Laboratorium 3

Interpolacja

Bartłomiej Szubiak

19.03.2024

Zadania

Zad 1

Dane są trzy węzły interpolacji (-1 , 2.4), (1 , 1.8), (2 ,4.5), proszę obliczyć wielomian interpolacyjny 2-go stopnia, wykorzystując:

a) jednomiany

b) wielomiany Lagrange'a

c) wielomiany wg wzoru Newton'a

Pokazać że trzy reprezentacje dają ten sam wielomian

niech:

$(x_o, y_o) = (-1\ , 2.4)$

$(x_1, y_1) = (1\ , 1.8)$

$(x_2, y_2) = (2\ , 4.5)$

(a) Jednomiany:

Wielomian będzie postaci: $W(x) = a_0 + a_1 x + a_2 x^2$

Szukane współczynniki a_0 , a_1 , a_2 znajdziemy rozwiązując poniższy układ równań:

$$\begin{matrix} y_o = a_0 + a_1 x_o + a_2 {x_o}^2 \\ y_1 = a_0 + a_1 x_1 + a_2 {x_1}^2 \\ y_2 = a_0 + a_1 x_2 + a_2 {x_2}^2 \end{matrix} \quad => \quad \begin{matrix} 2.4 = a_0 + a_1(-1) + a_2(-1)^2 \\ 1.8 = a_0 + a_1(1) + a_2(1)^2 \\ 4.5 = a_0 + a_1(2) + a_2(2)^2 \end{matrix}$$

Otrzymujemy:

$$a_0 = 1.1\ , a_1 = -0.3\ , a_2 = 1$$

Więc wielomian jest postaci: $W(x) = 1.1 - 0.3x + x^2$

***(b)* wielomiany Lagrange'a**

Otrzymujemy wtedy wielomian:

$$\mathrm{W(x)} = \frac{(\mathrm{x}-\mathrm{x}_1)(\mathrm{x}-\mathrm{x}_2)}{(\mathrm{x}_0-\mathrm{x}_1)(\mathrm{x}_0-\mathrm{x}_2)} \cdot \mathrm{y}_0 + \frac{(\mathrm{x}-\mathrm{x}_0)(\mathrm{x}-\mathrm{x}_2)}{(\mathrm{x}_1-\mathrm{x}_0)(\mathrm{x}_1-\mathrm{x}_2)} \cdot \mathrm{y}_1 + \frac{(\mathrm{x}-\mathrm{x}_0)(\mathrm{x}-\mathrm{x}_1)}{(\mathrm{x}_2-\mathrm{x}_0)(\mathrm{x}_2-\mathrm{x}_1)} \cdot \mathrm{y}_2 =$$

$$= \frac{(x-1)(x-2)}{(-1-1)(-1-2)} \cdot 2.4 + \frac{\big(x-(-1)\big)(x-2)}{\big(1-(-1)\big)(1-2)} \cdot 1.8 + \frac{\big(x-(-1)\big)(x-1)}{\big(2-(-1)\big)(2-1)} \cdot 4.5 =$$

$$= x^2 - 0.3x + 1.1$$

***(c)* wielomiany wg wzoru Newton'a**

$$W(x) = f[x_0] + f[x_0,x_1](x-x_0) + f[x_0,x_1,x_2](x-x_0)(x-x_1)$$

gdzie:

$\mathrm{f}[\mathrm{x}_0] = \mathrm{y}_0 = 2.4$, $\quad \mathrm{f}[\mathrm{x}_1] = \mathrm{y}_1 = 1.8$, $\quad \mathrm{f}[\mathrm{x}_2] = \mathrm{y}_2 = 4.5$,

