Notatki z Metod Numerycznych

2025-02-25 – 2025-06-17

Spis treści

Przedmowa

To są notatki z przedmiotu rachunek prawdopodobieństwa prowadzonego na kierunku IAD w 2024/2025 roku przez dr Tomasza Krajkę. Treści obejmują 15 wykładów. Notatki zawierają nie tylko treści omówione na wykładach, ale też implementacje algorytmów w Octave które powstały podczas zajęć laboratoryjnych.

Notatki mogą zawierać błędy (w tym gramatyczne).

[1] https://creativecommons.org/publicdomain/zero/1.0/deed.pl

2025-02-25

Achtung! 1

Warunki zaliczenia labów:

- projekt
- aktywność

Warunki zaliczenia wykładu:

- pierwszy termin pisemny albo ustny
- pozostałe ustne

1 Reprezentacja liczb w pamięci komputera. Arytmetyka zmiennopozycyjna, błędy obliczeń

Metody numeryczne (*analiza numeryczna*) to dział matematyki stosowanej zajmujący się opracowywaniem i badaniem metod przybliżonego rozwiązywania problemów obliczeniowych.

Najczęściej metody te przyjmują postać algorytmów które pozwalają rozwiązywać rozważane problemy za pomocą komputera.

1.1 Numeryczna reprezentacja liczb

W pamięci komputera liczby nie są reprezentowane w systemie dziesiętnym. Najczęściej stosowaną podstawą rozwinięć liczb 2(arytmetyka dwójkowa, binarna). W niektórych sytuacjach stosowane są również systemy przy podstawie 8 albo 16.

1.1.1 Reprezentacja liczb całkowitych

Liczby całkowite reprezentowane są w pamięci komputera w sposób stałoprzecinkowy

Twierdzenie 1.1

Dowolną liczbę całkowitą l można przedstawić w postaci jej rozwinięcia dwójkowego:

$$L = s\sum_{i=0}^{n} e_i \cdot 2^i \qquad (1)$$

gdzie $s \in \{-1, 1\}$ jest znakiem liczby, $e_i \in \{0, 1\}$ są jej cyframi rozwinięcia dwójkowego oraz $e_n \neq 0$ jeśli $l \neq 0$

W ogólności w systemie liczbowym przy podstawie p, dowolną liczbę całkowitą l można przedstawić w postaci

$$l = s \sum_{i=0}^{n} e_i \cdot p^i \qquad (2)$$

gdzie $s \in \{-1, 1\}$ jest znakiem liczby, $e_i \in \{0, 1, \ldots, p-1\}$ są jej cyframi rozwinięcia w systemie przy podstawie p oraz $e_n \neq 0$ jeśli $l \neq 0$

W ten sposób można reprezentować liczby całkowite z przedziału $l \in [-2^d + 1, 2^d - 1]$. Jeżeli argumenty działań arytmetycznych na liczbach całkowitych i ich wynik są reprezentowalne, to działania są wykonane dokładnie. Liczby całkowite w pamięci komputera mogą być w sposób powyższy na dwa sposoby: w postaci znak-moduł albo w postaci znak-uzupełnienie

1.1.2 Reprezentacja liczb rzeczywistych

Twierdzenie 1.2

Każdą liczbę rzeczywistą x można zapisać w postaci:

$$x = s \cdot 2^c \cdot m \qquad (3)$$

gdzie $s \in \{-1, 1\}$ jest znakiem liczby, c jest liczbą całkowitą zwaną cechą, a $m \in [\frac{1}{2}, 1)$ jest liczbą rzeczywistą zwaną mantysą.
Mantysa ma postać

$$m = \sum_{i=1}^{\infty} e_{-i} \cdot 2^{-i} \qquad (4)$$

gdzie $e_{-1} = 1$ a $e_{-i} \in \{0, 1\}$ dla $i > 1$. Jeśli $x \neq 0$, to takie przedstawienie jest jednoznaczne

W praktyce nie da się reprezentować mantysy z nieskończoną dokładnością. Przeznaczając t bitów na reprezentację mantysy otrzymujemy jej t-bitową reprezentacje(oznaczaną m_t)

$$m_t = \sum_{i=1}^{t} e_{-i} \cdot 2^{-i} + e_{-(t+1)} \cdot 2^{-t} \qquad (5)$$

Następuje tu zatem zaokrąglenie wartości liczbowej mantysy do t bitów. Pojawia się tu zatem błąd reprezentacji który można oszacować w sposób bezwzględny.

$$|m - m_t| \leq \frac{1}{2} \cdot 2^{-t} \qquad (6)$$

Reprezentacją liczby rzeczywistej x w pamięci komputera będziemy oznaczać przez $\mathrm{rd}(x)$ i definiujemy wzorem

$$\mathrm{rd}(x) = s \cdot 2^c \cdot m_t \qquad (7)$$

Czyli do reprezentacji liczby rzeczywistej potrzebujemy 3 liczb(s, c, m_t).

Błąd względny takiej reprezentacji można oszacować następująco:

$$\left|\frac{\mathrm{rd}(x) - x}{x}\right| = \left|\frac{s \cdot 2^c \cdot m_t - s \cdot 2^c \cdot m}{s \cdot 2^c \cdot m}\right| = \frac{|m_t - m|}{m} \leq \frac{\frac{1}{2} \cdot 2^{-t}}{\frac{1}{2}} = 2^{-t} \qquad (8)$$

Albo inaczej

$$\mathrm{rd}(x) = x(1 + \varepsilon), \quad |\varepsilon| \leq 2^{-t} \qquad (9)$$

2025-03-04

1.1.3 Reprezentacja liczb rzeczywistych cd.

Twierdzenie 1.3

W arytmetyce zmiennopozycyjnej można zatem reprezentować liczby rzeczywiste $x \neq 0$ z przedziału

$$\frac{1}{2} \cdot 2^{C_{\min}} \leq |x| < 2^{C_{\max}} \qquad (10)$$

gdzie $C_{\min}$ i $C_{\max}$ są odpowiednio najmniejszą i największą możliwą do reprezentowania wartością cechy.

W przypadku próby reprezentowania liczby mniejszej niż dolne ograniczenie mamy do czynienia z niedomiarem – liczba jest reprezentowana za pomocą 0, a w przypadku próby reprezentowania liczby większej od górnego ograniczenia z nadmiarem, następuje przerwanie obliczeń

Przykład 1.1

$$t = 4 \quad 0{,}33_{(10)} = 0{,}010101_{(2)} \cdot 2^{-1} \qquad (11)$$

$$m_t = 0{,}1010_{(2)} + 0{,}0001_{(2)} = 0{,}1011_{(2)} \quad \mathrm{rd}(x) = 0{,}1011 \cdot 2^{-1} \qquad (12)$$

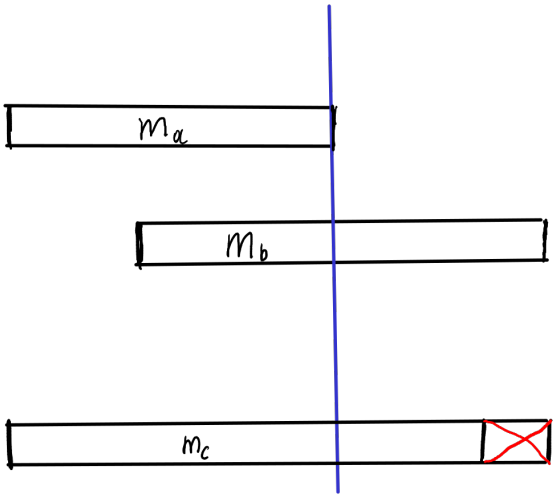
$$\begin{array}{r|l} 0{,}33 & \\ 0{,}66 & 0 \\ 1{,}32 & 1 \\ 0{,}32 & \\ 0{,}64 & 0 \\ 1{,}28 & 1 \\ 0{,}28 & \\ 0{,}56 & 0 \\ 1{,}06 & 1 \end{array} \qquad (13)$$

Działania na reprezentacjach liczb rzeczywistych są wykonywane w arytmetyce zmiennoprzecinkowej (fl). Niech $\tilde{a} = \mathrm{rd}(a), \tilde{b} = \mathrm{rd}(\mathrm{b})$, wtedy

$$\mathrm{fl}(\tilde{a} \square \tilde{b}) = \mathrm{rd}(\tilde{a} \square \tilde{b}) = (\tilde{a} \square \tilde{b})(1 + \varepsilon), \quad |\varepsilon| \leq 2^{-t} \qquad (14)$$

gdzie $\square \in \{+, -, *, /\}$.

