

Métodos de integración numérica para particiones no uniformes del intervalo de integración

Alessandri Canchoa Q. ¹

Resumen

En el presente trabajo de investigación se proponen métodos de integración numérica cuando se cuentan con valores de la función integrando en una partición no uniforme del intervalo integración. Aplicando el método Newton-Cotes, los splines cúbicos natural y los splines cúbicos libre se obtienen tres métodos, se verifica la eficiencia de los dos últimos métodos mediante un análisis de convergencia.

Palabras clave:

Abstract

Presently investigation work intends methods of numerical integration when they are had values of the function integrating in a partition it doesn't standardize of the integration interval. Applying the method Newton-Cotes, the natural cubic splines and the free cubic splines three methods are obtained, efficiency of the last two methods is verified by means of a convergence analysis.

Key Words:

1. Introducción

En el presente trabajo se trata el problema de aproximar una integral definida $\int_a^b f(x)dx$, para particiones no uniformes del intervalo de integración [a;b]. La mayoría de fórmulas de cuadratura de integración numérica exigen que la función f sea conocido en una partición uniforme del intervalo [a;b] o que f sea conocido en los ceros de un polinomio ortonormal específico para el caso de las fórmulas de cuadratura de Gauss. En muchas aplicaciones por algunos motivos tales como por los instrumentos de medición, sólo se cuentan con valores de la función sobre una partición no uniforme del intervalo de integración y no es posible aplicar los métodos clásicos de integración numérica. Este trabajo surge con la finalidad de contar con una herramienta para resolver este tipo de problemas que aparecen con frecuencia en las aplicaciones reales.

Una primera tentativa para el caso de particiones no uniforme sería aplicar cuadraturas de Newton-Cotes para subintervalos los cuales están basados en la interpolación polinomial. Se muestra que con esta propuesta se tienen problemas con la acumulación de errores de redondeo para una cantidad grande de nodos en la partición donde se espera un resultado de aproximación aceptable. Otra alternativa que se plantea es utilizar una interpolación de splines cúbicos, los cuales poseen un mejor comportamiento respecto de la interpolación polinomial y se pueden trabajar con nodos discretos no necesariamente igualmente espaciados. Para este caso se obtienen resultados de convergencia.

Se proponen métodos de integración numérica para el caso donde se cuentan con valores de la función integrando en una partición no uniforme del intervalo de integración.

También se proponen los respectivos algoritmos para su implementación.

2. Revisión de literatura

En este capítulo se presentan los conceptos básicos, definiciones, notaciones y algunos resultados que serán requeridos posteriormente.

2.1. Interpolación

Se trata del problema

- Aproximar una función que es muy difícil calcular con funciones simples.
- Encontrar una función simple para valores de una tabla.

El cálculo general para la solución de este problema consiste en encontrar una función $g(x, \alpha_0, \dots, \alpha_r)$, de la cual se eligen los parámetros tal que la función aproxime los valores de la tabla lo mejor posible.

La función puede ser un polinomio $g(x, \alpha_0, \dots, \alpha_r) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \dots + \alpha_r x^r$ para funciones periódicas se podría elegir

$$g(x, \alpha_0, \dots, \alpha_r) = \alpha_0 + \alpha_1 \text{Sen}(\alpha_2)x + \alpha_3 \text{Cos}(\alpha_4)x + \dots + \alpha_{r-1} \text{Cos}(\alpha_r)x,$$

para oscilaciones amortiguadas

$$g(x, \alpha_0, \dots, \alpha_r) = \alpha_0 + \alpha_1 e^{\alpha_2 x} \text{Sen}(\alpha_3)x + \alpha_4 e^{\alpha_5 x} \text{Cos}(\alpha_6)x + \dots + \alpha_{r-2} e^{\alpha_{r-1} x} \text{Cos}(\alpha_r)x,$$

para una función con polos

$$g(x, \alpha_0, \dots, \alpha_r) = \frac{\alpha_0 + \alpha_1 x + \dots + \alpha_s x^s}{\alpha_{s+1} x^{s+1} + \dots + \alpha_r x^r},$$

como un función racional fraccionaria.

Se puede determinar $\alpha_0, \dots, \alpha_r$, tal que la función g tenga la propiedad que el promedio de la desviación sea mínimo. En este caso se trata de la aproximación lineal o no lineal.

Otro método es, exigir que g tenga en las abscisas x_0, \dots, x_r los mismos valores como la función aproximada, en este caso se trata de una interpolación.

¹ Profesor Principal, Facultad de Ciencias, Departamento de Matemática de la UNALM. Lima, Perú. E-mail: canchoa@yahoo.com.

Definición 1.1

Se dice que una función $f : [a, b] \rightarrow \mathbb{R}$ es una interpolación o que interpola a un conjunto de datos $\{(x_i, y_i)\}_{i=0}^n$ si se cumple

$$f(x_i) = y_i, \quad i = 0, 1, \dots, n.$$

Por ejemplo, en el caso de una interpolación polinomial para los datos $\{(x_i, y_i)\}_{i=0}^n$, se busca un polinomio de grado $\leq n$.

$$P_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

tal que

$$P_n(x_i) = f_i, \quad i = 0, \dots, n.$$

Para la interpolación polinomial existen las fórmulas de Lagrange y de Newton que han sido muy bien estudiadas.

En ésta sección se va a desarrollar la interpolación por splines cúbicos, que es una interpolación polinomial por segmentos y que satisfacen ciertas propiedades muy favorables.

Al implementar este método se requiere resolver un sistema tridiagonal, a continuación se desarrolla éste tipo de interpolación.

1.2. Splines cúbicos

Dados los puntos interpolante $a = x_0 < x_1 < \dots < x_n = b$ en el intervalo $[a; b]$ con los valores correspondientes f_0, f_1, \dots, f_n utilizando la interpolación por splines cúbicos, se unen los puntos $(x_0, f_0), (x_1, f_1), \dots, (x_n, f_n)$ obteniendo una función spline cúbico S_Δ^3

a) S_Δ^3 es un polinomio de grado 3 en cada intervalo $[x_j, x_{j+1}]$, $(j = 0, 1, \dots, n-1)$

b) $S_\Delta^3(x_j) = f_j$, $j = 0, 1, \dots, n$.

c) $S_\Delta^3 \in C^2[a, b]$.

1.2.1. Construcción de un Spline cúbico

La condición (a) implica que S_Δ^3 se puede expresar como

$$S_\Delta^3(x) = P_j(x), \quad x_j \leq x \leq x_{j+1}, \quad j = 0, \dots, n-1,$$

donde

$$P_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3,$$

(1)

De esto se requieren determinar los $4n$ coeficiente a_j, b_j, c_j y $d_j, j=0, 1, 2, \dots, n-1$.

Por la condición c) se cumple

$$(S_\Delta^3)''(x_j) = 2c_j, \quad j = 0, 1, \dots, n,$$

Como la función

$$(S_\Delta^3)''(x) = c_j + 6d_j(x - x_j), \quad x \in [x_j, x_{j+1}]$$

es afín lineal se cumple

$$(S_\Delta^3)''(x) = 2c_j \frac{x_{j+1} - x}{h_{j+1}} + 2c_{j+1} \frac{x - x_j}{h_{j+1}}, \quad \text{para } x \in [x_j, x_{j+1}],$$

(2)

donde

$$h_{j+1} = x_{j+1} - x_j.$$

Integrando para $j=0, 1, \dots, n-1$ se cumplen

$$(S_\Delta^3)'(x) = -c_j \frac{(x_{j+1} - x)^2}{h_{j+1}} + c_{j+1} \frac{(x - x_j)^2}{h_{j+1}} + D_j, \quad \text{para } x \in [x_j, x_{j+1}]$$

(3)

$$S_\Delta^3(x) = c_j \frac{(x_{j+1} - x)^3}{3h_{j+1}} + c_{j+1} \frac{(x - x_j)^3}{3h_{j+1}} + D_j(x - x_j) + E_j, \quad \text{para } x \in [x_j, x_{j+1}]$$

De las condiciones interpolantes

$$S_\Delta^3(x_j) = f_j \text{ y } S_\Delta^3(x_{j+1}) = f_{j+1}, \quad \text{se}$$

obtienen

$$c_j \frac{h_{j+1}^2}{3} + E_j = f_j$$

$$c_{j+1} \frac{h_{j+1}^2}{3} + D_j h_{j+1} + E_j = f_{j+1}$$

de (1) en (4) se deduce

$$E_j = f_j - c_j \frac{h_{j+1}^2}{3}$$

$$D_j = \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{h_{j+1}}{3} (c_{j+1} - c_j) \quad (5)$$