$$f[x_0,x_1] = \frac{f[x_1]-f[x_0]}{x_1-x_0} = \frac{1.8-2.4}{1-(-1)} = -0.3$$

$$f[x_1,x_2] = \frac{f[x_2]-f[x_1]}{x_2-x_1} = \frac{4.5-1.8}{2-1} = 2.7$$

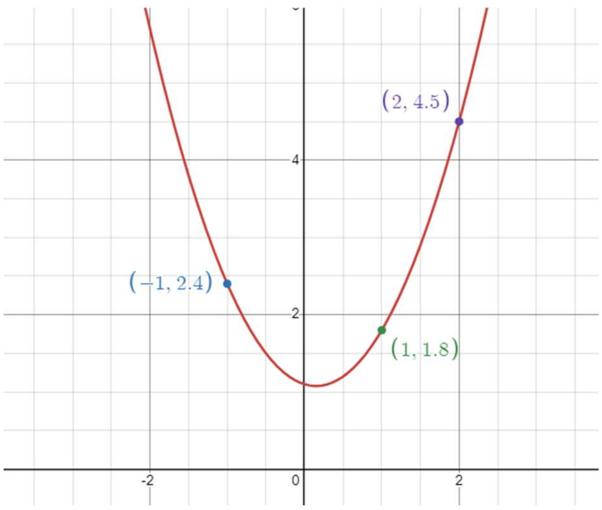
$$f[x_0,x_1,x_2] = \frac{f[x_1,x_2]-f[x_0,x_1]}{x_2-x_0} = \frac{2.7-(-0.3)}{2-(-1)} = 1$$

Wstawiając wartości:

$$W(x) = 2.4 - 0.3(x-(-1)) + 1(x-(-1))(x-1) = x^2 - 0.3x + 1.1$$

Wnioski:

Wszystkie trzy metody dają to samo rozwiązanie



Rysunek 1: Wykres funkcji $W(x) = 1.1 - 0.3x + x^2$

Zad 2

Wyrazić następujący wielomian metodą Hornera: p(t) = 3t^3 – 7t^2 + 5t – 4

Przekształcając:

$$p(t) = 3t^3 + 7t^2 + 5t - 4 = t(3t^2 + 7t + 5) - 4 = t(t(3t + 7) + 5) - 4$$

Zad 3

Ile mnożeń trzeba wykonać do ewaluacji wielomianu p(t) stopnia n-1 w danym punkcie t jeżeli wybieramy jako reprezentacje:

a) jednomiany

b) wielomiany Lagrange'a

c) wielomiany Newtona

- **(a)** Ewaluując mamy **n-1** mnożeń
- **(b)** Aby obliczyć L_k musimy wykonać *n*−1 mnożeń, gdzie *n* jest liczbą wyrazów, ponieważ mamy *n* wyrazów, aby obliczyć wszystkie wyrazy *L*, musimy wykonać *n*·(*n*−1) mnożeń. Dodatkowo każdy z wyrazów musi być pomnożony przez rzędną, więc w sumie otrzymujemy: $n \cdot (n-1) + n = \boldsymbol{n^2}$ mnożeń.
- **(c)** Wielomian stopnia *n*−1 w postaci Newtona jest sumą *n*+1 wielomianów o stopniach odpowiednio 0,1,…,*n*. Zgodnie z punktem **(a)**, wiemy, że na każdy składnik wielomianu stopnia k potrzeba k-1 mnożeń. Dodatkowo, każdy z tych *n* wielomianów jest mnożony przez współczynnik a_i, co daje kolejne *n* mnożeń. Ostatecznie ilość mnożeń wynosi:

$$\sum_{k=1}^{n+1} (k-1) = \frac{n(n+1)}{2}$$

Zadania domowe

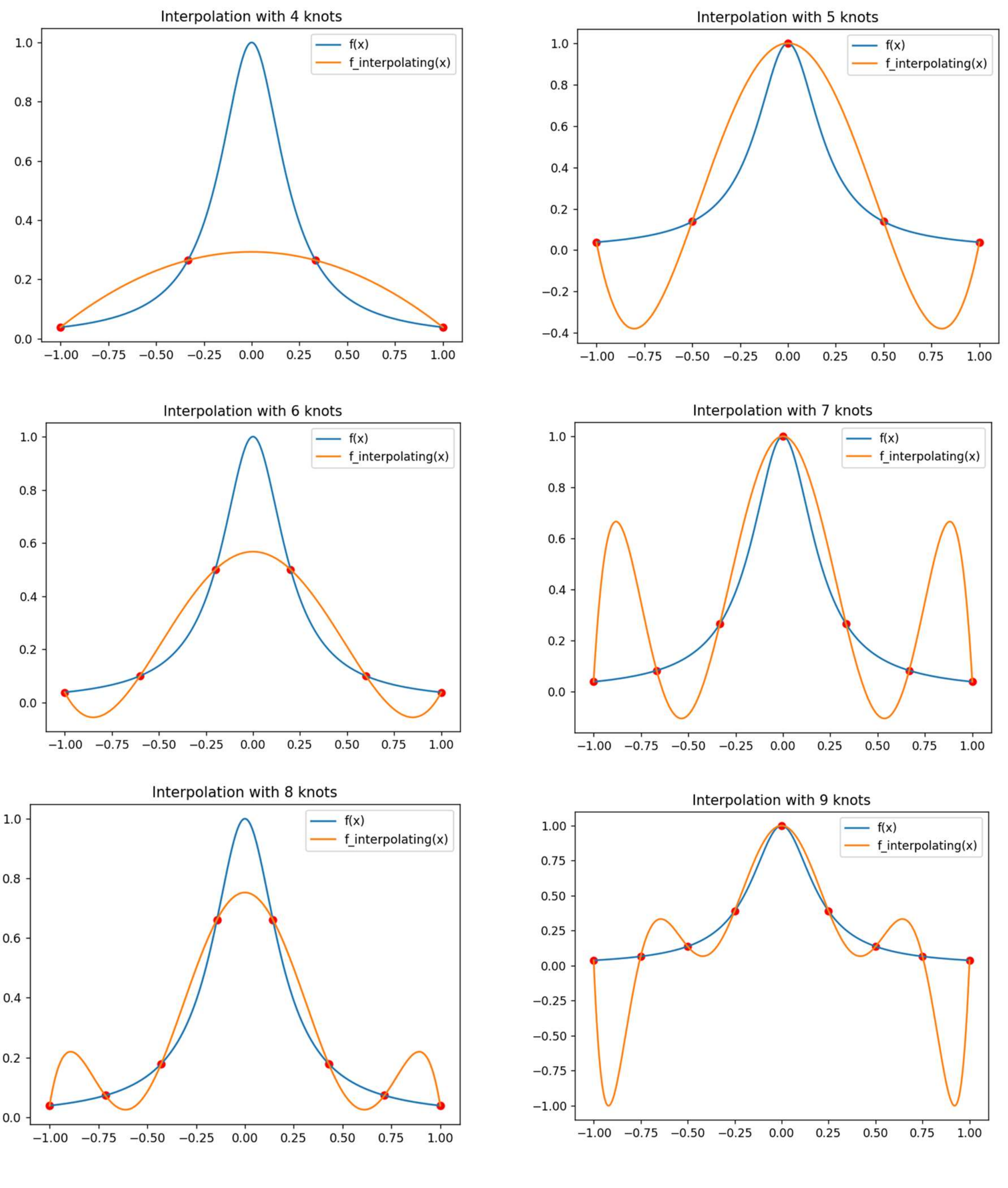
Zad 1

Znaleźć kompromis między granicą błędu a zachowaniem wielomianu interpolacyjnego dla funkcji Rungego f(t) = 1/(1 + 25t^2), dla równoodległych węzłów na przedziale [-1,1]:

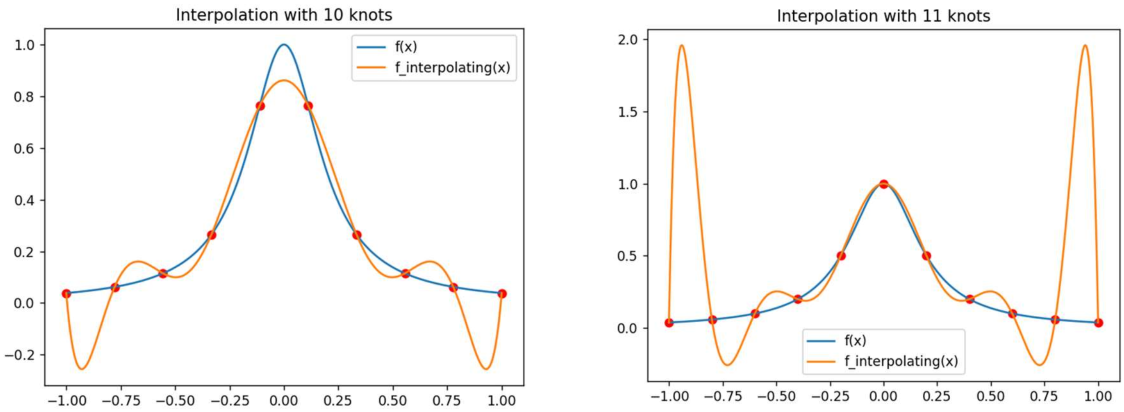
Z pomocą poniższego programu w Python generuje wykresy wielomianu interpolacyjnego na tle funkcji Rungego:

```
import numpy as np
import matplotlib.pyplot as plt
import scipy.interpolate as inter

def f(x): return 1/(25*(x**2) + 1)

def main(n:int):
    d = np.linspace(start=-1 , stop=1 , num=1000 , dtype=float) # domain
    knots = np.linspace(start=-1 , stop=1 , num=n , dtype=float)
    f_interpolating = inter.KroghInterpolator(xi=knots , yi=f(knots))
    plt.plot(d , f(d) , label='f(x)')
```

```
    plt.plot(d , f_interpolating(d) , label='f_interpolating(x)')
    plt.scatter(knots , f(knots) , color='red')
    plt.legend()
    plt.title(f'Interpolation with {n} knots')
    plt.show()

for n in range(4, 11+1):
    main(n)
```