Każde działanie arytmetyczne w arytmetyce fl wytwarza pewien mały względny błąd zaokrąglenia rzędu błędu reprezentacji



Rysunek 1: Wykres mantys

Przykładowo wykonując działanie "+" na reprezentacjach liczb $a \geq b$, $\mathrm{rd}(a) = m_a \cdot 2^{C_a}$ oraz $\mathrm{rd}(b) = m_b \cdot 2^{C_b}$, wówczas mamy

$$a + b = 2^{C_a}(m_a + m_b \cdot 2^{-(C_a - C_b)}) = 2^{C_a} \cdot m_s \qquad (15)$$

gdzie mantysa m_s jest obarczona błędem jej reprezentacji

Przykład 1.2

Niech $t = 4$ oraz

$$a = \mathrm{rd}(a) = 0{,}1101 \cdot 2^{-1} \quad b = \mathrm{rd}(b) = 0{,}1001 \cdot 2^{1} \qquad (16)$$

$$\begin{aligned} \mathrm{fl}(a + b) &= \mathrm{rd}(0{,}001101 + 0{,}1001) \cdot 2^{1} \\ &= \mathrm{rd}(0{,}110001) \cdot 2^{1} \\ &= 0{,}1100 \cdot 2^{1} \end{aligned} \qquad (17)$$

1.2 Uwarunkowanie zadania

Załóżmy, że naszym zadaniem jest obliczenie wektora wyników z jakiejś przestrzeni wyników $\overline{w} \in \mathbb{R}_w^C$ dla wektora danych $\overline{d} \in R_d$

$$\overline{w} = \varphi(\overline{d}) \quad \overline{d} \in D_0 \subset R_d \qquad (18)$$

gdzie

- R_d, R_w są skończenie wymiarowymi unormonowanymi przestrzeniami kartezjańskimi

Np.

$$\begin{aligned} & A \in R^{n\times m}, \\ \overline{y} = A\overline{x}, \quad & \overline{x} \in R^m = R_d, \\ & \overline{y} \in R^n = R_w \end{aligned} \qquad (19)$$

$$\begin{aligned} & s \neq 0, \\ s = \sum_{i=1}^{n} x_i, \quad & \overline{x} \in R^n = R_d \\ & s \in R = R_w \end{aligned} \qquad (20)$$

Zadanie dobrze uwarunkowane – Jeśli niewielkie względne zaburzenia danych powodują duże względne zaburzenia wyników, to zadanie nazywamy źle uwarunkowanym. W przeciwnym razie zadanie jest dobrze uwarunkowane

Przykład 1.3

$$s = \sum_{i=1}^{n} x_i, \qquad s \neq 0 \qquad (21)$$

Zamiast wartości x_i, mamy ich reprezentacje $\mathrm{rd}(x_i) = x_i(1+\varepsilon_i)$, gdzie $|\varepsilon_i| \leq 2^{-t}, i = 1, 2, \ldots, n$

$$\begin{aligned} \frac{\tilde{s}-s}{|s|} &= \frac{|\sum_{i=1}^n x_i(1+\varepsilon_i) - \sum_{i=1}^n x_i|}{|\sum_{i=1}^n x_i|} \\ &= \frac{|\sum_{i=1}^n x_i\varepsilon_i|}{|\sum_{i=1}^n x_i|} \\ &\leq \frac{\sum_{i=1}^n |x_i\varepsilon_i|}{|\sum_{i=1}^n x_i|} \\ &\leq (\max_{1\leq i\leq n} \varepsilon_i) \cdot \frac{\sum_{i=1}^n |x_i|}{|\sum_{i=1}^n x_i|} \\ &\leq 2^{-t} \cdot \frac{\sum_{i=1}^n |x_i|}{|\sum_{i=1}^n x_i|} \end{aligned} \qquad (22)$$

Wskaźnik uwarunkowania Wielkość $\frac{\sum_{i=1}^n |x_i|}{|\sum_{i=1}^n x_i|}$ nazywamy wskaźnikiem uwarunkowania za-

dania i oznaczamy $\mathrm{cond}(\overline{x})$. Mówi ona z jakim czynnikiem błędy reprezentacji danych przeniosły się na błędy wyniku.

Z najlepszą sytuacją mamy do czynienia wtedy, gdy $\mathrm{cond}(\overline{x}) \approx 1$, wówczas błędy danych przeniosły się na błędy wyniku bez ich zwielokrotnienia

W przykładzie 1.3 $\mathrm{cond}(\overline{x}) = 1$ jeśli wszystkie wartości x_i są tego samego znaku i zadanie jest wówczas bardzo dobrze uwarunkowane

Przykład 1.4

Twierdzenie 1.4: Rozwinięcie w szereg Taylora funkcji dwóch zmiennych

$$f(x+\Delta x, y+\Delta y) = f(x,y) + \frac{\partial f}{\partial x}(x,y)\cdot\Delta x + \frac{\partial f}{\partial y}(x,y)\Delta y + \dots \tag{23}$$

Dany jest trójmian $x^2 - 2px + q$, gdzie $(p,q) \in D$ oraz $D = \{(p,q) : p \neq 0 \wedge q \neq 0 \wedge \underbrace{p^2 - q > 0}_{\Delta > 0}\}$

Zbadajmy jak zaburzenie wartości pierwiastków $x_1(p,q)$ i $x_2(p,q)$ zależą od drobnych zaburzeń współczynników p i q. Dla pierwiastka $k \in \{1,2\}$ i ustalonych wartości współczynników p_0 i q_0 mamy zatem

$$\begin{aligned}\frac{x_k(p_0(1+\varepsilon_1), q_0(1+\varepsilon_2)) - x_k(p_0,q_0)}{x_k(p_0,q_0)} \approx &\frac{x_k(p_0,q_0)}{x_k} \\ &+ \frac{\partial x_k}{\partial p}\Big|_{(p_0,q_0)} \cdot \frac{p_0\varepsilon}{x_k} \\ &+ \frac{\partial x_k}{\partial q}\Big|_{(p_0,q_0)} \cdot \frac{q_0\varepsilon}{x_k} - \frac{x_k(p_0,q_0)}{x_k}\end{aligned} \tag{24}$$

$$\begin{aligned} x_1 &= \frac{2p + 2\sqrt{p^2-p}}{2} = p + \sqrt{p^2-q} \\ x_2 &= p - \sqrt{p^2-q}\end{aligned} \tag{25}$$

$$\begin{aligned}\frac{\partial x_1}{\partial p} &= 1 + \frac{2p}{2\sqrt{p^2-q}} = \frac{p+\sqrt{p^2-q}}{\sqrt{p^2-q}} \\ \frac{\partial x_2}{\partial p} &= 1 - \frac{p}{\sqrt{p^2-q}} = \frac{\sqrt{p^2-q}-p}{\sqrt{p^2-q}}\end{aligned} \tag{26}$$

Analogicznie dla pochodnych po q

Zatem

$$\mathrm{cond}_p^k(p_0, q_0) = \frac{\partial x_k}{\partial p}\Big|_{(p_0,q_0)} \cdot \frac{p_0}{x_k} = \frac{x_k}{x_k - p_0} \cdot \frac{p_0}{x_k} \qquad (27)$$

oraz

$$\mathrm{cond}_q^k(p_0, q_0) = \frac{\partial x_k}{\partial q}\Big|_{(p_0,q_0)} \cdot \frac{q_0}{x_k} = \frac{-q_0}{2x_k(x_k - p_0)} \qquad (28)$$

Wstawiając teraz w miejsce x_k wartości pierwiastków $x_1 = p_0 + \sqrt{p_0^2 - q}$ i $x_2 = p_0 - \sqrt{p_0^2 - q}$ mamy

$$\mathrm{cond}_p^1(p_0, q_0) = -\mathrm{cond}_p^2(p_0, q_0) = \frac{1}{\sqrt{1 - \frac{q_0}{p_0^2}}} \qquad (29)$$

$$\mathrm{cond}_q^1(p_0, q_0) = \frac{1 - \sqrt{1 - \frac{q_0}{p_0^2}}}{2\sqrt{1 - \frac{q_0}{p_0^2}}} \qquad (30)$$

$$\mathrm{cond}_q^2(p_0, q_0) = \frac{1 + \sqrt{1 - \frac{q_0}{p_0^2}}}{-2\sqrt{1 - \frac{q_0}{p_0^2}}} \qquad (31)$$

Jak widać dla $\frac{q_0}{p_0^2} \approx 1$ współczynnik rośnie w sposób nieograniczony i zadanie jest BARDZO źle uwarunkowane. Jeśli $\frac{q_0}{p_0^2} \ll 1$, to zadanie jest BARDZO dobrze uwarunkowane

---

2025-03-11

1.3 Algorytmy numerycznie poprawne

Przykład 1.5

Rozważmy dwa algorytmy wyznaczające wartość $y = a^2 - b^2$

Algorytm A1

$$\begin{aligned} z_1 &= a - b \\ z_2 &= a + b \qquad (32) \\ y &= z_1 \cdot z_2 \end{aligned}$$

Algorytm A2

$$\begin{aligned} z_1 &= a \cdot a \\ z_2 &= b \cdot b \\ y &= z_1 - z_2 \end{aligned} \tag{33}$$

