior a dos puede cruzarse o tener una cúspide para ciertas elecciones de los puntos de control.

-Técnicas de diseño que siguen o hacen uso del trazado de curvas de Bezier

Los gráficos por computadora

-¿Por qué una curva de Bézier tiene control global?

Porque no tiene control local, y de ahuevo que se debe tener algún tipo de control, en este caso global.

-¿Por qué el primer y último punto de la curva de Bézier coinciden con el primer y último punto del grafo de control?

Para poder cerrar la curva y que no explote tu dibujo.

-¿Por qué las curvas de Bézier tienen la propiedad de cápsula convexa? (O sea, la curva se mantiene dentro de la cápsula convexa del grafo de control.) ¿Qué implicación tiene esto?

Para un mejor manejo de la curva.

-La curva de Bézier usualmente no pasa por los puntos de control (salvo el primero y el último). ¿Por qué?

Porque solo tiene control global, no puede modificar una curva de Bézier de su punto medio a no ser que dividas la curva en dos, y ahora modificaría el principio o final de una de las dos curvas.

-¿Qué grado tiene la curva de Bézier?

Es de grado n, donde n > 0.

-¿Cómo es la relación de grado de una curva de Bezier con respecto a sus puntos de control?

Son puntos globales, al modificar ese punto se modifica toda la curva.

-¿Qué diferencias y ventajas muestran las curvas B-Spline con respecto a una curva de Bezier?

B-Spline ofrece más control y flexibilidad que las curvas Bezier.

-Conversión de un B-Spline a una curva de Bezier y viceversa

Encontre este articulo que no voy a transcribir porque es mucho y ya estoy cansado de la vida.

http://web.archive.org/web/20180714225212/http://www.infogoaround.org/JBook/bstobez.html

Bibliografica

B-spline - B-spline - qaz.wiki. (s. f.). qaz.wiki. Recuperado 23 de octubre de 2020, de https://es.qaz.wiki/wiki/B-spline#Cardinal_B-spline

colaboradores de Wikipedia. (2020, 9 agosto). B-spline. Wikipedia, la enciclopedia libre. https://es.wikipedia.org/wiki/B-spline

Universidad Nacional autónoma de México (México). (s. f.). CONCEPTUALIZACIÓN: CURVAS B-SPLINE. PDF. Recuperado 23 de octubre de 2020, de http://funes.uniandes.edu.co/11653/1/Ramos2016Conceptualizacion.pdf

colaboradores de Wikipedia. (2020, 25 agosto). Curva de Bézier. Wikipedia, la enciclopedia libre. https://es.wikipedia.org/wiki/Curva_de_B%C3%A9zier

Curva de Bézier - Bézier curve - qaz.wiki. (s. f.). qaz.wiki. Recuperado 23 de octubre de 2020, de https://es.qaz.wiki/wiki/B%C3%A9zier_curve#:%7E:text=%20Curva%20de%20B%C3%A9zier%20-%20B%C3%A9zier%20curve%20,cualquier%20grado%20n%20.%20Una%20definici%C3%B3n...%20More%20

```
Este es el metodo principal para dibujar la curva de Bezier.
   public void dibujarBezier(Graphics g, ArrayList<Point> puntos, int numeroPunto
s) {

        //Variables para almacenar el punto calculado
        double puntoX = 0, puntoY = 0;
        //En esta lista se guardan los puntos anteriores que nos sirven para pode
r
        //dibujar las líneas que van de punto a punto
        ArrayList<int[]> anteriores = new ArrayList<>();
        double avance = 1 / ((double) numeroPuntos);
        int tamLista = puntos.size();

        //Este ciclo realiza el número de iteraciones que el usuario desee en bas
e
        //al número de puntos que se desean
        for (double u = 0; u <= 1; u += avance) {
            //Este ciclo itera sobre los puntos a dibujar y realiza el cálculo de
l siguiente punto de la curva
            for (int k = 0; k < tamLista; k++) {
                double b = calcularB(u, tamLista - 1, k);
                puntoX += puntos.get(k).getX() * b;
                puntoY += puntos.get(k).getY() * b;
            }

            //Se almacenan y se dibuja el punto calculado anteriormente
            anteriores.add(new int[] {(int)puntoX, (int) puntoY});
            g.drawLine((int) puntoX, (int) puntoY, (int) puntoX, (int) puntoY);

            //Se dibuja la línea que va del punto anterior al recien calculado
            //de esa manera no tenemos que calcular punto por punto de toda la cu
rva
            if (anteriores.size() > 1) {
                g.drawLine(anteriores.get(anteriores.size() - 2)[0], anteriores.g
et(anteriores.size() - 2)[1], (int) puntoX, (int) puntoY);
                //Removemos el primer elemento que ya no se utiliza
                anteriores.remove(0);
            }

            //Borramos los valores anteriores para la siguiente iteración
            puntoX = puntoY = 0;
        }

        //Se dibuja el último trayecto de la curva
```

```
        g.drawLine((int) anteriores.get(anteriores.size() - 1)[0], (int) anteriores.get(anteriores.size() - 1)[1],
                (int) puntos.get(tamLista - 1).getX(), (int) puntos.get(tamLista - 1).getY());

    }
```

```
private double calcularB(double u, int n, int k) {

        return (factorial(n) / (factorial(k) * factorial(n - k))) *
                Math.pow(u, k) * Math.pow(1 - u, n - k);

    }
```

Este es el método principal para dibujar un B-spline

```
public void dibujarCardinal(Graphics g, ArrayList<Point> puntos, int numeroPuntos) {

    double a, b, c, d, u, x, y;
    ArrayList<Point> anteriores = new ArrayList<>();
    double tension = slider.getValue();

    if (puntos.size() >= 4) {
        for (int k = 1; k < (puntos.size() - 2); k++) {
            for (double i = 0; i < numeroPuntos; i++) {
                u = (double) (i / numeroPuntos);
                a = (2 * u * u * u) - (3 * u * u) + 1;
                b = (-2 * u * u * u) + (3 * u * u);
                c = (u * u * u) - (2 * u * u) + u;
                d = (u * u * u) - (u * u);
                double[] derivadas = calcularDerivadas(puntos.get(k + 1), puntos.get(k), puntos.get(k + 2), puntos.get(k - 1), tension);

                x = (a * ((double) puntos.get(k).getX())) + (b * ((double) puntos.get(k + 1).getX())) +
                        (c * ((double) derivadas[0])) + (d * ((double) derivadas[2]));
                y = (a * ((double) puntos.get(k).getY())) + (b * ((double) puntos.get(k + 1).getY())) +
                        (c * ((double) derivadas[1])) + (d * ((double) derivadas[3]));

                g.drawLine((int) x, (int) y, (int) x, (int) y);
```

```java
                anteriores.add(new Point((int) x, (int) y));

                if (anteriores.size() > 1) {
                    g.drawLine(anteriores.get(anteriores.size() - 2).x, anteriore
s.get(anteriores.size() - 2).y, (int) x, (int) y);
                    anteriores.remove(0);
                }
            }
        }

        g.drawLine(anteriores.get(anteriores.size() - 1).x, anteriores.get(anteri
ores.size() - 1).y,
            puntos.get(puntos.size() - 2).x, puntos.get(puntos.size() - 2).y);
    }

}
```