De esto se obtiene

$$a_j = f_j$$

$$b_j = (S_\Delta^3)'(x_j) = -c_j h_{j+1} + D_j = \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{(2c_j + c_{j+1})h_{j+1}}{3}$$

$$d_j = \frac{1}{6} (S_\Delta^3)'''(x_j) = \frac{c_{j+1} - c_j}{3h_{j+1}}$$

Luego los coeficientes a_j, b_j , y d dependen de c_j ; esto significa que la función S_Δ^3 está caracterizado por los coeficientes c_j . Por la continuidad de $(S_\Delta^3)'$ en los nodos $x=x_j$, para $j=1, 2, \dots, n-1$ proporcionan las siguientes $n-1$ ecuaciones.

$$(S_\Delta^3)'(x_j^-) = (S_\Delta^3)'(x_j^+), \quad j = 1, 2, \dots, n-1$$

Reemplazando D_j y E_j dados por (5) en (3), se obtiene

$$(S_\Delta^3)'(x) = -c_j \frac{(x_{j+1} - x)^2}{h_{j+1}} + c_{j+1} \frac{(x - x_j)^2}{h_{j+1}} + \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{h_{j+1}}{3} (c_{j+1} - c_j).$$

Para $j=1, 2, \dots, n-1$, se obtienen

$$(S_\Delta^3)'(x_j^-) = \frac{f_j - f_{j-1}}{h_j} + \frac{2h_j}{3} c_j + \frac{h_j}{3} c_{j-1}$$

$$(S_\Delta^3)'(x_j^+) = \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{2h_{j+1}}{3} c_j - \frac{h_{j+1}}{3} c_{j+1},$$

$$\text{Como } (S_\Delta^3)'(x_j^-) = (S_\Delta^3)'(x_j^+),$$

para $j=1, 2, \dots, n-1$ se obtiene

$$h_j c_{j-1} + 2(h_j + h_{j+1}) c_j + h_{j+1} c_{j+1} = 3 \left(\frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j-1}}{h_j} \right)$$

De esto se deduce

$$\frac{h_j}{h_j + h_{j+1}} c_{j-1} + 2c_j + \frac{h_{j+1}}{h_j + h_{j+1}} c_{j+1} = \frac{3}{h_j + h_{j+1}} \left(\frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j-1}}{h_j} \right),$$

(6)

Definiendo

$$v_j = \frac{h_j}{h_j + h_{j+1}},$$

$$w_j = 1 - v_j = \frac{h_{j+1}}{h_j + h_{j+1}},$$

$$g_j = \frac{3}{h_j + h_{j+1}} \left(\frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j-1}}{h_j} \right),$$

(7)

Se obtiene

$$v_j c_{j-1} + 2c_j + w_j c_{j+1} = g_j, j = 1, 2, \dots, n-1.$$

(8)

En total se tienen n-1 ecuaciones y n+1 incógnitas. Para completar y tener n+1 ecuaciones se requieren agregar dos condiciones.

Estas condiciones se imponen de diferentes formas

i) Imponiendo las restricciones

$(S_{\Delta}^3)'(x_0) = 0 = (S_{\Delta}^3)'(x_n)$, el Spline cúbico que se obtiene se denomina Spline natural.

ii) Para la interpolación de una función periódica mediante los splines cúbicos es conveniente exigir.

$$(S_{\Delta}^3)'(x_0) = (S_{\Delta}^3)'(x_n) \text{ y } (S_{\Delta}^3)''(x_0) = (S_{\Delta}^3)''(x_n)$$

iii) Otra posibilidad es exigir

$(S_{\Delta}^3)'(x_0) = m_0$ y $(S_{\Delta}^3)'(x_n) = m_1$, donde los números m_0 y m_1 son dados.

Splines cúbicos natural o Splines natural

Usando la condición para los splines natural, de (i) se tiene

$$(S_{\Delta}^3)''(x_0) = 2c_0 = 0, \quad (S_{\Delta}^3)''(x_n) = 2c_n = 0$$

De esto se obtiene $c_0=c_n=0$, luego el sistema tridiagonal con (n-1) ecuaciones con (n-1) incógnitas c_1, c_2, \dots, c_{n-1} .

$$\begin{pmatrix} 2 & w_1 & 0 & \dots & \dots & 0 \\ v_2 & 2 & w_2 & \dots & \dots & \vdots \\ 0 & v_3 & 2 & \dots & \dots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & 0 \\ \vdots & \vdots & \vdots & \vdots & 2 & w_{n-2} \\ 0 & \dots & \dots & 0 & v_{n-1} & 2 \end{pmatrix} \cdot \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ \vdots \\ c_{n-1} \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \\ \vdots \\ \vdots \\ g_{n-1} \end{pmatrix}$$

(9)

Este sistema lineal tiene la forma $Ac=g$.

Proposición 1.2 La matriz $A \in \mathbb{R}^{(n-1) \times (n-1)}$ es no singular para cualquier partición del intervalo $[a, b]$.

Prueba. Para

$$x = (x_1, \dots, x_{n-1})^T, b = (b_1, \dots, b_{n-1})^T \in \mathbb{R}^{(n-1)}$$

$$Ax = b \implies \|x\|_{\infty} \leq \|b\|_{\infty}$$

se cumple

Sea p tal que

$$|x_p| = \max\{|x_i| : i = 1, 2, \dots, n-1\} = \|x\|_{\infty}$$

Como $v_p + w_p = 1$, se obtiene.

$$\begin{aligned} \|b\|_{\infty} &= \max\{|b_i| : i = 1, 2, \dots, n-1\} \\ &\geq |b_p| \\ &= |v_p x_{p-1} + 2x_p + w_p x_{p+1}| \\ &\geq 2|x_p| - v_p |x_{p-1}| - w_p |x_{p+1}| \\ &\geq 2|x_p| - v_p |x_p| - w_p |x_p| \\ &\geq (2 - v_p - w_p) |x_p| \\ &= 1 \times |x_p| = \|x\|_{\infty}. \end{aligned}$$

Supongamos que la matriz A es singular, luego el sistema $Ax=0$ tiene una solución $x \neq 0$ que

significaría, $0 < \|x\|_{\infty} \leq \|0\|_{\infty} = 0$, lo cual es una contradicción. Por consiguiente la matriz A es no singular.

Este resultado implica que el sistema (9) tiene una única solución. A continuación se propone un algoritmo para determinar los coeficientes de un spline cúbico natural.

función $[a, b, c, d] = \text{SplineCúbicoN}(x, f, n)$
% Entrada: $x := (x_0, x_1, \dots, x_n)$, $f := (f_0, f_1, \dots, f_n)$, con $x_j \neq x_k$ para $j \neq k$
% Interpola los puntos: $\{(x_0, f_0), (x_1, f_1), \dots, (x_n, f_n)\}$ con el Spline cúbico natural,
% $S_{\Delta}^3(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3$, $x_j \leq x \leq x_{j+1}$, $j = 0, \dots, n-1$
% Salida : $a = (a_0, \dots, a_{n-1})$, $b = (b_0, \dots, b_{n-1})$, $c = (c_0, \dots, c_{n-1})$, $d = (d_0, \dots, d_{n-1})$.
$a_0 := f_0$
Para $j := 1, \dots, n$
$h_j := x_j - x_{j-1}$
$a_j := f_j$.
Para $j := 1, 2, \dots, n-1$.
$g_j := \frac{3}{h_j + h_{j+1}} \left(\frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j-1}}{h_j} \right)$
$v_j = \frac{h_j}{h_j + h_{j+1}}$
$w_j := 1 - v_j$
$s_j := 2$
$s_n := 2$
$c_0 := 0$
$c_n := 0$
$(c_1, c_2, \dots, c_{n-1}) := \text{tridiagonal}(v, s, v, g, n-1)$
Para $j := 0, 1, \dots, n-1$
$d_j := \frac{c_{j+1} - c_j}{3h_{j+1}}$
$b_j = \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{h_{j+1}}{3}(c_{j+1} + 2c_j)$

función $[a, b, c, d] = \text{SplineCúbicoN}(x, f, n)$
% Entrada: $x := (x_0, x_1, \dots, x_n), f := (f_0, f_1, \dots, f_n)$, con $x_j \neq x_k$ para $j \neq k$
% Interpola los puntos: $\{(x_0, f_0), (x_1, f_1), \dots, (x_n, f_n)\}$ con el Spline cúbico natural,
% $S_{\Delta}^3(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3, x_j \leq x \leq x_{j+1}, j = 0, \dots, n - 1$
% Salida : $a = (a_0, \dots, a_{n-1}), b = (b_0, \dots, b_{n-1}), c = (c_0, \dots, c_{n-1}), d = (d_0, \dots, d_{n-1})$.
$a_0 := f_0$
Para $j := 1, \dots, n$
$h_j := x_j - x_{j-1}$
$a_j := f_j,$
Para $j := 1, 2, \dots, n - 1$.
$g_j := \frac{3}{h_j + h_{j+1}} \left(\frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j-1}}{h_j} \right)$
$v_j = \frac{h_j}{h_j + h_{j+1}}$
$w_j := 1 - v_j$
$s_j := 2$
$s_n := 2$
$c_0 := 0$
$c_n := 0$
$(c_1, c_2, \dots, c_{n-1}) := \text{tridiagonal}(v, s, v, g, n - 1)$
Para $j := 0, 1, \dots, n - 1$
$d_j := \frac{c_{j+1} - c_j}{3h_{j+1}}$
$b_j = \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{h_{j+1}}{3}(c_{j+1} + 2c_j)$