Przyjmujmy się teraz jak wygląda realizacja tych algorytmów w arytmetyce fl.
Realizacja algorytmu A1

$$\begin{aligned} \mathrm{fl}(z_1) &= (a-b)(1+\varepsilon_1), \quad |\varepsilon_1| \le 2^{-t} \\ \mathrm{fl}(z_2) &= (a+b)(1+\varepsilon_2), \quad |\varepsilon_2| \le 2^{-t} \\ \mathrm{fl}(y) &= \mathrm{fl}(z1) \cdot \mathrm{fl}(z2) \cdot (1+\varepsilon_3) \qquad \left(|\varepsilon_3| < 2^{-t}\right) \\ &= (a-b)(1+\varepsilon_1) \cdot (a+b)(1+\varepsilon_2)(1+\varepsilon_3)) \\ &= (a^2-b^2)(1+\varepsilon_1)(1+\varepsilon_2)(1+\varepsilon_3) \\ 1+E_1 &= (1+\varepsilon_1)(1+\varepsilon_2)(1+\varepsilon_3) \approx 1+\varepsilon_1+\varepsilon_2+\varepsilon_3 \end{aligned} \tag{34}$$

Stąd

$$|E_1| \le |\varepsilon_1| + |\varepsilon_2| + |\varepsilon_3| \le 3 \cdot 2^{-t} \tag{35}$$

Realizacja algorytmu A2

$$\begin{aligned} \mathrm{fl}(z_1) &= a \cdot a \cdot (1+\eta_1), \quad |\eta_1| \le 2^{-t} \\ \mathrm{fl}(z_2) &= b \cdot b \cdot (1+\eta_2), \quad |\eta_2| \le 2^{-t} \\ \mathrm{fl}(y) &= (\mathrm{fl}(z_1) - \mathrm{fl}(z_2))(1+\eta_3) \qquad \left(|\eta_3| \le 2^{-t}\right) \\ &= (a^2(1+\eta_1) - b^2(1+\eta_2))(1+\eta_3) \\ &= (a^2 - b^2 + a^2\eta_1 - b^2\eta_2)(1+\eta_3) \\ &= (a^2-b^2)(1+\frac{a^2\eta_1 - b^2\eta_2}{a^2-b^2})(1+\eta_3) \\ &= (a^2-b^2)(1+E_2) \\ 1+E_2 &= (1+\frac{a^2\eta_1 - b^2\eta_2}{a^2-b^2})(1+\eta_3) \approx 1+\eta_3+\frac{a^2\eta_1 - b^2\eta_2}{a^2-b^2} \\ |E_2| &\le |\eta_3| + \frac{a^2|\eta_1| + b^2|\eta_2|}{|a^2-b^2|} \le 2^{-t}(1+\frac{a^2+b^2}{|a^2-b^2|}) \end{aligned} \tag{36}$$

Dla $a^2 \approx b^2$ błąd wyniku w realizacji algorytmem A2 może być bardzo duży

Numeryczna poprawność Algorytm nazywamy numerycznie poprawnym jeśli obliczone w arytmetyce fl rozwiązanie jest nieco zaburzonym rozwiązaniem dokładnym dla nieco zaburzonych

danych.

Dla A1 mamy

$$\begin{aligned} \mathrm{fl}(y) &= (a^2 - b^2)(1+\varepsilon_1)(1+\varepsilon_2)(1\varepsilon_2) \\ &= ((a\sqrt{(1+\varepsilon_1)(1+\varepsilon_2)})^2 - (b\sqrt{(1+\varepsilon_1)(1+\varepsilon_2)})^2)(1+\varepsilon_3) \\ &= ((a(1+\xi_1))^2 - (b(1+\xi_1))^2)(1+\varepsilon_3) \\ 1+\varepsilon_1 &= \sqrt{(1+\varepsilon_1)(1+\varepsilon_2)} \\ 1+2\xi_1+\xi_1^2 &= 1+\varepsilon_1+\varepsilon_2+\varepsilon_1\varepsilon_3 \\ 2\xi_1 &= \varepsilon_1+\varepsilon_2, \qquad |\xi_1| \le 2^{-t} \end{aligned} \tag{37}$$

Realizacja algorytmu w arytmetyce fl polega na:

- Zastąpieniu danych ich reprezentacjami w arytmetyce fl
- Wykonaniu działań w arytmetyce fl

Przykład 1.6

Wyznaczmy $\sum_{i=1}^{n} x_i, s \neq 0$. Algorytm ma postać:

$$\begin{aligned} s_1 &= x_1 \\ s_2 &= s_1 + x_2 \\ s_n &= s_{n-1} + x_n \end{aligned} \tag{38}$$

$$\tilde{x}_i = \mathrm{rd}(x_i) = x_i(1+\alpha_i), \quad |\alpha_i| \le 2^{-t}, i \in \{1,2,\ldots,n\} \tag{39}$$

$$\begin{aligned} \mathrm{fl}(s_1) &= \tilde{x}_1 \\ \mathrm{fl}(s_2) &= (\mathrm{fl}(s_1)+\tilde{x}_2)(1+\varepsilon_2) = (\tilde{x}_1+\tilde{x}_2)(1+\varepsilon_2), \quad |\varepsilon_2| \le 2^{-t} \\ \mathrm{fl}(s_3) &= (\mathrm{fl}(s_2)+\tilde{x}_3)(1+\varepsilon_3) = ((\tilde{x}_1+\tilde{x}_2)(1+\varepsilon_2)+\tilde{x}_3)(1+\varepsilon_3)) \\ &= \tilde{x}_1(1+\varepsilon_1)(1+\varepsilon_2)(1+\varepsilon_3) + \tilde{x}_2(1+\varepsilon_2)(1+\varepsilon_3) + \tilde{x}_3(1+\varepsilon_3) \end{aligned} \tag{40}$$

Ogólnie

$$\begin{aligned}\mathrm{fl}(s_n) &= \sum_{k=1}^{n} \tilde{x}_k \prod_{i=k}^{n}(1+\varepsilon_i) \\ &= \sum_{k=1}^{n} x_k(1+\varepsilon_k)\prod_{i=k}^{n}(1+\varepsilon_i) \qquad (41)\\ &= \sum_{k=1}^{n} x_k(1+E_k)\end{aligned}$$

$$1 + E_k = 1 + \alpha_k + \sum_{i=k}^{n} \varepsilon_i \qquad (42)$$

$$|E_k| \leq \begin{cases} n \cdot 2^{-t}, & k = 1 \\ (n-k+2)\cdot 2^{-t}, & k > 1 \end{cases} \qquad (43)$$

Zatem największym błędem obarczone są początkowe składniki sumy. Algorytm jest numerycznie poprawny, a najlepszą metodę realizacji tego algorytmu jest uprzednie posortowanie rosnąco sumowanych elementów według ich wartości bezwzględnych

2 Rozwiązanie układu równań liniowych

Niech będzie dany układ równań liniowych

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = b_n \end{cases} \qquad (44)$$

gdzie a_{ij}, x_i oraz b_i dla $1 \leq i, j \leq n$ są liczbami rzeczywistymi

Ten układ równań można także zapisać w postaci macierzowej

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \qquad (45)$$

albo w skróconej postaci macierzowej

$$Ax = b \qquad (46)$$

gdzie $A \in \mathbb{R}_{n\times n}$ jest macierzą kwadratową rozmiaru n, a $x, b \in \mathbb{R}_{n\times 1}$ są wektorami

Twierdzenie 2.1

Układ równań $Ax = b$ ma dokładnie 1 rozwiązanie wtw, gdy $\det A \neq 0$, czyli gdy macierz współczynników układu równań jest nieosobliwa

Algebraicznie, tego typu układy równań rozwiązuje się poprzez przemnożenie obu stron równania $Ax = b$ przez macierz odwrotną do macierzy A, czyli macierz A^{-1}. Jako że $A^{-1} \cdot A = I$, gdzie I jest macierzą jednostkową, mamy:

$$\begin{aligned} Ax &= b \quad \Big| \cdot A^{-1} \\ A^{-1}Ax &= A^{-1}b \\ Ix &= A^{-1}b \\ x &= A^{-1}b \end{aligned} \qquad (47)$$

2.1 Układy równań łatwe do rozwiązania

2.1.1 Macierz przekątniowa

Macierz A jest macierzą nieosobliwą i przekątniową, czyli $a_{kk} \neq 0$ dla $k \in \{1, 2, \ldots, n\}$ oraz $a_{ij} = 0$ dla $i \neq j$ (macierze przekątniowe będziemy oznaczali literą D)

$$\begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \qquad (48)$$

Rozwiązaniem takiego układu równań będzie wektor postaci

$$x = \begin{bmatrix} \frac{b_1}{a_{11}} \\ \frac{b_2}{a_{22}} \\ \vdots \\ \frac{b_n}{a_{nn}} \end{bmatrix} \qquad (49)$$

2.1.2 Macierz górnie trójkątna

Macierz A jest macierzą nieosobliwą i górnie trójkątną, czyli $a_{kk} \neq 0$, oraz $a_{ij} = 0$ dla $i > j$ (macierze górnie trójkątne będziemy oznaczali literą U)

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \qquad (50)$$

Taki układ równań rozwiązujemy za pomocą tzw. podstawiania wstecz wyrażonego wzorem

$$x_i = \frac{b_i - \sum_{j=i+1}^{n} a_{ij} x_j}{a_{ii}}, \quad 1 \leq i \leq n \qquad (51)$$

Wyznaczanie wektora x rozpoczynamy od ostatniej niewiadomej (x_n), i w kolejnych krokach przechodzimy do poprzednich współrzędnych, aż do wyznaczenia x_1