Rysunek 2.1: Wykres funkcji interpolującej na tle funkcji interpolowanej dla 4-9 węzłów



Rysunek 2.2: Wykres funkcji interpolującej na tle funkcji interpolowanej dla 10-11 węzłów

Wnioski:

Dla małych n parzystych rozbieżności są mniejsze niż dla nieparzystych, lecz najlepszych kompromisem wydaje się być n = 8.

Zad 2

(a) sprawdzić czy pierwsze siedem wielomianów Legendre'a są wzajemnie ortogonalne

(b) sprawdzić czy one spełniają wzór na rekurencję

(c) wyrazić każdy z sześciu pierwszych jednomianów 1, t, …, t^6 jako liniową kombinację pierwszych siedmiu wielomianów Legendre'a, p0, …, p6

(a) Jeśli wielomiany Legendre'a są ortogonalne na przedziale [-1,1] względem iloczynu skalarnego w przestrzeni L^2 to:

$$\forall\, n, m \in \{0,1,2,\dots,6\}, \qquad n \neq m\ , \qquad \int_{-1}^{1} P_n(x)P_m(x)\ dx = \mathbf{0}$$

Wielomiany Legendre'a są postaci:

$$P_n(x) = \frac{1}{2^n n!}\cdot\frac{d^n}{dx^n}(x^2-1)^n$$

Rozważając całke postaci:

$$2^n n! \int_{-1}^{1} f(x)P_n(x)\ dx \xRightarrow{P_n(x) = \frac{1}{2^n n!}\frac{d^n}{dx^n}(x^2-1)^n} 2^n n! \int_{-1}^{1} f(x)\cdot\frac{1}{2^n n!}\cdot\frac{d^n}{dx^n}(x^2-1)^n\,dx =$$

$$= \int_{-1}^{1} f(x)\frac{d^n}{dx^n}(x^2-1)^n\,dx =$$

Używając całkowania przez części:

$$u = f(x) \qquad v' = \frac{d^n}{dx^n}(x^2-1)^n$$

$$u' = f'(x) \qquad v = \frac{d^{n-1}}{dx^{n-1}}(x^2-1)^n$$

$$= \left[f(x) \frac{d^{n-1}}{dx^{n-1}} (x^2 - 1)^n \right]_{-1}^{1} - \int_{-1}^{1} f'(x) \frac{d^{n-1}}{dx^{n-1}} (x^2 - 1)^n \, dx =$$

Pierwszy człon się upraszcza bo wstawiając do wyrażenia $(x^2 - 1)^n$ wartości 1 , -1 mamy $0^n = 0$ a pochodna 0 to dalej 0

$$= - \int_{-1}^{1} f'(x) \frac{d^{n-1}}{dx^{n-1}} (x^2 - 1)^n \, dx$$

Otrzymaliśmy rekurencyjny wzór na całkę, stosując go :

$$\int_{-1}^{1} f(x) P_n(x) \; dx = \frac{1}{2^n n!} \int_{-1}^{1} f^{(n)}(x) (x^2 - 1)^n \; dx$$