En este algoritmo, se requiere la función tridiagonal (.) para solucionar un sistema tridiagonal, el cual se desarrolla en la siguiente subsección. Este y los algoritmos que se presentan en el presente trabajo han sido implementados con el software MATLAB, con el cual se han obtenidos las .guras y resultados que son presentados en este trabajo. En la Figura 1) se muestra un ejemplo de una interpolación mediante un spline cúbico natural.

Splines Cúbicos Libre

En los splines cúbicos naturales exigimos $(S_{\Delta}^3)''(x_0) = 0 = (S_{\Delta}^3)''(x_n)$, esto significa que si la función $f''(x)$ no esta cerca de cero en $x=x_0$ y $x=x_n$ se estará introduciendo un error. Para dejar libre el comportamiento en la frontera del intervalo $[x_0, x_n]$, se puede exigir la continuidad de $(S_{\Delta}^3)'''(x)$ en los puntos x_1 y x_{n-1} . A este tipo de splines cúbicos que se obtienen con estas restricciones lo denominaremos splines cúbicos libre.

De (2) se obtiene

$$(S_{\Delta}^3)'''(x) = \frac{2(c_{j+1} - c_j)}{h_{j+1}}, \text{ para } x \in [x_j, x_{j+1}],$$

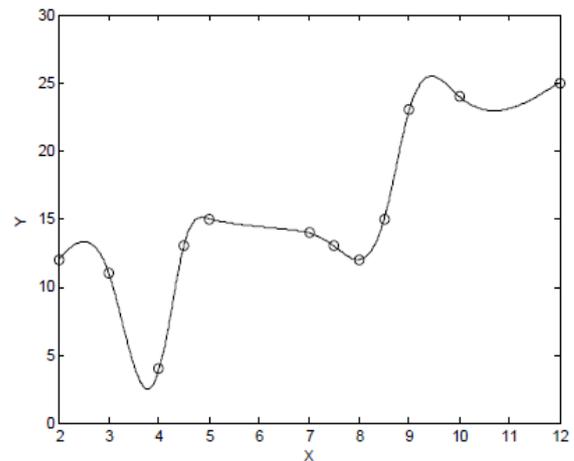
(10)

Exigiendo continuidad de $(S_{\Delta}^3)'''(x)$ en los puntos x_1 y x_{n-1} , se debe cumplir

$$(S_{\Delta}^3)'''(x_1^-) = (S_{\Delta}^3)'''(x_1^+) \text{ y } (S_{\Delta}^3)'''(x_{n-1}^-) = (S_{\Delta}^3)'''(x_{n-1}^+)$$

, luego reemplazando en

Figura 1: Interpolación mediante una función spline cúbico natural.



$$\frac{2(c_1 - c_0)}{h_1} = \frac{2(c_2 - c_1)}{h_2},$$

(11)

$$\frac{2(c_{n-1} - c_{n-2})}{h_{n-1}} = \frac{2(c_n - c_{n-1})}{h_n}.$$

Por otro lado de (6) y (7) se tiene

$$\frac{h_1}{h_1 + h_2}c_0 + 2c_1 + \frac{h_2}{h_1 + h_2}c_2 = g_1,$$

(12)

$$\frac{h_{n-1}}{h_{n-1} + h_n}c_{n-2} + 2c_{n-1} + \frac{h_n}{h_{n-1} + h_n}c_n = g_{n-1}.$$

(13)

Se (11) se obtienen

$$c_0 = \left(\frac{h_1+h_2}{h_2} \right) c_1 - \frac{h_1}{h_2} c_2, \quad (14)$$

$$c_n = \left(\frac{h_{n-1}+h_n}{h_{n-1}} \right) c_{n-1} - \frac{h_n}{h_{n-1}} c_{n-2}. \quad (15)$$

De (12) y (14) se obtiene

$$\left(2 + \frac{h_1}{h_2} \right) c_1 + \left(1 - \frac{h_1}{h_2} \right) c_2 = g_1. \quad (16)$$

De (13) y (15) se obtiene

$$\left(1 - \frac{h_n}{h_{n-1}} \right) c_{n-2} + \left(2 + \frac{h_n}{h_{n-1}} \right) c_{n-1} = g_{n-1} \quad (17)$$

Definiendo

$$v_{n-1} := 1 - \frac{h_n}{h_{n-1}}, \quad v_j := \frac{h_j}{h_j + h_{j+1}}, \quad j = 2, 3, \dots, n-2$$

$$w_1 := 1 - \frac{h_1}{h_2}, \quad w_j := 1 - v_j = \frac{h_{j+1}}{h_j + h_{j+1}}, \quad j = 2, 3, \dots, n-2$$

función $[a, b, c, d] = \text{SplineCúbicoL}(x, f, n)$
% Entrada: $x := (x_0, x_1, \dots, x_n), f := (f_0, f_1, \dots, f_n)$, con $x_j \neq x_k$ para $j \neq k$
% Interpola los puntos: $\{(x_0, f_0), (x_1, f_1), \dots, (x_n, f_n)\}$ con el Spline Cúbico Libre,
% $S_{\Delta}^3(x) = a_j + b_j(x-x_j) + c_j(x-x_j)^2 + d_j(x-x_j)^3, x_j \leq x \leq x_{j+1}, j = 0, \dots, n-1$
% Salida : $a = (a_0, \dots, a_{n-1}), b = (b_0, \dots, b_{n-1}), c = (c_0, \dots, c_{n-1}), d = (d_0, \dots, d_{n-1})$.
$a_0 := f_0$
Para $j := 1, \dots, n$
$h_j := x_j - x_{j-1}$
$a_j := f_j$
Para $j := 1, 2, \dots, n-1$.
$g_j := \frac{3}{h_j + h_{j+1}} \left(\frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j-1}}{h_j} \right)$
$v_j := \frac{h_j}{h_j + h_{j+1}}$
$w_j := 1 - v_j$
$s_j := 2$
$s_1 := 2 + \frac{h_1}{h_2}$
$s_{n-1} := 2 + \frac{h_n}{h_{n-1}}$
$v_{n-1} := 1 - \frac{h_n}{h_{n-1}}$
$w_1 := 1 - \frac{h_1}{h_2}$
$(c_1, c_2, \dots, c_{n-1}) := \text{tridiagonal}(v, s, v, g, n-1)$
$c_0 := \left(\frac{h_1+h_2}{h_2} \right) c_1 - \frac{h_1}{h_2} c_2$
$c_n := \left(\frac{h_{n-1}+h_n}{h_{n-1}} \right) c_{n-1} - \frac{h_n}{h_{n-1}} c_{n-2}$
Para $j := 0, 1, \dots, n-1$
$d_j := \frac{c_{j+1} - c_j}{3h_{j+1}}$
$b_j := \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{h_{j+1}}{3} (c_{j+1} + 2c_j)$

1.3 El algoritmo compuesto para sistemas triagonales

Consideremos el sistema triagonal

$$\begin{aligned} a_1 x_1 + c_1 x_2 &= f_1 \\ b_2 x_1 + a_2 x_2 + c_2 x_3 &= f_2 \\ b_3 x_2 + a_3 x_3 + c_3 x_4 &= f_3 \\ \vdots &\vdots \\ b_{n-1} x_{n-2} + a_{n-1} x_{n-1} + c_{n-1} x_n &= f_{n-1} \\ b_n x_{n-1} + a_n x_n &= f_n \end{aligned}$$