Przykładowa implementacja w Octave:

```
function x = rozw_u(A, b)
  x = [];
  [h, w] = size(A);
  for i = w:-1:1
    x(i) = (b(i) - sum(A(i, i+1:w) .* x(i+1:w))) / A(i, i);
  end
  x = x';
end
```

2.1.3 Macierz dolnie trójkątna

Macierz A jest macierzą nieosobliwą i dolnie trójkątną, czyli $a_{kk} \neq 0$ oraz $a_{ij} = 0$ dla $i < j$(macierze dolnie trójkątne będziemy oznaczali literą L)

$$\begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ a_{21} & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \tag{52}$$

Taki układ równań rozwiązujemy za pomocą tak zwanego podstawiania w przód, wyrażonego wzorem

$$x_i = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j}{a_{ii}} \quad 1 \leq i \leq n \tag{53}$$

Wyznaczanie wektora x rozpoczynamy od pierwszej niewiadomej (x_1) i w kolejnych krokach przechodzimy następnych współrzędnych aż do wyznaczenia (x_n)

Przykładowa implementacja w Octave:

```
function x = rozw_l(A, b)
  x = [];
  [h, w] = size(A);
  for i = 1:w
    x(i) = (b(i) - sum(A(i, 1:i-1) .* x(1:i-1))) / A(i, i);
  end
  x = x';
end
```

2025-03-18

2.1.4 Macierze rozkładalne

Macierz A da się rozłożyć do postaci iloczynu macierzy dolnie trójkątnej i górnie trójkątnej $A = L \cdot U$

$$\begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \cdot \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{14} \\ 0 & u_{22} & \cdots & u_{24} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{44} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \tag{54}$$

Wyznaczanie wektora x można podzielić wówczas na dwa etapy:

Oznaczając $Ux = z$, rozwiązujemy najpierw układ równań

$$L \cdot z = b \qquad (55)$$

gdzie

- L jest macierzą dolnie trójkątną

Zatem rozwiązania tego układu równań (wyznaczanie wektora z) odbywa się przez zastosowanie podstawiania wprzód (jak w punkcie 3)

Następnie rozwiązujemy układ równań $Ux = z$, gdzie U jest macierzą górną. Zatem rozwiązanie tego układu równań odbywa się poprzez zastosowanie podstawiania wstecz (jak w punkcie 2)

Większość dokładnych metod rozwiązywania układów równań liniowych polega na sprowadzeniu macierzy współczynników układu równań A, do jednej z powyższych postaci (ewentualnie ich iloczynu), a następnie rozwiązanie ich zgodnie z powyższymi wzorami.

2.2 Rozkłady $A = LU$

Jedną z metod przekształcenia macierzy A, jest jej rozkład na iloczyn macierzy dolnie trójkątnej i górnie trójkątnej $A = LU$. Jeśli rozkład taki istnieje to nie jest on jednoznaczny, należy z każdej pary liczb znajdujących się na głównych przekątnych macierzy L i U czyli l_{ii}, u_{ii}, nadać dokładnie jednej z nich dowolną wartość różną od zera.

Rozkład taki opiera się o wzór na mnożenie macierzy

$$\begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \cdot \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \qquad (56)$$

$$a_{ij} = \sum_{s=1}^{n} l_{is} \cdot u_{sj} = \sum_{s=1}^{\min(i,j)} l_{is} \cdot u_{sj} \qquad (57)$$

Ponieważ $l_{is} = 0$ dla $s > i$ i $u_{sj} = 0$ dla $s > j$, wystarczy ograniczyć się jedynie do $\min(i, j)$ składników tej sumy, jako że pozostałe są równe 0.

Kolejność wyznaczania elementów macierzy L i U

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ \downarrow & 2 & \cdots & 0 \\ \downarrow & \downarrow & \ddots & \vdots \\ \downarrow & \downarrow & \cdots & n \end{bmatrix} \cdot \begin{bmatrix} 1 & \rightarrow & \rightarrow & \rightarrow \\ 0 & 2 & \rightarrow & \rightarrow \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & n \end{bmatrix} \tag{58}$$

Metoda ta składa się z n kroków. W k-tym kroku metody rozpoczynamy od wyznaczania elementów na głównej przekątnej l_{kk} i u_{kk}. Jeden z tych elementów ma z góry nadaną wartość, zaś drugi wyznaczamy ze wzoru na element a_{kk} macierzy wynikowej

$$a_{kk} = \sum_{s=1}^{k-1} l_{ks} \cdot u_{sk} + l_{kk} \cdot u_{kk} \tag{59}$$

czyli

$$l_{kk}u_{kk} = a_{kk} - \sum_{s=1}^{k-1} l_{ks}u_{sk} \tag{60}$$

W dalszym ciągu k-tego kroku wyznaczamy resztę elementów k-tego wiersza macierzy U, oraz resztę k-tej kolumny macierzy L, ze wzorów na elementy a_{kj} i a_{ik}:

$$a_{kj} = \sum_{s=1}^{k-1} l_{ks}u_{sj} + l_{kk} \cdot u_{kj} \qquad (k+1 \leq j \leq n) \tag{61}$$

czyli

$$u_{kj} = \frac{a_{kj} - \sum_{s=1}^{k-1} l_{ks}u_{sj}}{l_{kk}} \tag{62}$$

oraz

$$a_{ik} = \sum_{s=1}^{k-1} l_{is}u_{sk} + l_{ik}u_{kk} \qquad (k+1 \leq i \leq n) \tag{63}$$

czyli

$$l_{ik} = \frac{a_{ik} - \sum_{s=1}^{k-1} l_{is}u_{sk}}{u_{kk}} \tag{64}$$

Czynności te powtarzamy dla kolejnych kroków aż do n-tego wyznaczając kolejne wiersze macierzy U, oraz kolejne kolumny macierzy

Twierdzenie 2.2

Jeżeli wszystkie minory główne macierzy kwadratowej A są nieosobliwe, to ma ona rozkład $A = LU$

Zatem metodę tę możemy stosować wówczas gdy wyznaczniki macierzy A_k postaci

$$A_k = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1k} \\ a_{21} & a_{22} & \cdots & a_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kk} \end{bmatrix} \tag{65}$$

dla wszystkich $1 \leq k \leq n$ są różne od zera

W zależności od sposobu w jaki zostaną ustalone elementy spośród par l_{ii} i u_{ii} mamy różne rozkłady LU macierzy A.

- Jeśli $l_{ii} = 1$ dla $1 \leq i \leq n$, czyli w macierzy dolnie trójkątnej elementy na głównej przekątnej mają wartość 1, to taki rozkład nazywamy rozkładem Doolittle'a.
- Jeśli $u_{ii} = 1$ dla $1 \leq i \leq n$, czyli w macierzy górnie trójkątnej elementy na głównej przekątnej mają wartość 1, to taki rozkład nazywamy rozkładem Croute'a
- Jeśli $l_{ii} = u_{ii} = 1$, to modyfikując nieco algorytm rozkładu LU możemy uzyskać rozkład $A = LDU$, gdzie macierze L i U mają wartości 1 na głównej przekątnej, a macierz D jest macierzą diagonalną
- Jeśli zachodzi $U = L^T$, czyli $u_{ij} = l_{ji}$, dla $i \leq i, j \leq n$, to taki rozkład nazywamy rozkładem Cholesky'ego.

Szczególnym spośród tych rozkładów jest rozkład Cholesky'ego. Wymaga on nieco mocniejszych założeń o macierzy A.

Twierdzenie 2.3

Jeśli macierz A jest rzeczywista, symetryczna ($A = A^T$) i dodatnio określona ($\bigwedge_{\vec{x} \neq \vec{0}} \vec{x}^T A \vec{x} > 0$), to ma ona rozkład $A = LL^T$, gdzie L jest macierzą dolnie trójkątną o dodatnich elementach na głównej przekątnej

W rozkładzie Cholesky'ego postępujemy tak samo jak w zwykłym rozkładzie LU, pamiętając że $u_{ij} = l_{ji}$. Mamy tu zatem

$$l_{kk} = \sqrt{a_{kk} - \sum_{s=1}^{k-1} l_{ks}^2} \qquad (66)$$

oraz

$$l_{ik} = \frac{a_{ik} - \sum_{s=1}^{k-1} l_{is} l_{ks}}{l_{kk}} \qquad (67)$$

Achtung! 2

Zauważmy, że z pierwszego z powyższych wzorów wynika, że

$$a_{kk} = \sum_{s=1}^{k} l_{ks}^2 \geq l_{kj}^2 \qquad (68)$$

dla dowolnego $j \leq k$, czyli

$$|l_{kj}| \leq \sqrt{a_{kk}} \quad 1 \leq j \leq k \qquad (69)$$

czyli każdy element wiersza macierzy L (odpowiednio kolumny macierzy L^T) może być ograniczony z góry przez pierwiastek z elementu na głównej przekątnej macierzy A w tym wierszu.

