Tarea 5 - Problema 2 - Programación avanzada

Nicolás Aylwin

1. Desarrollo teórico

Asumimos que se desean interpolar $n+1$ puntos, siendo el primer punto (x_0, y_0) y el último (x_n, y_n). Nuestra función Spline sera una función a trozos de tal manera:

$$S(x) = \begin{cases} C_1(x) & , \quad x_0 \leq x \leq x_1 \\ \vdots & \\ C_i(x) & , \quad x_{i-1} < x \leq x_i \\ \vdots & \\ C_n(x) & , \quad x_{n-1} < x \leq x_n \end{cases}$$

El desafio resulta ser encontrar los n polinomios $C_i(x)$ que componen al Spline. Las tres condiciones basicas son:

1) $C_i(x_{i-1}) = y_{i-1} \quad \wedge \quad C_i(x_i) = y_i \quad \forall i \in \{1, \dots, n\}$

2) $C_i'(x_i) = C_{i+1}'(x_i) \quad \forall i \in \{1, \dots, n-1\}$

3) $C_i''(x_i) = C_{i+1}''(x_i) \quad \forall i \in \{1, \dots, n-1\}$

La primera condición da cuenta de que el spline es continuo para cada punto. Las otras dos dan cuentan de continuidad en la primera y la segunda derivada (así se fuerza la suavidad del spline).

Como buscamos splines cúbicos, cada $C_i(x)$ es un polinomio cúbico, por lo que por cada $C_i(x)$ hay cuatro incógnitas. Se concluye que hay $4n$ incógnitas a determinar, y las tres condiciones enunciadas nos entregan $n+n+(n-1)+(n-1) = 4n-2$ ecuaciones. Las otras dos las obtenemos de la condición de periodicidad. Sean estas dos ecuaciones faltantes:

1) $C_1'(x_0) = C_n'(x_n)$

2) $C_1''(x_0) = C_n''(x_n)$

Ahora definamos la segunda derivada evaluada en alguno de los puntos donde deseamos interpolar:

$$M_i = S''(x_i) \quad \forall i \in \{0, \dots, n\}$$

En particular se tiene:

$$\begin{aligned} &S''(x_i) = C_i''(x_i) = C_{i+1}''(x_i) = M_i \quad \forall i \in \{1, \dots, n-1\} \\ &S''(x_0) = C_1''(x_0) = M_0 \\ &S''(x_n) = C_n''(x_n) = M_n \end{aligned}$$

Para determinar cada C_i es cúbico, cada C_i'' es lineal. Ahora interpolemos $C_i''(x)$ entre x_{i-1} y x_i según lagrange. Se tiene:

$$C_i''(x) = C_i''(x_{i-1})\frac{(x-x_i)}{x_{i-1}-x_i} + C_i''(x_i)\frac{(x-x_{i-1})}{x_i-x_{i-1}}$$

Usando los M_i recién definidos y si definimos $h_i = x_i - x_{i-1}$ la expresión anterior queda:

$$C_i''(x) = M_{i-1}\frac{(x_i-x)}{h_i} + M_i\frac{(x-x_{i-1})}{h_i}$$

Integrando una vez:

$$C_i'(x) = -M_{i-1}\frac{(x_i-x)^2}{2h_i} + M_i\frac{(x-x_{i-1})^2}{2h_i} + A$$

Integrando nuevamente:

$$C_i(x) = M_{i-1}\frac{(x_i-x)^3}{6h_i} + M_i\frac{(x-x_{i-1})^3}{6h_i} + Ax + B$$

Si usamos la condición 1) sobre la continuidad del spline para los puntos (x_{i-1}, y_{i-1}) y (x_i, y_i), se pueden obtener las constantes A y B. Luego con álgebra podemos llegar a la siguiente expresión para cada C_i:

$$C_i(x) = M_{i-1}\frac{(x_i-x)^3}{6h_i} + M_i\frac{(x-x_{i-1})^3}{6h_i} + \left(y_{i-1} - \frac{M_{i-1}h_i^2}{6}\right)\frac{(x_i-x)}{h_i} + \left(y_i - \frac{M_i h_i^2}{6}\right)\frac{(x-x_{i-1})}{h_i} \tag{1}$$