Para solucionar este sistema triagonal, se factoriza la matriz triagonal de coeficiente A en la forma LU y luego se debe solucionar los sistemas $Ly=b$ y $Ux=y$, uno a continuación del otro. Siguiendo estas ideas se

función $x = \text{tridiagonal}(b, a, c, f, n)$
% Entrada: Matriz triagonal $A := \text{tri}(b, a, c)$
% $b = (b_1, b_2, \dots, b_{n-1}), a = (a_1, a_2, \dots, a_n), c = (c_1, c_2, \dots, c_{n-1})$,
% No se usa b_1
% Salida: Solución $(x_1, \dots, x_n)^T$ del sistema triagonal $Ax = f$
$l_1 := a_1,$
$u_1 := c_1/a_1$
Para $j := 2, \dots, n-1$
$l_j := a_j - b_j u_{j-1}$
$u_j := c_j/l_j$
$l_n := a_n - b_n u_{n-1}$
$y_1 := f_1/l_1$
$j = 2, \dots, n$
$y_j := (f_j - b_j y_{j-1})/l_j$
Poner $x_n := y_n$
Para $j = n-1, \dots, 1$
$x_j := y_j - u_j x_{j+1}$

obtiene el siguiente algoritmo.

1.4. Fórmulas Clásicas de Integración Numérica

Para el cálculo aproximado de integrales definidas de la forma

$$I(f) := \int_a^b f(x) dx, \quad f: [a, b] \rightarrow R$$

, se usan fórmulas de cuadratura de la forma

$$K_n(f) = \sum_{j=0}^n \alpha_j f(x_j), \quad \text{donde las}$$

constantes α_j son llamados pesos de integración.

1.4.1. Fórmulas de Newton - Cotes:

Consideramos una partición del intervalo

$$[a, b]: a = x_0 < x_1 < \dots < x_n = b,$$

un resultado conocido del análisis numérico es que existe un único polinomio interpolante $P_n(x)$ de grado menor o igual que n con $P_n(x_0) = f_0; P_n(x_1) = f_1, \dots, P_n(x_n) = f_n$. El polinomio $P_n(x)$ en la forma de Lagrange, se expresa como sigue

$$P_n(x) = \sum_{j=0}^n f(x_j)L_j(x), \quad \text{con } L_j(x) = \prod_{\substack{k=0 \\ k \neq j}}^n \frac{x - x_k}{x_j - x_k}, \quad j = 0, \dots, n.$$

Utilizando este polinomio como aproximación para

$$K_n(f) = \int_a^b P_n(x)dx = \sum_{j=0}^n \int_a^b L_j(x)dx \cdot f(x_j) = \sum_{j=0}^n \alpha_j f(x_j)$$

donde

$$\alpha_j := \int_a^b L_j(x)dx \quad j = 0, 1, \dots, n.$$

Definición 1.3 La formula de cuadratura de Newton-Cotes está definido por

$$K_n(f) = \sum_{j=0}^n \alpha_j f(x_j), \quad \text{con } \alpha_j = \int_a^b L_j(x)dx, \quad j = 0, \dots, n$$

Donde $\alpha_j = \int_a^b L_j(x)$ son los pesos de integración y con el error de integración

$$E_n(f) := I(f) - K_n(f) = \int_a^b f(x)dx - \sum_{j=0}^n \alpha_j f(x_j)$$

Para polinomios P_k de grado menor o igual a n se cumple $E_n(P_k) = 0$. Como el polinomio interpolante es único de esto se obtiene el siguiente resultado.

Teorema 1.4 Para $a = x_0 < x_1 < \dots < x_n = b$, existe una única fórmula de cuadratura la cual es exacta para todos los polinomios de grado menor o igual a n .

$$K_n(f) = \sum_{j=0}^n \alpha_j f(x_j), \quad \text{con } \alpha_j = \int_a^b L_j(x), \quad j = 0, \dots, n,$$

Para una partición uniforme del intervalo $[a; b]$ realizando un cambio de variables en la integral correspondiente a los pesos, se obtiene el siguiente resultado.

Teorema 1.5 Para los puntos de partición equidistantes

$$x_j := a + jh, \quad j = 0, \dots, n, \quad h = \frac{b-a}{n}$$

del intervalo $[a;b]$, los pesos de integración están dados por

$$\alpha_j^{(n)} = h \frac{(-1)^{n-j}}{j!(n-j)!} \int_0^n \prod_{\substack{k=0 \\ k \neq j}}^n (t - k) dt.$$

Utilizando la fórmula para el error de una función interpolante se obtiene

$$E_n(f) = I(f) - K_n(f) = \int_a^b (f(x)dx - P_n(x))dx = \int_a^b \frac{f^{(n+1)}(\xi_x)}{(n+1)!} \rho_n(x)dx,$$

donde

$$\rho_n(x) = \prod_{i=0}^n (x - x_i) \text{ y algún } \xi_x \in]a, b[.$$

A partir de estos resultados se obtienen las siguientes reglas:

a) Regla trapezoidal (n=1):

$$K_1(f) = \frac{b-a}{2} [f(a) + f(b)]$$

$$E_1(f) = -\frac{1}{12} h^3 f''(\xi), \quad h = b - a$$

b) Regla de Simpson (n=2):

$$K_2(f) = \frac{b-a}{6} [f(a) + 4f(\frac{a+b}{2}) + f(b)]$$

$$E_2(f) = -\frac{h^5}{90} f^{(4)}(\xi) = -\frac{1}{2880} (b-a)^5 f^{(4)}(\xi), \quad a < \xi < b$$

Para $n \geq 3$ se obtienen fórmulas similares. Para $n = 8$ y $n \geq 10$ algunos de los pesos α_j se vuelve negativos, lo cual no tiene ningún sentido. En la práctica no se utilizan las fórmulas de orden muy grande. Es más eficiente componer las fórmulas de orden bajo subdividiendo el intervalo $[a,b]$, el desarrollo de estas ideas se presentan a continuación.

1.4.2. Regla Trapezoidal compuesta:

La composición de la fórmula trapezoidal se puede realizar de la siguiente manera:

Sea $x_j = a + jh$, $j = 0, \dots, n$, donde $n \in \mathbf{N}$, $h = (b-a)/n$, y $f \in C^2[a, b]$. Aplicando la regla trapezoidal en los subintervalos $[x_{j-1}, x_j]$, $j = 1, \dots, n$ se obtiene

$$\int_a^b f(x)dx = \sum_{j=1}^n \int_{x_{j-1}}^{x_j} f(x)dx$$

$$= \sum_{j=1}^n \left\{ \frac{h}{2} (f(x_{j-1}) + f(x_j)) - \frac{h^3}{12} f''(\xi_j) \right\}, \quad \xi_j \in]x_{j-1}, x_j[$$

$$= T_n(f) - \sum_{j=1}^n \frac{h^3}{12} f''(\xi_j)$$

donde

$$T_n(f) := h \left[\frac{1}{2} f(x_0) + \sum_{j=1}^{n-1} f(x_j) + \frac{1}{2} f(x_n) \right],$$

es conocido como la fórmula de la Regla Trapezoidal Compuesta.

Aplicando el teorema del valor intermedio se obtiene el error:

$$\int_a^b f(x)dx - T_n(f) = -\frac{h^2(b-a)}{12} f''(\xi), \quad h = \frac{b-a}{n}.$$

1.4.3. Regla Compuesta de Simpson:

De manera similar se obtiene la Regla Compuesta de Simpson. Sean $n = 2m$ (par),

$m \in \mathbf{N}$, $h = (b-a)/n$, $x_j = a + jh$, $j = 0, \dots, n$, $f \in C^4[a, b]$.

Aplicando la regla de Simpson en los subintervalos $[x_{j-1}, x_j]$, $j = 1, \dots, n$ se obtiene

$$\int_a^b f(x)dx = S_n[a, b] - \frac{h^4(b-a)}{180} \left(\frac{1}{m} \sum_{j=1}^m f^{(4)}(\xi_j) \right), \quad \xi_j \in]x_{j-1}, x_j[.$$

donde

$$S_n[a, b] = \frac{h}{3} [f(x_0) + 4 \sum_{j=1}^m f(x_{2j-1}) + 2 \sum_{j=1}^{m-1} f(x_{2j}) + f(x_n)],$$

es conocido como la regla compuesta de Simpson.