2025-03-25

Przykład 2.1

> Znajdź rozkłady Doolittle'a i Cholesky'ego macierzy

$$A = \begin{bmatrix} 60 & 30 & 20 \\ 30 & 20 & 15 \\ 20 & 15 & 12 \end{bmatrix} \qquad (70)$$

Rozkład Doolittle'a

$$
\begin{aligned}
l_{11} &= l_{22} = l_{33} = 1 & u_{11} &= \frac{a_{11}}{l_{11}} = \frac{60}{1} = 60 \\
u_{12} &= \frac{a_{12}}{l_{11}} = 30 & u_{13} &= \frac{a_{13}}{l_{11}} \\
l_{21} &= \frac{a_{21}}{u_{11}} = \frac{30}{60} = \frac{1}{2} & l_{31} &= \frac{a_{31}}{u_{11}} = \frac{20}{60} = \frac{1}{3} \\
u_{22} &= \frac{a_{22} - l_{21}u_{12}}{l_{22}} = \frac{20 - \frac{1}{2}\cdot 30}{1} = 5 & u_{23} &= \frac{a_{23} - l_{21}u_{13}}{l_{22}} = \frac{15 - \frac{1}{2}\cdot 20}{1} = 5 \\
l_{32} &= \frac{a_{32} - l_{31}u_{12}}{u_{22}} = \frac{15 - \frac{1}{3}\cdot 30}{5} = 1 & u_{33} &= \frac{a_{33} - l_{31}u_{13} - l_{32}u_{23}}{l_{33}} = \frac{12 - \frac{1}{3}\cdot 20 - 15}{1} = \frac{1}{3}
\end{aligned}
\tag{71}
$$

Zatem rozkład Doolittle'a ma postać

$$
\begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{2} & 1 & 0 \\ \frac{1}{3} & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 60 & 30 & 20 \\ 0 & 5 & 5 \\ 0 & 0 & \frac{1}{3} \end{bmatrix} = \begin{bmatrix} 60 & 30 & 20 \\ 30 & 20 & 15 \\ 20 & 15 & 12 \end{bmatrix} \tag{72}
$$

Przykładowa implementacja w Octave

```
function [L, U] = doolittle_dec(A)
  [h, w] = size(A);
  L = eye(h);
  U = zeros(h);
  for i = 1:h
    for j = i:h
      U(i, j) = A(i, j) - dot(L(i, 1:i-1), U(1:i-1, j));
    end
    for j = i+1:h
      L(j, i) = (A(j, i) - dot(L(j, 1:i-1), U(1:i-1, i)))/U(i, i);
    end
  end
end
```

Rozkład Cholesky'ego

$$
\begin{aligned}
l_{11} &= \sqrt{a_{11}} = \sqrt{60} & l_{21} &= \frac{a_{21}}{l_{11}} = \frac{30}{\sqrt{60}} = \frac{1}{2}\sqrt{60} \\
l_{31} &= \frac{a_{31}}{l_{11}} = \frac{20}{\sqrt{60}} = \frac{1}{3}\sqrt{60} & l_{22} &= \sqrt{a_{22} - l_{21}^2} = \sqrt{20 - 15} = \sqrt{5} \\
l_{32} &= \frac{a_{32} - l_{31}l_{21}}{l_{22}} = \frac{15 - 10}{\sqrt{5}} = \sqrt{5} \quad l_{33} = \sqrt{a_{33} - l_{31}^2 - l_{32}^2} = \sqrt{12 - \frac{60}{9} - 5} = \frac{\sqrt{3}}{3}
\end{aligned} \tag{73}
$$

Zatem rozkład Cholesky'ego ma postać

$$
\begin{bmatrix} \sqrt{60} & 0 & 0 \\ \frac{\sqrt{60}}{2} & \sqrt{5} & 0 \\ \frac{1}{3}\sqrt{60} & \sqrt{5} & \frac{\sqrt{3}}{3} \end{bmatrix} \cdot \begin{bmatrix} \sqrt{60} & \frac{\sqrt{60}}{2} & \frac{1}{3}\sqrt{60} \\ 0 & \sqrt{5} & \sqrt{5} \\ 0 & 0 & \frac{\sqrt{3}}{3} \end{bmatrix} = \begin{bmatrix} 60 & 30 & 20 \\ 30 & 20 & 15 \\ 20 & 15 & 12 \end{bmatrix} \tag{74}
$$

Przykładowa implementacja w Octave

```
function D = cholesky_dec(A)
  [h, w] = size(A);
  D = zeros(h);
  for i = 1:h
    D(i, i) = sqrt(A(i, i) - sum(D(i, 1:i-1) .^ 2));
    for j = (i + 1):h
      D(j, i) = (A(i, j) - dot(D(i, 1:i-1), D(j, 1:i-1)))/D(i, i);
    end
  end
end
```

2.3 Metoda eliminacji Gaussa

Kolejna metoda przekształcenia macierzy układu równań A to metoda eliminacji Gaussa. Podobnie jak poprzednia metoda sprowadza ona macierz A do postaci iloczynu macierzy dolnie trójkątnej i górnie trójkątnej. W podstawowej wersji metoda ta polega na wyznaczeniu ciągu macierzy $A^{(i)}$, takich że $A^{(1)} = A$ poprzez przekształcenia macierzy będącej poprzednim elementem tego ciągu. k-ty krok tego przekształcenia polega na wykonywaniu działań na wierszach macierzy które prowadzą do wyzerowania wszystkich elementów w kolumnie k, znajdującej się pod elementem na głównej przekątnej macierzy $A^{(k)}$. Przekształcenia te mogą być interpretowane jako odejmowanie od siebie równań układu przemnożonych przez pewne stałe zwane mnożnikami. Uzyskujemy w ten sposób macierz górnie trójkątną $U = A^{(n)}$, podczas gdy macierz dolnie trójkątna L zostaje utworzona z tych mnożników.

Elementy macierzy $A^{(k+1)}$ powstają z elementów macierzy $A^{(k)}$ zgodnie z poniższym wzorem rekurencyjnym.

$$a_{ij}^{(k+1)} = \begin{cases} a_{ij}^{(k)}, & i \leq k \\ a_{ij}^{(k)} - \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}} \cdot a_{kj}^{(k)}, & i \geq k+1 \wedge j \geq k+1 \\ 0, & i \geq k+1 \wedge j \leq k \end{cases} \tag{75}$$

zaś elementy dolnie trójkątnej macierzy L składają się z mnożników wykorzystanych w tworzeniu ciągu macierzy $A^{(k)}$ i wyrażają się następującym wzorem

$$l_{ik} = \begin{cases} \frac{a_{ik}^{(k)}}{a_{kk}(k)}, & i \geq k+1 \\ 1, & i = k \\ 0, & i \leq k-1 \end{cases} \tag{76}$$

Twierdzenie 2.4

Jeśli wszystkie elementy główne $a_{kk}^{(k)}$ obliczone za pomocą powyższych wzorów są różne od zera, to macierz A ma rozkład LU. W szczególności taki rozkład mają macierze:

1) Dodatnio określone
2) O dominującej głównej przekątnej, tzn.

$$\bigwedge_{1 \leq i \leq n} |a_{ii}| \geq \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}| \tag{77}$$

Przykład 2.2

Stosując metodę eliminacji Gaussa rozwiąż układ równań $Ax = b$ dla

$$A = \begin{bmatrix} 4 & -3 & 1 \\ 2 & 2 & -4 \\ 1 & -1 & 1 \end{bmatrix} \quad b = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} \tag{78}$$

Mamy $A = A^{(1)}$. W pierwszym kroku zerujemy wyrazy w pierwszej kolumnie (a_{21} i a_{31}). Wyznaczamy mnożniki do wyzerowania pierwszej kolumny. Są to:

$$l_{21}\frac{a_{21}}{a_{11}} = \frac{2}{4} = \frac{1}{2} \quad l_{31} = \frac{a_{31}}{a_{11}} = \frac{1}{4} \tag{79}$$

a następnie od drugiego wiersza macierzy A odejmujemy pierwszy pomnożony przez l_{21} oraz od trzeciego wiersza macierzy A odejmujemy pierwszy pomnożony przez l_{31}. Otrzymujemy macierz $A^{(2)}$

$$A^{(2)} = \begin{bmatrix} 4 & -3 & 1 \\ 0 & \frac{7}{2} & -\frac{9}{2} \\ 0 & -\frac{1}{4} & \frac{3}{4} \end{bmatrix} \tag{80}$$

W drugim kroku zerujemy wyraz w drugiej kolumnie $(a_{32}^{(2)})$.
Mnożnik do wyzerowania tego wyrazu jest równy

$$l_{32} = \frac{a_{32}^{(2)}}{a_{22}^{(2)}} = \frac{-\frac{1}{4}}{\frac{7}{2}} = -\frac{1}{14} \tag{81}$$

a następnie od trzeciego wiersza macierzy $A^{(2)}$ odejmujemy drugi pomnożony pomnożony przez l_{32}. Otrzymujemy $A^{(3)}$

$$A^{(3)} = \begin{bmatrix} 4 & -3 & 1 \\ 0 & \frac{7}{2} & -\frac{9}{2} \\ 0 & 0 & \frac{3}{7} \end{bmatrix} \tag{82}$$

czyli

$$\begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{2} & 1 & 0 \\ \frac{1}{4} & -\frac{1}{14} & 1 \end{bmatrix} \cdot \begin{bmatrix} 4 & -3 & 1 \\ 0 & \frac{7}{2} & -\frac{9}{2} \\ 0 & 0 & \frac{3}{7} \end{bmatrix} = A \tag{83}$$