En esta ecuación las únicas incógnitas son las M_i, por lo que si obtenemos estas está resuelto el problema. Para esto derivemos la Ec.(1) y evaluemosla en x_i:

$$C_i'(x) = -M_{i-1}\frac{(x_i-x)^2}{2h_i} + M_i\frac{(x-x_{i-1})^2}{2h_i} + \frac{y_i-y_{i-1}}{hi} - \frac{(M_i-M_{i-1})h_i}{6}$$
$$C_i'(x_i) = M_i\frac{h_i}{2} + \frac{y_i-y_{i-1}}{hi} - \frac{(M_i-M_{i-1})h_i}{6}$$

Esta misma evaluación en el siguiente polinomio será:

$$C_{i+1}'(x_i) = -M_i\frac{h_{i+1}}{2} + \frac{y_{i+1}-y_i}{h_{i+1}} - \frac{(M_{i+1}-M_i)h_{i+1}}{6}$$

Y por las condiciones tenemos que:

$$C_{i+1}'(x_i) = C_i'(x_i)$$
$$M_i\frac{h_i}{2} + \frac{y_i-y_{i-1}}{hi} - \frac{(M_i-M_{i-1})h_i}{6} = -M_i\frac{h_{i+1}}{2} + \frac{y_{i+1}-y_i}{h_{i+1}} - \frac{(M_{i+1}-M_i)h_{i+1}}{6}$$

Y reordenando esta última expresión para juntar todas las M correspondientes, si además definimos $k_i = y_i - y_{i-1}$ tendremos finalmente:

$$\frac{h_i}{6}M_{i-1} + \frac{h_i+h_{i+1}}{3}M_i + \frac{h_{i+1}}{6}M_{i+1} = \frac{k_{i+1}}{h_{i+1}} - \frac{k_i}{h_i} \tag{2}$$

Esta última expresión es valida para $i \in \{1, \ldots, n-1\}$ claramente. Para $i = 1$ y $i = n-1$, se necesita conocer el valor de M_0 y M_n respectivamente. Utilicemos las condiciones mencionadas sobre la periodicidad para obtener estos:

$$\begin{gathered} C_1''(x_0) = C_n''(x_n) \\ M_0 = M_n \end{gathered}$$

Esto último es directo de la definición de M_i. De la otra condición:

$$\begin{gathered} C_1'(x_0) = C_n'(x_n) \\ -M_0\frac{h_1}{2} + \frac{k_1}{h_1} - \frac{(M_1 - M_0)}{6}h_1 = M_n\frac{h_n}{2} + \frac{k_n}{h_n} + \frac{(M_n - M_{n-1})}{6}h_n \\ -\frac{M_0 h_1}{3} + \frac{k_1}{h_1} - \frac{M_1 h_1}{6} = \frac{M_n h_n}{3} + \frac{k_n}{h_n} + \frac{M_{n-1} h_n}{6} \end{gathered}$$

Considerando $M_0 = M_n$, la resolución de la última expresión queda:

$$M_0 = M_n = \frac{3}{(h_1 + h_n)}\left(\frac{k_1}{h_1} - \frac{k_n}{h_n} - \frac{h_1 M_1}{6} - \frac{h_n M_{n-1}}{6}\right)$$

Siendo así, volvamos a la Ec.(2) con $i = 1$ y $i = n-1$ (esta es la primera fila y última fila de la matriz respectivamente):

$$\begin{gathered} \frac{h_1}{6}M_0 + \frac{(h_1 + h_2)}{3}M_1 + \frac{h_2}{6}M_2 = \frac{k_2}{h_2} - \frac{k_1}{h_1} \\ \frac{h_{n-1}}{6}M_{n-2} + \frac{(h_{n-1} + h_n)}{3}M_{n-1} + \frac{h_n}{6}M_n = \frac{k_n}{h_n} - \frac{k_{n-1}}{h_{n-1}} \end{gathered}$$