Aplicando el teorema del valor intermedio

se obtiene el error

$$\int_a^b f(x) dx - S_n[a, b] = -\frac{h^4(b-a)}{180} f^{(4)}(\xi)$$

Denotaremos con π_n al espacio de los polinomios con coeficientes reales de grado menor o igual que n y con π al espacio de todos los polinomios.

La regla trapezoidal compuesta y la regla compuesta de Simpson tienen pesos positivos. A partir de los errores de estos métodos, se tiene que estos métodos de cuadratura convergen para todas las funciones de $C^2[a, b]$ o de $C^4[a, b]$ respectivamente y en consecuencia también en Π . A partir de esto mediante resultados del análisis funcional se deduce la convergencia en $C[a, b]$, esto significa que para funciones f continuas se tiene convergencia.

1.4.4. Fórmulas de Cuadratura de Gauss

Las fórmulas de cuadratura de Gauss aproximan integrales de la siguiente forma

$$I(f) = \int_a^b w(x) f(x) dx \quad \text{donde } w:]a, b[\rightarrow \mathbb{R}$$

es una función no negativa, integrable y con un número finito de ceros en $]a, b[$. La función w es denominada función de peso.

Definición 1.6 Una fórmula de cuadratura

$$G_n = \sum_{j=0}^n A_j f(x_j) \quad \text{con } (n+1) \text{ puntos de partición}$$

$$I(f) := \int_a^b w(x) f(x) dx \quad \text{es denominada del tipo}$$

Gauss, si $G_n(P) = I(P)$ para todo polinomio P de grado menor o igual que $(2n+1)$:

El siguiente resultado describe la naturaleza de los puntos de partición y de los pesos. Este tipo de resultados pueden encontrarse en [9].

Teorema 1.7 Sea $n \in \mathbb{N}$, $w(x)$ una función de peso y $p_i \in \Pi_i$, $i = 0, \dots, n+1$ dados por el proceso de la ortonormalización de $\{1, \dots, x^{n+1}\}$ con el producto escalar

$$(f, g) := \int_a^b w(x) f(x) g(x) dx \quad \text{para todo } f, g \in C[a, b]$$

Entonces se cumplen

1. Los $(n+1)$ ceros de P_{n+1} son distintos y se encuentran en $]a, b[$.
2. Sean $x_0 < x_1 < \dots < x_n$ los ceros de P_{n+1} . Definiendo.

$$G_n(f) := \sum_{j=0}^n A_j f(x_j)$$

3. $A_n > 0$, para $j=0, 1, \dots, n$

Los puntos de partición y los pesos son únicos, esto es, la fórmula de cuadratura de Gauss es única $f \in C^{2n+2}[a, b]$

5. Si, el error de la integración por las fórmulas de cuadratura del tipo Gauss está dada por

$$R_{G_n}(f) = I(f) - G_n(f) = \frac{f^{(2n+2)}(\xi)}{(2n+2)!} \int_a^b q_{n+1}^2(x) w(x) dx$$

donde

$$a < \xi < b \text{ y } q_{n+1}(x) := \prod_{k=0}^n (x - x_k).$$

Los polinomios ortogonales p_i dependen de la función de peso $w(x)$.

En estas fórmulas se está considerando n puntos de partición. Como las fórmulas de cuadratura de Gauss

G_n tienen pesos positivos y es exacto en Π_{2n+1} para

$n+1$ puntos de partición, esto implica que, $\{G_n\}$ es convergente en Π . Por consiguiente la sucesión fG_n es también convergente en $C[a, b]$.

Estos métodos se aplican cuando la función integrando es conocida. En estos métodos se deben estudiar los errores y su aplicación no es directa. También se tienen métodos de cálculo casi automáticos tales como los Métodos de cuadratura adaptativa para el cálculo aproximado de integrales definidas.

3. Discusión

Se quiere estimar el valor de la integral $\int_a^b f(x) dx$, para una partición en general no equidistante del intervalo $[a, b]$: $a = x_0 < x_1 < \dots < x_n = b$ donde se conocen $f(x_j) = f_j$.

Los métodos clásicos de integración numérica no son aplicables a éste problema, sin embargo se podría utilizar la idea general de las fórmulas de Newton-Cotes y la interpolación mediante los splines cúbicos. Para las fórmulas de Newton-Cotes, según el Teorema (1.4), se tiene la existencia de una única fórmula de cuadratura de la forma

$$K_n(f) = \sum_{j=0}^n \alpha_j f(x_j), \quad \text{con } \alpha_j = \int_a^b L_j(x), \quad j = 0, \dots, n,$$

la cual es exacta para todos los polinomios de grado menor o igual a n . Para el error sólo se tiene

$$E_n(f) := I(f) - K_n(f) = \int_a^b f(x) dx - \sum_{j=0}^n \alpha_j f(x_j),$$

no se tiene una acotación explícita

Para el caso de una partición uniforme, de la fórmula de cuadratura de Newton Cotes general se obtienen fórmulas especiales para valores pequeños de n , para $n = 8$ y $n \geq 10$ algunos de los pesos α_j son negativos, lo

cual no tiene ningún sentido, no se puede asegurar la convergencia. En el método de cuadratura de Gauss, los puntos de partición x_j son las raíces de un polinomio ortonormal P_n . Lo que implica que ningún de estos métodos se pueden aplicar para una partición arbitraria del intervalo de integración. Sólo se podría rescatar el método de Newton-Cotes para una partición pequeña o dividiendo el intervalo de integración en subintervalos.

Se presenta el algoritmo de Newton Cotes:

función $Int = Newton_Cotes(x, f, n)$
Entrada: $\% f = (f_0, f_1, f_2, \dots, f_n), \quad x = (x_0, x_1, x_2, \dots, x_n)$ $\% con \ a = x_0 < x_1 < x_2 < \dots < x_n = b$ Salida: $Int = \sum_{j=0}^n \alpha_j f_j$
$Int := 0$
Para $j := 0, \dots, n$
$\alpha_j := \int_a^b L_j(x) dx$
$Int := Int + \alpha_j f_j$

En este algoritmo se requiere calcular la integral de los polinomios

$$\alpha_j = \frac{1}{M_j} \int_a^b \prod_{\substack{k=0 \\ k \neq j}}^n (x - x_k) dx \text{ donde } M_j = \prod_{\substack{k=0 \\ k \neq j}}^n (x_j - x_k),$$

se tiene que expandir, integrar y al evaluar es recomendable aplicar el método de Horner.

2.0.5. Método de Newton-Cotes compuesto para la Integración Numérica

Es conocido que para el caso de particiones uniformes del intervalo de integración, es más eficiente componer las fórmulas de cuadratura de Newton-Cotes de orden bajo subdividiendo el intervalo $[a; b]$, así se han obtenido, por ejemplo, la regla trapezoidal compuesta y la regla compuesta de Simpson. Siguiendo esta idea, consideramos

$\{r_0, r_1, \dots, r_p\} \subset \{0, 1, 2, \dots, n\}$ tal que $r_0 = 0 < r_1 < \dots < r_p = n$. Luego, en cada subintervalo $[x_{r_i}, x_{r_{i+1}}], i = 0, 1, 2, \dots, p-1$ se aplica la fórmula de cuadratura de Newton-Cotes y se obtiene

$$K(f) = \sum_{i=0}^{p-1} \sum_{j=r_i}^{r_{i+1}} \alpha_j^{(i)} f(x_j),$$

Con

$$\alpha_j^{(i)} = \int_a^b L_j^{(i)}(x), \quad L_j^{(i)}(x) = \prod_{\substack{k=r_i \\ k \neq j}}^{r_{i+1}} \frac{x - x_k}{x_j - x_k}, \quad j = r_i, \dots, r_{i+1},$$

la fórmula obtenida es exacta para cualquier polinomios de grado menor o igual a

$$\min \{r_{i+1} - r_i : i = 0, 1, \dots, p-1\}.$$

A partir de esto se propone el siguiente algoritmo:

función $Int = Newton_Cotes_Compuesto(x, f, r, p, n)$
Entrada: $\% f = (f_0, f_1, f_2, \dots, f_n), \quad x = (x_0, x_1, x_2, \dots, x_n)$ $\% con \ a = x_0 < x_1 < x_2 < \dots < x_n = b$ $\% r = (r_0, r_1, \dots, r_p)$ Salida: $Int = \sum_{j=0}^n \alpha_j f_j$
$Int := 0$
Para $j := 0, \dots, p-1$
$Int = Int + Newton_Cotes((x_{r_j}, \dots, x_{r_{j+1}}), (f_{r_j}, \dots, f_{r_{j+1}}), r_{j+1} - r_j)$

La implementación del método de Newton-Cotes compuesto se ha realizado aplicando el método de Newton-Cotes cada cuatro puntos interpolantes, para un número de nodos $n + 1$ donde n es múltiplo de 3.