Przykładowa implementacja w Octave:

```
function [L, U] = gauss(A)
  [h, w] = size(A);
  L = eye(h);
  for i = 1:h
    L(i+1:h, i) = A(i+1:h, i) / A(i, i);
    for j = i + 1:h
      A(j, :) = A(j, :) - A(i, :) / A(i, i) * A(j, i);
    end
```

```
    end
    U = A;
end
```

Pozostaje rozwiązać "łatwy do rozwiązania" układ równań $LUx = b$.
Czyli oznaczając $Ux = z$ rozwiążemy najpierw układ równań $Lz = b$.
Stosujemy podstawianie w przód:

$$z = \begin{bmatrix} 2 \\ -1 \\ \frac{3}{7} \end{bmatrix} \quad \begin{aligned} z_1 &= b_1 = 2 \\ z_2 &= b_2 - l_{21}z_1 = 0 - \frac{1}{2} \cdot 2 = -1 \\ z_3 &= b_3 - l_{31}z_1 - l_{32}z_2 = 1 - \frac{1}{4} \cdot 2 + \frac{1}{14} \cdot (-1) = \frac{3}{7} \end{aligned} \tag{84}$$

Pozostaje teraz rozwiązać układ równań $Ux = z$, czyli

$$\begin{bmatrix} 4 & -3 & 1 \\ 0 & \frac{7}{2} & -\frac{9}{2} \\ 0 & 0 & \frac{3}{7} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \\ \frac{3}{7} \end{bmatrix} \tag{85}$$

Tym razem stosujemy podstawienie wstecz.

$$\begin{aligned} x_3 &= \frac{z_3}{u_{33}} = \frac{\frac{3}{7}}{\frac{3}{7}} = 1 \\ x_2 &= \frac{z_2 - u_{23}x_3}{u_{22}} = \frac{-1 + \frac{9}{2} \cdot 1}{\frac{7}{2}} = 1 \\ x_1 &= \frac{z_1 - u_{12}x_2 - u_{13}x_3}{u_{11}} = \frac{2 + 3 \cdot 1 - 1 \cdot 1}{1} = 1 \end{aligned} \tag{86}$$

Rozwiązaniem tego układu równań jest wektor

$$x = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{87}$$

Achtung! 3

Czasem łączy się operacje rozkładu macierzy A z rozwiązaniem układu równań $Lz = b$ poprzez dołączenie do macierzy A wektora b i wykonanie na nim tych samych operacji co na wierszach macierzy A. Uzyskujemy wówczas macierz górnie trójkątną U oraz przekształcony wektor b

którego składowe są równe składowym wektora z.

Przykład 2.3

Dla macierzy z poprzedniego przykładu zastosuj ten zmodyfikowany algorytm metody eliminacji Gaussa

$$\begin{bmatrix} 4 & -3 & 1 & 2 \\ 2 & 2 & -4 & 0 \\ 1 & -1 & 1 & 1 \end{bmatrix} \overset{\substack{w_2=w_2-\frac{1}{2}w_1 \\ w_3=w_3-\frac{1}{4}w_1}}{\longrightarrow} \begin{bmatrix} 4 & -3 & 1 & 2 \\ 0 & \frac{7}{2} & -\frac{9}{2} & -1 \\ 0 & -\frac{1}{4} & \frac{3}{7} & \frac{1}{2} \end{bmatrix} \overset{w_3=w_3+\frac{1}{14}w_2}{\longrightarrow} \begin{bmatrix} 4 & -3 & 1 & 2 \\ 0 & \frac{7}{2} & -\frac{9}{2} & -1 \\ 0 & 0 & \frac{3}{7} & \frac{3}{7} \end{bmatrix} \tag{88}$$

Czyli układ równań $Ax = b$ jest równoważny następującemu układowi równań

$$\begin{bmatrix} 4 & -3 & 1 \\ 0 & \frac{7}{2} & -\frac{9}{2} \\ 0 & 0 & \frac{3}{7} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \\ \frac{3}{7} \end{bmatrix} \tag{89}$$

który rozwiązujemy tak jak w poprzednim przykładzie stosując podstawianie wstecz
Przykładowa implementacja w Octave:

```
function [U, z] = gauss_comb(A, b)
  [h, w] = size(A);
  A(:, w+1) = b;
  L = eye(h);
  for i = 1:h
    L(i+1:h, i) = A(i+1:h, i) / A(i, i);
    for j = i + 1:h
      A(j, :) = A(j, :) - A(i, :) / A(i, i) * A(j, i);
    end
  end
  U = A(:, 1:w);
  z = A(:, w+1);
end
```

2025-04-01

Przy rozwiązywaniu układów równań metodą eliminacji Gaussa mogą pojawić się pewne problemy.

Na przykład, jeśli macierz A ma postać

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \tag{90}$$

to jako że w pierwszym kroku element główny $a_{11} = 0$, nie można dokończyć wykonanie tego kroku, ani tym bardziej przejść do kolejnych kroków. Rozważmy też poniższy przykład:

Przykład 2.4

Rozwiąż układ równań

$$\begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \tag{91}$$

gdzie ε jest bardzo małą dodatnią liczbą różną od zera. Mamy wtedy

$$\left[\begin{array}{cc|c} \varepsilon & 1 & 1 \\ 1 & 1 & 2 \end{array}\right] \overset{w_2 = w_2 - \frac{1}{\varepsilon} w_1}{\longrightarrow} \left[\begin{array}{cc|c} \varepsilon & 1 & 1 \\ 0 & 1 - \frac{1}{\varepsilon} & 2 - \frac{1}{\varepsilon} \end{array}\right] \tag{92}$$

Zatem

$$x_2 = \frac{2 - \frac{1}{\varepsilon}}{1 - \frac{1}{\varepsilon}} \tag{93}$$

$$x_1 = \frac{1 - x_2}{\varepsilon} \tag{94}$$

Jako że ε jest bardzo małe, czyli $\frac{1}{\varepsilon}$ jest bardzo duże w arytmetyce fl

$$2 - \frac{1}{\varepsilon} \approx 1 - \frac{1}{\varepsilon} \approx -\frac{1}{\varepsilon} \tag{95}$$

czyli $x_2 \approx 1$, zaś $x_1 \approx 0$, podczas gdy dokładnym rozwiązaniem jest $x_2 = \frac{2\varepsilon - 1}{\varepsilon - 1} \approx 1$ oraz $x_1 = \frac{1 - \frac{2\varepsilon - 1}{\varepsilon - 1}}{\varepsilon} = -\frac{\varepsilon}{\varepsilon(\varepsilon - 1)} = \frac{1}{1 - \varepsilon} \approx 1$
Problemem w tym wypadku jest fakt, że element główny ($a_{11} = \varepsilon$) jest względnie mały w porównaniu z resztą wartości w tym wierszu

Powyższe problemy prowadzą do modyfikacji podstawowej metody eliminacji Gaussa do metody eliminacji Gaussa ze skalowanym wyborem wierszy głównych. Polega ona na zmianie kolejności równań w układzie równań tak, aby element główny w danym kroku był możliwie największy. Ponieważ wymiana wierszy w macierzy jest z programistycznego punktu widzenia kosztowna, zwykle w praktycznej reali-

zacji tego algorytmu przechowuje się w wektorze permutacji p informacje w jakiej kolejności zostały wybrane główne wierszy podczas eliminacji Gaussa.

2.4 Metoda eliminacji Gaussa ze skalowanym wyborem wierszy głównych

1. Tworzymy wektor permutacji $p = [1, 2, 3, \ldots, n]$.
2. Tworzymy wektor skal $s = \{s_i = \max_{1 \leq j \leq n} |a_{ij}| : i = 1, 2, \ldots, n\}$
3. Wykonujemy $n - 1$ kroków eliminacji Gaussa (każdy dla eliminacji kolejnej kolumny), składających się się z
   (i) Wyboru indeksu kolejnego wiersza głównego, takiego, że
   $$\frac{|a_{p_j k}|}{s_{p_j}} \geq \frac{|a_{p_i k}|}{s_{p_i}} \quad i = k, k+1, \ldots, n \qquad (96)$$
   gdzie k jest numerem kroku(albo indeksem eliminowanej kolumny)
   (ii) Aktualizacji wektora permutacji $p_k \leftrightarrow p_j$
   (iii) Wyznaczeniu mnożnika $\frac{a_{p_i k}}{a_{p_k k}}$ do eliminacji p_i-go wiersza, $i = k+1, k+2, \ldots, n$
   (iv) Eliminacji p_i-tego wiersza $i = k+1, k+2, \ldots, n$ wraz z odpowiadającym mu wyrazem wolnym b_{p_i}
4. Rozwiązujemy otrzymany układ równań z macierzą górnie trójkątną U i zmodyfikowanym wektorem b(będziemy go oznaczali b_*)

Rozwiązanie układu równań $Ux = b_*$ przebiega podobnie jak poprzednio, pamiętamy jednak, o zmianie kolejności wierszy i elementów wektora b_* zgodnie z wektorem permutacji p.