Przyjmując $m \; < \; n$ to stopień wielomianu P_m jest co najwyżej równy $n - 1$, tak więc n-ta pochodna wynosi 0. Po podstawieniu $f(x) = P_m(x)$:

$$\int_{-1}^{1} P_m(x) P_n(x) \; dx = \frac{1}{2^n n!} \int_{-1}^{1} P_m^{(n)}(x) (x^2 - 1)^n \; dx = 0$$

Wnioski:
Z tego wynika że kolejne wielomiany Legendre'a są ortogonalne w sensie metryki z przestrzeni L^2 w tym m.in. siedem pierwszych

(b) Wzór rekurencyjny:

$$P_{n+1}(x) = \frac{2n+1}{n+1} x P_n(x) - \frac{n}{n+1} P_{n-1}(x) \quad (n = 1, 2, \ldots).$$

$$P_0(x) = 1, \qquad P_1(x) = x$$

Korzystając z poniższego skryptu Python obliczam kolejne wyrazy:

```
import numpy as np

arr = []

# Tworzymy pierwsze dwa wielomiany Legendre'a

arr.append(np.poly1d([1]))

arr.append(np.poly1d([1, 0]))

# Definiujemy zmienną x

x = np.poly1d([1, 0])

# Generujemy kolejne wielomiany Legendre'a

for n in range(1, 6):

    new_poly = ((2 * n + 1) / (n + 1)) * arr[n] * x - (n / (n + 1)) * arr[n - 1]

    arr.append(new_poly)
```

```
# Wyświetlanie
for i , el in enumerate(arr):
   print(f"Legendre_{i} =")
   print(el)
```

Otrzymuje:

```
Legendre_0 =


1
Legendre_1 =


1 x
Legendre_2 =
    2
1.5 x - 0.5
Legendre_3 =
    3
2.5 x - 1.5 x
Legendre_4 =
      4       2
4.375 x - 3.75 x + 0.375
Legendre_5 =
      5       3
7.875 x - 8.75 x + 1.875 x
Legendre_6 =
      6        4        2
14.44 x - 19.69 x + 6.562 x - 0.3125
```

Wnioski:

Powyższe wielomiany są takie same jak podawane w tablicach

(c) Używając obliczonych wielomianów Legendre'a

$1 = \mathrm{P}_0$

$t = \mathrm{P}_1$

$t^2 = \frac{2}{3}\left(\frac{1}{2}P_0 + P_2\right)$

$t^3 = \frac{2}{5}\left(P_3 - \frac{3}{2}P_1\right)$

$t^4 = \frac{7}{35}P_0 + \frac{20}{35}P_2 + \frac{8}{35}P_4$

$t^5 = \frac{27}{63}P_1 + \frac{28}{63}P_3 + \frac{8}{63}P_5$

$t^6 = \frac{33}{231}P_0 + \frac{110}{231}P_2 + \frac{72}{231}P_4 + \frac{16}{231}P_6$

Zad 3

Dana jest funkcja określona w trzech punktach x0, x1, x2,

rozmieszczonych w jednakowych odstępach (x1=x0+h, x2=x1+h):

f(x0) = y0, f(x1) = y1, f(x2) = y2

Proszę wykonać interpolację danej funkcji sklejanymi funkcjami sześciennymi.

Mając punkty:

$$(x_0, y_0), \qquad (x_1, y_1), \qquad (x_2, y_2)$$

Niech s_i oznacza interpolującą funkcje sześcienna taką, że:

$$s_i(x) = \mathrm{a_i} + \mathrm{b_i}(\mathrm{x} - \mathrm{x_i}) + \mathrm{c_i}(\mathrm{x} - \mathrm{x_i})^2 + \mathrm{d_i}(\mathrm{x} - \mathrm{x_i})^3 \text{ dla } x\epsilon[x_i, x_{i+1}] \text{ w.p.p } s_i(x) = 0$$

Używając tego, że interpolacja podziałowa, liniowa dla $x\epsilon[x_i, x_{i+1}]$ jest postaci:

$$y(x) = \frac{x_{i+1} - x}{x_{i+1} - x_i} \cdot y_i + \frac{x - x_i}{x_{i+1} - x_i} \cdot y_{i+1}$$