2.0.6. Método de Splines Cúbicos

Para la Integración Numérica se va a aplicar los splines cúbicos en el cálculo de integrales, para el caso de particiones del intervalo de integración igualmente espaciados se pueden encontrar por ejemplo en [4]. En esta sección se va a aplicar los splines cúbicos para la integración en el caso donde el intervalo de partición no es igualmente espaciado y se va a realizar un análisis de convergencia.

Proposición 2.1 Dados los puntos interpolantes $a = x_0 < x_1 < \dots < x_n = b$ en el intervalo $[a, b]$ con los valores correspondientes f_0, f_1, \dots, f_n , para la interpolación por el spline cúbico natural S_{Δ}^3 se cumple.

$$\int_a^b S_{\Delta}^3(x) dx = \sum_{j=0}^{n-1} \frac{h_{j+1}}{2} \left[f_j + f_{j+1} - \frac{1}{6} h_{j+1}^2 (c_j + c_{j+1}) \right].$$

donde

$$h_{j+1} = x_{j+1} - x_j, \quad j = 0, 1, \dots, n-1.$$

Prueba. A partir de (4) y (5) se obtiene

$$\begin{aligned}
\int_a^b S_{\Delta}^3(x) dx &= \sum_{j=0}^{n-1} \int_{x_j}^{x_{j+1}} S_{\Delta}^3(x) dx \\
&= \sum_{j=0}^{n-1} \int_{x_j}^{x_{j+1}} \left(c_j \frac{(x_{j+1}-x)^3}{3h_{j+1}} + c_{j+1} \frac{(x-x_j)^3}{3h_{j+1}} + D_j(x-x_j) + E_j \right) dx \\
&= \sum_{j=0}^{n-1} \left[-c_j \frac{(x_{j+1}-x)^4}{12h_{j+1}} + c_{j+1} \frac{(x-x_j)^4}{12h_{j+1}} + \frac{D_j}{2}(x-x_j)^2 + E_j x \right]_{x=x_j}^{x_{j+1}} \\
&= \sum_{j=0}^{n-1} h_{j+1} \left[c_{j+1} \frac{h_{j+1}^2}{12} + c_j \frac{h_{j+1}^2}{12} + \frac{D_j}{2} h_{j+1} + E_j \right] \\
&= \sum_{j=0}^{n-1} h_{j+1} \left[\frac{c_{j+1}h_{j+1}^2}{12} + \frac{c_j h_{j+1}^2}{12} + \frac{1}{2} \left(\frac{f_{j+1}-f_j}{h_{j+1}} - \frac{h_{j+1}}{3} (c_{j+1}-c_j) \right) h_{j+1} + f_j - \frac{c_j h_{j+1}^2}{3} \right] \\
&= \sum_{j=0}^{n-1} h_{j+1} \left(\frac{1}{2} (f_j + f_{j+1}) - \frac{1}{12} h_{j+1}^2 (c_{j+1} + c_j) \right)
\end{aligned}$$

2.0.7. Análisis de Convergencia

Para analizar la convergencia veamos el siguiente resultado que se puede encontrar, por ejemplo en la página. 105 de [9].

Teorema 2.2

Sea $f \in C^4[a, b]$ y $|f^{(4)}(x)| \leq L$ para $x \in [a, b]$. Sea $\Delta = \{a = x_0 < \dots < x_n = b\}$ una partición del intervalo $[a, b]$, y K una constante tal que

$$\frac{\|\Delta\|_{\infty}}{|x_{j+1} - x_j|} \leq K \text{ para } j = 0, \dots, n-1.$$

Si S_{Δ}^3 es el spline que interpola los valores de la función f en los nodos $x_0, x_1, \dots, x_n \in \Delta$ y satisface $(S_{\Delta}^3)'(x) = f'(x)$ para $x = a; b$; entonces existen constantes $C_k \leq 2$, que no dependen de la partición Δ , tal que para $x \in [a, b]$ se cumple

$$\left| f^{(i)}(x) - (S_{\Delta}^3)^{(i)}(x) \right| \leq C_k L K \|\Delta\|_{\infty}^{4-i}, i = 0, 1, 2, 3.$$

donde $\|\Delta\|_{\infty} = \max_j |x_{j+1} - x_j|$

En la prueba de este Teorema se tiene $C_3 = 2$, $C_2 = C_1 = 7/4$ y $C_0 = 7/8$ Manteniendo las hipótesis del Teorema precedente con $i = 0$, se propone el siguiente resultado.

Proposición 2.3 Se cumple

$$\left| \int_a^x f(t) dt - \int_a^x S_{\Delta}^3(t) dt \right| \leq \frac{7}{8} L K \|\Delta\|_{\infty}^4 (b-a), x \in [a, b].$$

Prueba.

$$\begin{aligned}
\left| \int_a^x f(t) dt - \int_a^x S_{\Delta}^3(t) dt \right| &\leq \int_a^x |f(t) - S_{\Delta}^3(t)| dt \\
&\leq \frac{7}{8} L K \|\Delta\|_{\infty}^4 (b-a)
\end{aligned}$$

Según el Teorema (2.2), para las sucesiones

$$\Delta_k = \{a = x_0^{(k)} < \dots < x_{n_k}^{(k)} = b\}, k = 0, 1, \dots$$

de particiones del intervalo $[a; b]$ con

$$\|\Delta_k\|_{\infty} \rightarrow 0 \text{ y } \sup_{m,j} \frac{\|\Delta_k\|_{\infty}}{x_{j+1}^{(k)} - x_j^{(k)}} \leq K < +\infty,$$

las correspondientes funciones splines $S_{\Delta_k}^3$ y sus correspondientes tres primeras derivadas convergen uniformemente sobre $[a, b]$ a f y a sus correspondientes derivadas. Por la Proposición (2.3) se obtiene que la función $S_k(x) = \int_a^x S_{\Delta_k}^3(t) dt$ converge uniformemente a $F(x) = \int_a^x f(t) dt$ sobre $[a, b]$.

En particular para una partición uniforme $x_j = a + j \|\Delta\|_{\infty}$, $\|\Delta\|_{\infty} = \frac{b-a}{n}$, la cota

$K = 1$ satisface la hipótesis del Teorema (2.2) y se obtiene el siguiente Corolario.

Corolario 2.4 Se cumple

$$\left| \int_a^b f(x) dx - \int_a^b S_{\Delta}^3(x) dx \right| \leq \frac{7}{8} \frac{L}{n^4} (b-a)^5$$

Para la regla compuesta de Simpson se tiene el error

$$\begin{aligned}
\left| \int_a^b f(x) dx - S_n[a, b] \right| &= \frac{(b-a)}{180} \left(\frac{b-a}{n} \right)^4 |f^{(4)}(\xi)| \\
&\leq \frac{1}{180} \frac{L}{n^4} (b-a)^5
\end{aligned}$$

Esto significa que para el caso de partición uniforme la cota de error del método de integración aplicando spline cúbicos tiene el mismo orden que la regla compuesta de Simpson y supera a la regla trapezoidal compuesta.