Usando la expresión sobre M_0 y M_n, además de reagrupar términos se llega a:

$$\left(\frac{(h_1 + h_2)}{3} - \frac{h_1^2}{12(h_1 + h_n)}\right)M_1 + \frac{h_2}{6}M_2 - \frac{h_1 h_n}{12(h_1 + h_n)}M_{n-1} = \\ \frac{k_2}{h_2} - \frac{k_1}{h_1} + \frac{k_n h_1}{2h_n(h_1 + h_n)} - \frac{k_1}{2(h_1 + h_n)}$$

$$-\frac{h_1 h_n}{12(h_1 + h_n)}M_1 + \frac{h_{n-1}}{6}M_{n-2} + \left(\frac{(h_{n-1} + h_n)}{3} - \frac{h_n^2}{12(h_1 + h_n)}\right)M_{n-1} = \\ \frac{k_n}{h_n} - \frac{k_{n-1}}{h_{n-1}} - \frac{k_1 h_n}{2h_1(h_1 + h_n)} + \frac{k_n}{2(h_1 + h_n)}$$

Donde una expresión es la primera fila de la matriz y la otra es la última. Teniendo estos valores basta expresar la matriz resultante, que de haber entregado $n+1$ puntos quedará de $n-1$ filas y misma cantidad de columnas, con el vector de incógnitas conteniendo M_i con $i \in \{1, \ldots, n-1\}$. Se puede notar que en la primera y en la última fila aparece un termino extra en las esquinas (las dos esquinas que antes eran cero, con splines naturales) y que los valores al otro lado de la ecuación para la primera y la última fila son mucho mas complejos que para todas las otras.

2. Programa

El programa tiene dependencia de una clase de interpoladores (interpol.h e interpol.cc), una clase de polinomios en templates (poli.h) y una clase de matrices de templates (matriz.h). En la de interpoladores, se tiene el objeto “i_pol”, que almacena los puntos entregados en una matriz y además la cantidad de datos entregados (por comodidad en los métodos). Se definen dos métodos, lagrange y splines. Este último tiene una opción para hacerlos periódicos, puesto solo cambia la primera y la última fila de la matriz (que se agregan a parte), el primer y último valor del vector al otro lado de la ecuación, y los valores de M_0 y M_n. Como la matriz a resolver resulta ser tridiagonal periódica se implemento en la clase de matrices (y se utilizó en la de interpoladores) el método ANW (ahlberg-nilsen-walsh) para resolver esta.

El método de splines devuelve una matriz de polinomios (los $C_i(x)$). Con una función externa a la clase (pero dentro de interpol.cc) llamada “puntos” se permite evaluar ordenadamente cada polinomio donde corresponda, y se devuelve una matriz con los datos evaluados. Luego con otra funcion definida en el mismo .cc, llamada ‘file’ se logra crear un archivo con tales datos y de cierto nombre.

El programa en si requerido solo cuenta con main, donde se recibe el archivo, la cantidad de puntos, se crea el interpolador, se aplica spline con el char ‘p’ para que sean periodicos, se usa la función puntos, luego file, y se acaba el programa.

3. Resultados y compilación

La carpeta contiene un archivo datos.txt de prueba y un script de gnuplot llamado grafico.gp. Al realizar make se compilara y se utilizará el programa con el archivo de prueba y 1000 puntos. Luego se correra el script de gnuplot y se creara una imagen de formato png, donde se muestra el spline y los puntos de datos.txt. Por último se abre la imagen recién creada.

De desear probar con otros datos basta usar el ejecutable como se pide en la tarea.