Przykład 2.5

Stosując metodę eliminacji Gaussa ze skalowanym wyborem wierszy głównych rozwiąż układ równań:

$$\begin{bmatrix} 2 & 3 & -6 \\ 1 & -6 & 8 \\ 3 & -2 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \qquad (97)$$

Początkowo wektor permutacji $p = [1, 2, 3]$, a wektor skal $s = [6, 8, 3]$
Sprawdzamy ilorazy wartości bezwzględnych elementów w pierwszej kolumnie przez skalę danego wiersza i wybieramy wiersz któremu odpowiada największa z tych wartości:

$$\frac{|a_{p_1 1}|}{s_{p_1}} = \frac{2}{6} = \frac{1}{3} \qquad \frac{|a_{p_2 1}|}{s_{p_2}} = \frac{1}{8} \qquad \frac{|a_{p_3 1}|}{s_{p_3}} = \frac{3}{3} = 1 \tag{98}$$

Zatem do pierwszego kroku jako wiersz główny wybieramy wiersz 3.
Aktualizujemy wektor permutacji $p = [3, 2, 1]$ i otrzymujemy

$$\left[\begin{array}{ccc|c} 2 & 3 & -6 & 1 \\ 1 & -6 & 8 & 1 \\ 3 & -2 & 1 & 1 \end{array}\right] \overset{\substack{w_1 = w_1 - \frac{2}{3} w_3 \\ w_2 = w_2 - \frac{1}{3} w_3}}{\longrightarrow} \left[\begin{array}{ccc|c} 0 & \frac{13}{3} & -\frac{20}{3} & \frac{1}{3} \\ 0 & -\frac{16}{3} & \frac{23}{3} & \frac{2}{3} \\ 3 & -2 & 1 & 1 \end{array}\right] \tag{99}$$

W kolejnym kroku sprawdzamy ilorazy wartości bezwzględnych elementów w drugiej kolumnie (wierszy wyznaczonych przez p_2 i p_3 – czyli wierszy 2 i 1) przez skalę danego wiersza

$$\frac{|a_{p_2 2}|}{s_{p_2}} = \frac{\frac{16}{3}}{8} = \frac{2}{3} \qquad \frac{|a_{p_3 2}|}{s_{p_3}} = \frac{\frac{13}{3}}{6} = \frac{13}{18} \tag{100}$$

Zatem kolejnym wierszem głównym jest wiersz wyznaczony przez p_3, czyli wiersz pierwszy.
Aktualizujemy wektor permutacji wymieniając p_2 z p_3 i otrzymujemy $p = [3, 1, 2]$ i dostajemy:

$$\left[\begin{array}{ccc|c} 0 & \frac{13}{3} & -\frac{20}{3} & \frac{1}{3} \\ 0 & -\frac{16}{3} & \frac{23}{3} & \frac{2}{3} \\ 3 & -2 & 1 & 1 \end{array}\right] \overset{w_2 = w_2 + \frac{16}{13} w_1}{\longrightarrow} \left[\begin{array}{ccc|c} 0 & \frac{13}{3} & -\frac{20}{3} & \frac{1}{3} \\ 0 & 0 & -\frac{7}{13} & \frac{14}{13} \\ 3 & -2 & 1 & 1 \end{array}\right] \tag{101}$$

Czyli macierz U i wektor b_* mają postać

$$\begin{bmatrix} 0 & \frac{13}{3} & -\frac{20}{3} \\ 0 & 0 & -\frac{7}{13} \\ 3 & -2 & 1 \end{bmatrix} \qquad b_* = \begin{bmatrix} \frac{1}{3} \\ \frac{14}{3} \\ 1 \end{bmatrix} \tag{102}$$

Rozwiązujemy teraz układ równań $Ux = b_*$.
Stosujemy podstawianie wstecz zgodnie z wektorem p (czyli w kolejności równań 2, 1, 3)
Rozwiązaniem są więc:

$$\begin{gathered} x_3 = \frac{b_{*p_3}}{U_{p_3 3}} = \frac{\frac{14}{13}}{-\frac{7}{13}} = -2 \qquad x_2 = \frac{b_{*p_2} - u_{p_2 3} \cdot x_3}{u_{p_2 2}} = \frac{\frac{1}{3} + \frac{20}{3} \cdot (-2)}{\frac{13}{3}} = -3 \\ x_1 = \frac{b_{*p_1} - u_{p_1 2} x_2 - u_{p_1 3} x_3}{u_{p_1 1}} = \frac{1 - (-2)(-3) - 1(-2)}{3} = -1 \end{gathered} \tag{103}$$

Przykładowa implementacja w Octave:

```
function x = gauss_perm(A, b)
  [h, w] = size(A);
  x = zeros(h, 1);

  % Wektor permutacji
  p = 1:h;

  % Wektor skal
  s = max(abs(A'));

  % Tworzymy macierz rozszerzoną
  A(:, h+1) = b;

  for j = 1:h-1
    % Wybór kolejnego wiersza głównego
    [_, i] = max(abs(A(p(j:h), j))' ./ s(p(j:h)));

    % Musimy uwzględnić to, że `i` był wybrany z wektora długości
    % h - j + 1
    i = i + j - 1;

    % Aktualizacja wektora permutacji
    p([j, i]) = p([i, j]);

    % Eliminacja pozostałych wierszy
    for k = j+1:h
      A(p(k), :) = A(p(k), :) - A(p(j), :) * A(p(k), j) / A(p(j), j);
    end
  end

  % Rozwiązanie układu podstawianiem wstecz
  for i = h:-1:1
    x(i) = ( A(p(i), h + 1) -
             dot(A(p(i), i+1:h), x(i+1:h))
           ) / A(p(i), i);
  end
end
```

Można oszacować liczbę działań niezbędnych do przeprowadzenia metody eliminacji Gaussa ze skalowanym wyborem wierszy głównych. Aby wyznaczyć współczynniki $\frac{|a_{p_j k}|}{p_j}$ dla każdego spośród n wierszy, niezbędne do wyboru wierszy głównych, należy wykonać n dzieleń.

W pierwszym kroku eliminacji do każdego wiersza należy wyznaczyć mnożnik $\frac{a_{p_i k}}{a_{p_k k}}$ co wymaga jednego dzielenia, oraz $n-1$ mnożeń i tyle samo odejmowań. Eliminacja jednego wiersza w pierwszym kroku wymaga $2(n-1)+1 = 2n-1$ działań. Przemnażając to przez liczbę wierszy i dodając operacje niezbędne do wyboru wierszy głównych otrzymujemy liczbę operacji w pojedynczym kroku metody:

$$(2n-1)\cdot(n-1)+n = 2n^2-3n+1+n = 2n^2-2n+1 \qquad (104)$$

Ponieważ w każdym kolejnym kroku wykonujemy te same operacje tylko dla zmniejszonego o 1 rozmiaru macierzy, zatem łączna liczba operacji jest równa

$$\sum_{i=1}^{n}(2i^2-2i+1) = 2\cdot\frac{n(n+1)(2n+1)}{6} - 2\cdot\frac{n(n+1)}{2} + n = \frac{2}{3}n^3+\frac{n}{3} \qquad (105)$$

Twierdzenie 2.5

Wszystkie przedstawione algorytmy rozwiązywania układów równań liniowych są numerycznie poprawne. Wskaźnik uwarunkowania zadania rozwiązywania układu równań postaci

$$Ax = b \qquad (106)$$

zależy od macierzy A i jest równy

$$\kappa(A) = \|A\|\cdot\left\|A^{-1}\right\| \qquad (107)$$

2025-04-08

2.5 Inne modyfikacje metody eliminacji Gaussa

Powyżej przedstawione algorytmy rozwiązywania układów równań liniowych można uzupełnić o dodatkowe modyfikacje poprawiające działanie tych algorytmów. Przykładowymi modyfikacjami są:

2.5.1 Wstępne wyważanie wierszy

Polega na wstępnym podzieleniu każdego z wierszy (i odpowiadających im wyrazów wolnych) przez wartość bezwzględną (co do modułu) elementu wiersza $r_i = \frac{1}{\max_{1\le j\le n}|a_{ij}|}$.