4. Resultados

A partir de la Proposición (2.1) y modificando el algoritmo de la función SplineCúbicoN() desarrollado en la primera sección, se propone el siguiente algoritmo para el cálculo aproximado de la integral

$$\int_a^b f(x) dx, \quad f: [a, b] \rightarrow \mathbf{R},$$

aplicando la interpolación por splines cúbicos natural y donde se conocen los valores $f_0 := f(x_0), f_1 := f(x_1), \dots, f_n := f(x_n)$, de la función f en una partición

$\Delta = \{x_0 = a < x_1 < \dots < x_n = b\}$ del intervalo [a,b]. Específicamente el algoritmo calcula la integral

$\int_a^b S_{\Delta}^3(x) dx$, donde $S_{\Delta}^3(x)$ es el spline cúbico natural que interpola los puntos $\{(x_0, f_0), (x_1, f_1), \dots, (x_n, f_n)\}$

función $Int = Int_Spline_cúbico_N(x, f, n)$
% Entrada: $x := (x_0, x_1, \dots, x_n), f := (f_0, f_1, \dots, f_n)$, con $x_j \neq x_k$ para $j \neq k$
% Calcula la siguiente integral aplicando Splines cúbicos natural,
% Salida : $Int := \int_a^b S_{\Delta}^3(x) dx$
$a_0 := f_0$
Para $j := 1, \dots, n$
$h_j := x_j - x_{j-1}$
$a_j := f_j$
Para $j := 1, 2, \dots, n-1$
$g_j := \frac{3}{h_j + h_{j+1}} \left(\frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j-1}}{h_j} \right)$
$v_j = \frac{h_j}{h_j + h_{j+1}}$
$w_j := 1 - v_j$
$s_j := 2$
$s_n := 2$
$c_0 := 0$
$c_n := 0$
$(c_1, c_2, \dots, c_{n-1}) := tridiagonal(v, s, v, g, n-1)$
$Int := 0$
Para $j := 0, 1, \dots, n-1$
$Int := Int + \frac{h_{j+1}}{2} \left[f_j + f_{j+1} - \frac{1}{6} h_{j+1}^2 (c_j + c_{j+1}) \right]$

También modificando el algoritmo de la función SplineCúbicoL () desarrollado en la primera sección

para los splines cúbico libre, se propone el siguiente algoritmo para el cálculo aproximado de la integral.

función $Int = Int_Spline_cúbico_L(x, f, n)$
% Entrada: $x := (x_0, x_1, \dots, x_n), f := (f_0, f_1, \dots, f_n)$, con $x_j \neq x_k$ para $j \neq k$
% Calcula la siguiente integral aplicando Splines cúbicos libre
% Salida : $Int := \int_a^b S_{\Delta}^3(x) dx$
$a_0 := f_0$
Para $j := 1, \dots, n$
$h_j := x_j - x_{j-1}$
$a_j := f_j$
Para $j := 1, 2, \dots, n-1$
$g_j := \frac{3}{h_j + h_{j+1}} \left(\frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j-1}}{h_j} \right)$
$v_j := \frac{h_j}{h_j + h_{j+1}}$
$w_j := 1 - v_j$
$s_j := 2$
$s_1 := 2 + \frac{h_1}{h_2}$
$s_{n-1} := 2 + \frac{h_n}{h_{n-1}}$
$v_{n-1} := 1 - \frac{h_n}{h_{n-1}}$
$w_1 := 1 - \frac{h_1}{h_2}$
$(c_1, c_2, \dots, c_{n-1}) := tridiagonal(v, s, v, g, n-1)$
$c_0 := \left(\frac{h_1 + h_2}{h_2} \right) c_1 - \frac{h_1}{h_2} c_2$
$c_n := \left(\frac{h_{n-1} + h_n}{h_{n-1}} \right) c_{n-1} - \frac{h_n}{h_{n-1}} c_{n-2}$
$Int := 0$
Para $j := 0, 1, \dots, n-1$
$Int := Int + \frac{h_{j+1}}{2} \left[f_j + f_{j+1} - \frac{1}{6} h_{j+1}^2 (c_j + c_{j+1}) \right]$

En los siguientes ejemplos en cada caso se va a considerar los puntos de partición, definidos como sigue

$$X_0 := a, x_n := b,$$

$$X_k := a + \Delta x \cdot (k-1 + \text{rand}()); k = 1; 2; \dots; n-1$$

donde $\Delta x := (b-a)/n$ y $\text{rand}()$ es un número random entre 0 y 1. Se cumple

$$a + \Delta x \cdot (k-1) \leq x_k \leq a + \Delta x \cdot k.$$

Denotamos con $\Delta_{\min} := \min_j \{x_{j+1} - x_j\}$ a la menor distancia entre los nodos.

Ejemplo 1: Se quiere calcular la siguiente integral

$$\int_1^4 f(x) dx \quad \text{donde} \quad f(x) = \frac{100}{x^2} \text{Sen}\left(\frac{10}{x}\right)$$

Para los ejemplos en el método de Newton-Cotes compuesto se considera que este método aplica el método de Newton-Cotes para cada cuatro puntos interpolantes.

En las tablas que se presentan a continuación contienen resultados de 10 corridas

en cada caso de los programas que han sido implementados con el software MATLAB.

a) Considerando 25 nodos obtenidos al azar, en la Tabla 1.1 a) se muestra la distancia mínima entre los puntos de la partición y la norma de la partición, esto indica que los puntos de partición no son equidistantes. En la tabla 1.2 a) se observa que los errores relativos son muy grandes para ambos métodos, esto significa que las aproximaciones son pésimas.

Tabla 1.1 a) Distancia mínima y Norma de la partición del intervalo [a,b].

k	Δ_{\min}	$\ \Delta\ _{\infty}$
1	0.009542863278271	0.210222337695950
2	0.021097524047095	0.197049973522668
3	0.031423544164884	0.240182495111788
4	0.027722110744429	0.213045454985825
5	0.014855212528378	0.225014944748059
6	0.012266273434978	0.231434090935039
7	0.023699809831106	0.245732352998048
8	0.032850703886944	0.249152827462857
9	0.000463607558235	0.210723909941148
10	0.004336575500066	0.225814687255378

Tabla 1.2 a) Valores aproximados de la integral y sus respectivos errores relativos obtenidos aplicando integración numérica basada en los splines cúbico y el Método de Newton-Cotes.

k	Spline Natural	Error Relativo	Newton-Cotes	Error Relativo
1	0.188245256122692	0.503676214679767	-0.181737928792820	1.479166692497800
2	0.491124051522013	0.294888133352703	1.073550701901175	1.830502925149431
3	0.469849542699073	0.238796176682370	0.553156370704532	0.458441341029788
4	0.373980224555942	0.013971005115063	0.289654467544173	0.236302657886152
5	0.208957461506873	0.449066816332397	-0.035346607007724	1.093194177370749
6	0.459041410315275	0.210299664804945	0.110037808898223	0.709876450723865
7	0.316607208450364	0.165239584814088	0.096280125325162	0.746149691967029
8	0.349451530973071	0.078642882105554	0.825117373065037	1.175488436564958
9	0.582008936353433	0.534513455111271	1.085773323352421	1.862728851423318
10	0.574997826506071	0.516028099090024	1.711623067879446	3.512832129685475

b) Considerando 241 nodos obtenidos al azar. En la tabla 1 b) se observa que los errores relativos son pequeños y no hay mucha diferencia entre ellos, esto

significa que las aproximaciones han mejorado al incrementar el número de nodos.

Tabla 1 b) Valores aproximados de la integral y sus respectivos errores relativos obtenidos aplicando integración numérica basada en los splines cúbicos y el Método de Newton-Cotes.

k	Spline Natural	Error Relativo	Newton-Cotes	Error Relativo
1	0.378494492266748	0.002068774565807	0.379235462096649	0.000115147906842
2	0.378014579842650	0.003334102340098	0.379336239654278	0.000150560243832
3	0.378874064114266	0.001068002806446	0.379331877377538	0.000139058749725
4	0.378656797628285	0.001640843402624	0.379333030899906	0.000142100104390
5	0.378553459150150	0.001913303626558	0.379344092409006	0.000171264664395
6	0.378242536928881	0.002733075114979	0.379308170533180	0.000076553744434
7	0.379242657977694	0.000096175386672	0.379345507748937	0.000174996322162
8	0.377659697265640	0.004269778848466	0.379342097467274	0.000166004839780
9	0.377733676551970	0.004074726499296	0.379339528108680	0.000159230518825
10	0.379163771488076	0.000304165972699	0.379324796224116	0.000120388718175

c) Considerando 2401 nodos obtenidos al azar. En la tabla 1 c) se observa que los errores relativos son

pequeños, al aplicar integración basada en los splines cúbicos natural se observan mejoras y en el Método

de Newton-Cotes empeora respecto al caso anterior.

Tabla 1 c) Valores aproximados de la integral y sus respectivos errores relativos obtenidos aplicando integración numérica basada en los splines cúbicos y el Método de Newton-Cotes.

k	Spline Natural	Error Relativo	Newton-Cotes	Error Relativo
1	0.379270197550200	0.000023565084804	0.379951294260364	0.001772201269794
2	0.379275845280602	0.000008674388541	0.378586208107804	0.001826958361008
3	0.379272361653333	0.000017859252524	0.381643771709950	0.006234554434223
4	0.379272567708796	0.000017315970693	0.379472219631287	0.000509082409580
5	0.379273095494472	0.000015924421234	0.378672691452895	0.001598938053421
6	0.379271016796158	0.000021405076825	0.378698049656981	0.001532079105153
7	0.379276391167087	0.000007235114841	0.377194136506647	0.005497267301344
8	0.379273055934272	0.000016028724889	0.379836169144189	0.001468664624981
9	0.379272816187662	0.000016660836143	0.378915714311319	0.000958188706018
10	0.379275229934469	0.000010296798203	0.379888340584568	0.001606218831168

d) Considerando 24001 nodos obtenidos al azar. En la tabla 1 d) se observa que los errores relativos son muy pequeños para el caso donde se aplican la integración basada en los splines cúbicos natural y en

el caso de la aplicación del Método de Newton-Cotes los errores relativos son muy grandes, a pesar de incrementarse el número de nodos.