Oraz że druga pochodna $s_i(x)$ jest liniowa:

$$s_{\mathrm{i}}^{(2)}(x) = \frac{s_{\mathrm{i}}^{(2)}(x_i)(x_{i+1} - x)}{h} + \frac{s_{\mathrm{i}}^{(2)}(x_{i+1})(x - x_i)}{h}$$

Całkując dwukrotnie otrzymuję:

$$s_i(x) = \frac{s_{\mathrm{i}}^{(2)}(x_i)}{6h}(x_{i+1} - x)^3 + \frac{s_{\mathrm{i}}^{(2)}(x_{i+1})}{6h}(x - x_i)^3 + C(x - x_i) + D(x_{i+1} - x)$$

gdzie C, D - stałe całkowania,

korzystając z warunków interpolacji obliczam C , D:

$s_i(x_i) = y_i \quad , \quad s_i(x_{i+1}) = y_{i+1}$

$$s_i(x) = \frac{s_{\mathrm{i}}^{(2)}(x_i)}{6h}(x_{i+1}-x)^3 + \frac{s_{\mathrm{i}}^{(2)}(x_{i+1})}{6h}(x-x_i)^3 + \left(\frac{y_{i+1}}{h} - \frac{s_{\mathrm{i}}^{(2)}(x_{i+1})\mathrm{h}}{6}\right)(x-x_i)$$
$$+ \left(\frac{y}{h} - \frac{s_{\mathrm{i}}^{(2)}(x_i)\mathrm{h}}{6}\right)(x_{i+1}-x)$$

W powyższym wzorze nie znamy $s_i''(x)$. Aby je wyliczyć korzystamy z warunku ciągłości pierwszej pochodnej. Różniczkujemy więc $s_i(x)$

$$s_i'(x_i) = -\frac{h}{3}s_{\mathrm{i}}^{(2)}(x_i) - \frac{h}{6}s_{\mathrm{i}}^{(2)}(x_{i+1}) - \frac{y_i + y_{i+1}}{h}$$

Dla przejrzystości wprowadzam symbole:

$$\sigma_i = \frac{1}{6}s''(x_i) \quad , \qquad \Delta_i = \frac{y_{i+1}-y_i}{h}$$

Wtedy:

$s_i'(x_i) = -2\sigma_i h - \sigma_{i+1}h + \Delta_i$

$s_i'(x_i) = \Delta_i - h(\sigma_{i+1} + 2\sigma_i)$

$s_{i-1}'(x_i) = \Delta_{i-1} + h(2\sigma_i + \sigma_{i-1})$

Warunkiem ciągłości jest: $s_i'(x_i) = s_{i-1}'(x_i)$

$$\Delta_{i-1} + h(2\sigma_i + \sigma_{i-1}) = \Delta_i - h(\sigma_{i+1} + 2\sigma_i)$$

Otrzymujemy wtedy układ (n-2) równań liniowych:

$$h\sigma_{i-1} + 4h\sigma_i + h\sigma_{i+1} = \Delta_i - \Delta_{i-1}, \quad i = 2,3,\dots,n-1$$

Ponieważ mamy n niewiadomych σ_i konieczne jest określanie dwóch dodatkowych warunków

$$\begin{matrix} -h\sigma_1 + h\sigma_2 = h^2\,\Delta_1^{(3)} & & -\sigma_1 + \sigma_2 = h\Delta_1^{(3)} \\ & => & \\ h\sigma_{n-1} - h\sigma_n = -h^2\,\Delta_{n-3}^{(3)} & & \sigma_{n-1} - \sigma_n = -h\Delta_{n-3}^{(3)} \end{matrix}$$

Niestety mając tylko 3 punkty nie jesteśmy wstanie skorzystać poprawnie z tej metody potrzebujemy min 4 punktów, tak wynika z logiki wszystkich powyższych równań.

Dla czterech punktów wynikiem jest rozwiązanie układu równań:

$$\begin{bmatrix} -1 & 1 & 0 \\ h & 4h & h \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \end{bmatrix} = \begin{bmatrix} h\Delta_1^{(3)} \\ \Delta_2 - \Delta_1 \\ -h\Delta_{n-3}^{(3)} \end{bmatrix}$$

Bibliografia:

https://home.agh.edu.pl/~funika/mownit/lab3/int_fun.pdf
wykład dr inż. Katarzyna Rycerz
https://pl.wikipedia.org/wiki/Wielomiany_Legendre%E2%80%99a