Tabla 1 d) Valores aproximados de la integral y sus respectivos errores relativos obtenidos aplicando integración numérica basada en los splines cúbicos y el Método de Newton-Cotes.

k	Spline Natural	Error Relativo	Newton-Cotes	Error Relativo
1	0.379279069888087	0.000000172451090	-0.992771049213576	3.617520862150554
2	0.379279079810183	0.000000146290686	-7.136401545263160	19.815697677935809
3	0.379279085207876	0.000000132059233	7.642662297178062	19.150494941489175
4	0.379279078688566	0.000000149247919	2.986792176396594	6.874918228950341
5	0.379279076521332	0.000000154962006	-1.912874524625895	6.043447810903524
6	0.379279081088123	0.000000142921294	4.261292058517633	10.235239858900044
7	0.379279073227524	0.000000163646394	6.242439400007082	15.458694452434569
8	0.379279075462995	0.000000157752397	2.071524837018252	4.461742142514680
9	0.379279086044501	0.000000129853403	-0.511595286037824	2.348862192589784
10	0.379279082014769	0.000000140478116	4.504179478574307	10.875632112134957

En los resultados de las aproximaciones de la integral, se aprecia una mejor aproximación aplicando el método de spline cúbicos natural, esto era de esperarse debido al análisis de convergencia de éste método. Respecto al uso del método de Newton-Cotes compuesto, se pueden obtener resultados muy pésimos a pesar de tener un número grande de nodos, esto no solamente se debe al hecho que el polinomio interpolante en general no es una buena aproximación sino también a la acumulación de los errores de redondeo en la evaluación de fórmulas que son complicadas.

Ejemplo 2: Se quiere calcular la siguiente integral

$$\int_1^4 f(x) dx \quad \text{donde consideramos la función de prueba}$$

definida por $f(x) = e^{x-1} + e^{4-x}$. Se cumple $f''(1), f''(4)$ no son cercanos a cero. No se cumplen las dos restricciones adicionales de los splines cúbicos natural:

$$(S_{\Delta}^3)''(X_0) = 0 = (S_{\Delta}^3)''(X_n)$$

a) Considerando 25 nodos obtenidos al azar, en la Tabla 2 a) se muestran las aproximaciones de las integrales obtenidas mediante los splines cúbicos natural y splines cúbicos libre, en éste caso se observa que en general los errores relativos aplicando el segundo método son menores. Esto puede deberse al hecho que los splines cúbicos libre no están sujetos a las dos restricciones adicionales de los splines cúbicos natural.

Tabla 2 a) Valores aproximados de la integral y sus respectivos errores relativos obtenidos aplicando integración numérica basado en los splines cúbicos natural y splines cúbicos libre.

k	Spline Natural	Error Relativo	Spline cúbicos Libre	Error Relativo
1	38.177319730794622	0.000163628732176	38.173658737140151	0.000067718575988
2	38.177683583383498	0.000173160887083	38.171704158561916	0.000016512823011
3	38.172498001440061	0.000037309798264	38.171026569750836	0.000001238545835
4	38.175441693004643	0.000114428183154	38.172588590564708	0.000039683038404
5	38.176625524931573	0.000145442032324	38.174678901643738	0.000094444690838
6	38.179356866221376	0.000216997296942	38.172849021466121	0.000046505767637
7	38.177699986031897	0.000173590601177	38.172285577853991	0.000031744757392
8	38.173473492859380	0.000062865574432	38.170983224176325	0.000002374106618
9	38.178066411736985	0.000183190166192	38.173112208601395	0.000053400704268
10	38.176690681424091	0.000147148992228	38.173067563523915	0.000052231099303

b) Considerando 241 nodos obtenidos al azar, en la Tabla 2 b) se observa que los errores relativos son pequeños en ambos métodos no se distingue mucha diferencia entre ellos, las aproximaciones mejoran al incrementar el número de nodos. Se tiene que tener en cuenta que la función de prueba es un polinomio y

tiene buenas características y además al aumentar el número de nodos los resultados mejoran, esto era de esperarse debido al análisis de convergencia y por la deducción de fórmulas simples para la integración numérica, con lo cual se disminuye el problema de la acumulación de errores por redondeo.

Tabla 1.2 b) Valores aproximados de la integral y sus respectivos errores relativos obtenidos aplicando integración numérica basado en los splines cúbicos natural y splines cúbicos libre.

k	Spline Natural	Error Relativo	Spline cúbicos Libre	Error Relativo
1	38.171110177977617	0.000000951809803	38.171094571481269	0.000000542953180
2	38.171097423479509	0.000000617669397	38.171096789417568	0.000000601058338
3	38.171099340214766	0.000000667883736	38.171095055917029	0.000000555644354
4	38.171102347765029	0.000000746675082	38.171095244919435	0.000000560595811
5	38.171110856281295	0.000000969579900	38.171107582620429	0.000000883817029
6	38.171101454618501	0.000000723276564	38.171097040617894	0.000000607639246
7	38.171095040785353	0.000000555247937	38.171095940271456	0.000000578812537
8	38.171107252399970	0.000000875165963	38.171101926158485	0.000000735629897
9	38.171091491482002	0.000000462263826	38.171083143383122	0.000000243561599
10	38.171107686365730	0.000000886534933	38.171099254194750	0.000000665630197

4. Conclusiones

Se ha propuesto tres métodos para el cálculo aproximado de una integral definida para el caso donde se conocen los valores de la función en una partición no uniforme del intervalo de integración. Para cada método se ha propuesto su respectivo algoritmo lo que facilita su implementación.

Aplicando en la integración los de splines cúbicos natural y splines cúbicos libre se obtienen buenas aproximaciones de integrales para particiones no uniformes del intervalo de integración. Esto se sustenta en el análisis de convergencia que se ha deducido para éste método y de otro lado por las formulaciones simplificadas que se han obtenido para el cálculo de la integral, con los cuales se disminuyen el problema acumulación de los errores de redondeo.

Al aplicar la propuesta del método de Newton-Cotes compuesto, se observa en algunos casos se obtienen resultados muy pésimos a pesar de contar con un número muy grande de nodos. Posiblemente, esto se deba a la acumulación de los errores de redondeo al tener que evaluar formulas complicadas en cada paso. Es importante tener en cuenta la naturaleza de las funciones integrandos. Al trabajar con una función de prueba que no satisface las dos hipótesis adicionales para los splines cúbicos natural se aprecia una ligera mejora en los resultados aplicando los splines cúbicos libre que no están sujetas a estas restricciones. Para el caso de funciones periódicas es recomendable deducir una formulación de integración similar donde los splines cúbicos consideran la naturaleza de ser periódica.

5. Referencias bibliográficas

- Böhn, W.; Faring, G.; Kahmann, J. (1984), A survey of curve and surface methods in Cagd. Elsevier Science Publishers B.V. (North-Holland).
- Boehm, W., Prautzsch, H. (1993), Numerical Methods, Vieweg.
- Burden, R. L. (1998), Análisis numérico. International Thomson Editores, International Thomson, Sexta Edición.
- Gerald, C. F.; Wheatley, P.O. (2000), Análisis numérico con aplicaciones. Pearson Educación. Sexta edición, México.
- Kincaid, D., Cheney, W. (1994), Análisis Numérico, Addison-Wesley.
- Nürnberg, G. (1989), Approximation by Spline Functions, Springer-Verlag.
- Lyche, T.; Schumaker, L.L. (1973), Computation of smoothing and interpolating natural splines via local bases, SIAM J. Number. Anal. Vol. 10, No 6, December.
- Smith, A. W. (1988), Análisis Numérico, Prentice Hall.
- Stoer, J.; Bulirsch, R. (1993), Introduction to Numerical Analysis, Springer-Verlag.
- Park, J. H. (1999), Chebyshev Approximation of Discrete Polynomials and Splines, Thesis of Doctor of Philosophy in Bradley Department of Electrical Engineering.
- Penny, J. (1995), Numerical methods using MATLAB, Ellis Horwood Limited.