Prowadzi to do zamiany układu równań

$$\sum_{j=1}^{n} a_{ij}x_j = b_i \qquad \text{na układ} \qquad \sum_{j=1}^{n} (r_i a_{ij})x_j = b_i r_i \tag{108}$$

Przykładowa implementacja w Octave:

```
function x = gauss_scaled_rows(A, b)
  [n, m] = size(A);
  r = max(abs(A'));
  A = A ./ r';
  b = b ./ r';
  L = eye(n);
  for k = 1:n-1
    for i = k+1:n
      L(i, k) = A(i, k) / A(k, k);
      A(i, :) = A(i, :) - L(i, k) * A(k, :);
    end
  end
  z = rozw_l(L, b);
  x = rozw_u(A, z);
end
```

2.5.2 Wstępne wyważanie kolumn

Polega na wstępnym podzieleniu każdej z kolumn przez wartość bezwzględną z maksymalnego (co do modułu) elementu w kolumnie $c_j = \dfrac{1}{\max_{1 \le i \le n} |a_{ij}|}$

Prowadzi to do zamiany układu równań

$$\sum_{j=1}^{n} a_{ij}x_j = b_i \qquad \text{na układ} \qquad \sum_{j=1}^{n} (c_j a_{ij})\frac{x_j}{c_j} = b_i \tag{109}$$

Przyjmując wtedy $y_j = \frac{x_j}{c_j}$ rozwiązujemy ten układ równań względem wektora y i wyznaczamy x z powyższej zależności

Przykładowa implementacja w Octave:

```
function x = gauss_scaled_cols(A, b)
  [n, m] = size(A);
  r = max(abs(A));
  A = A ./ r;
  L = eye(n);
```

```
  for k = 1: n -1
    for i = k+1:n
      L(i, k) = A(i, k) / A(k, k);
      A(i, :) = A(i, :) - L(i, k) * A(k, :);
    end
  end
  z = rozw_l(L, b);
  y = rozw_u(A, z);
  x = y ./ r';
end
```

2.5.3 Pełny wybór elementów głównych

W kolejnych krokach wybieramy elementy główne nie tylko spośród elementów w aktualnie zerowanej kolumnie k-tej, ale spośród wszystkich pozostałych $(n-k+1)^2$ elementów. Zastosowanie tej metody wymaga wprowadzenia dodatkowego wektora permutacji kolumn, ale w praktyce nie działa istotnie lepiej od algorytmu eliminacji Gaussa ograniczającego się do wyboru wierszy głównych

2.5.4 Wyważanie lub skalowanie w każdym kroku eliminacji

W każdym kroku działania metody wyznaczamy nowy wektor skal s lub w każdym kroku dokonujemy wyważanie wierszy albo kolumn.

2.5.5 Poprawianie iteracyjne końcowego rozwiązania

Przykład 2.6

$$A_{3\times 3}\cdot x = b \qquad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{110}$$

Dokładne rozwiązanie to:

$$\begin{aligned} a_{11}(x_1+\varepsilon_1)+a_{12}(x_2+\varepsilon_2)+a_{13}(x_3+\varepsilon_3) &= b_1 \\ a_{21}(x_1+\varepsilon_1)+a_{22}(x_2+\varepsilon_2)+a_{23}(x_3+\varepsilon_3) &= b_2 \\ a_{31}(x_1+\varepsilon_1)+a_{32}(x_2+\varepsilon_2)+a_{33}(x_3+\varepsilon_3) &= b_3 \end{aligned} \tag{111}$$

Gdzie

$$\varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} \qquad (112)$$

$$Ax + A\varepsilon = b \iff A\varepsilon = \underbrace{(b - Ax)}_{b^*} \qquad (113)$$

W metodzie poprawiania iteracyjnego zakładamy iż jako rozwiązanie układu równań $Ax = b$ otrzymujemy jedynie jego przybliżony wektor rozwiązań $x^{(0)}$ podczas gdy wartość dokładna wektora rozwiązań x wyraża się wzorem

$$x = x^{(0)} + A^{-1}(b - Ax^{(0)}) = x^{(0)} + e^{(0)} \qquad (114)$$

gdzie różnicę $r^{(0)} = b - Ax^{(0)}$ nazywamy wektorem rezydualnym, a wektor błędu $e^{(0)}$ możemy wyznaczyć rozwiązując układ równań

$$Ae^{(0)} = r^{(0)} \qquad (115)$$

W ten sposób możemy określić kolejne przybliżenia rozwiązań $x^{(i+1)}$ za pomocą następujących wzorów rekurencyjnych:

$$\begin{aligned} r^{(i)} &= b - Ax^{(i)}, \\ Ae^{(i)} &= r^{(i)}, \\ x^{(i+1)} &= x^{(i)} + e^{(i)} \end{aligned} \qquad (116)$$

Uzyskujemy ciąg wektorów przybliżonych rozwiązań tym szybciej zbieżny do wektora dokładnego x im dokładniej jest rozwiązany zarówno podstawowy układ równań $Ax^{(i)} = b$ jak i układy równań $Ae^{(i)} = r^{(i)}$.

Przykładowa implementacja «poprawionej» metody Gaussa:

```
function x = rozw_gauss_popraw(A, b, pop = 0)
  [h, w] = size(A);
  L = eye(h);
  U = A;
  for i = 1:h
    L(i+1:h, i) = A(i+1:h, i) / A(i, i);
    U(i+1:h, :) = A(i+1:h, :) - A(i, :) .* A(i+1:h, i) / A(i, i);
```

```
  end
  z = rozw_l(L, b);
  x = rozw_u(U, z);
  for i = 1:pop
    r = b - A * x;
    ee = rozw_l(L, r);
    e = rozw_u(U, ee);
    x = x + e;
  end
end
```

Przykładowa implementacja «poprawionej» metody Doolittle'a w Octave:

```
function x = rozw_doolittle_popraw(A, b, pop=0)
  [h, w] = size(A);
  L = eye(h);
  U = zeros(h);
  for i = 1:h
    U(i, i) = A(i, i) - L(i, 1:i-1) * U(1:i-1, i);
    for j = i+1:h
      U(i, j) = A(i, j) - L(i, 1:i-1) * U(1:i-1, j);
      L(j, i) = (A(j, i) - L(j, 1:i-1) * U(1:i-1, i))/U(i, i);
    end
  end

  z = rozw_l(L, b);
  x = rozw_u(U, z);
  for i = 1:pop
    r = b - A * x;
    ee = rozw_l(L, r);
    e = rozw_u(U, ee);
    x = x + e;
  end
end
```

2.6 Metody iteracyjne rozwiązywania układów równań liniowych

Podane dotychczas metody rozwiązywania układów równań liniowych są metodami dokładnymi. Po ich zastosowaniu otrzymujemy rozwiązanie które byłoby dokładne gdyby nie błędy wynikające z przeprowadzania działań w arytmetyce zmiennopozycyjnej fl, oraz błędy reprezentacji liczb w pamięci komputera. W metodach iteracyjnych konstruuje się natomiast ciąg wektorów rozwiązań $x^{(i)}$, w taki sposób, aby ciąg ten był zbieżny do rozwiązania dokładnego x. Zaletą zastosowania metod iteracyjnych jest ich szybkość działania (z reguły wymagają mniejszej liczby operacji), oraz mniejsze

wymagania pamięciowe, jednakże odbywa się to kosztem mniejszej dokładności uzyskanego rozwiązania. Przekształcamy w następującej sposób rozwiązujemy układ równań liniowych w postaci macierzowej

$$\begin{aligned} Ax &= b \quad \Big| + Qx - Ax \\ Qx &= (Q-A)x + b \end{aligned} \tag{117}$$

gdzie Q jest pewną nieosobliwą macierzą. Otrzymane równanie jest równoważne wyjściowemu układowi równań i ma ten sam zbiór rozwiązań. Powyższe równanie można traktować jako wzór rekurencyjny służący do wyznaczania kolejnych przybliżeń rozwiązania x, zapisując je w postaci

$$Qx^{(k)} = (Q-A)x^{(k-1)} + b \tag{118}$$

Jeśli ciąg $x^{(k)}$ jest zbieżny do x, to spełnia on powyższe równanie o czym łatwo się przekonać przechodząc w nim do granicy przy $k \to \infty$.

Dobierając różne postaci macierzy Q uzyskujemy różne metody iteracyjne. Aby dana metoda była wygodna w zastosowaniach macierz Q powinna być dobrana w taki sposób, aby wyznaczanie kolejnych wartości $x^{(k)}$ było łatwe (aby łatwo było wyznaczyć macierz Q^{-1}) oraz aby wyznaczony w ten sposób ciąg rozwiązań $x^{(k))}$ był możliwie szybko zbieżny do rozwiązania dokładnego. Mnożąc lewostronnie równanie (118) otrzymujemy:

$$\begin{aligned} Q^{-1}Qx^{(k)} &= Q^{-1}(Q-A)x^{(k-1)} + Q^{-1}b \\ x^{(k)} &= (I - Q^{-1}A)x^{(k-1)} + Q^{-1}b \end{aligned} \tag{119}$$

Jeśli $\|I - Q^{-1}A\| < 1$, to $x^{(k)}$ zdefiniowany jak powyżej jest zbieżny do rozwiązania układu równań $Ax = b$, bez względu na sposób doboru pierwszego przybliżenia rozwiązania $x^{(0)}$

2.6.1 Metoda Richardsona

Macierz Q jest w niej macierzą jednostkową I, czyli

$$x^{(k)} = (I-A)x^{(k-1)} + b = x^{(k-1)} + (b - Ax^{(k-1)}) = x^{(k-1)} + r^{(k-1)} \tag{120}$$

gdzie $r^{(k-1)} = b - Ax^{k-1}$ jest wektorem rezydualnym.

Metoda ta jest zbieżna, gdy $\|I - A\| < 1$. W metodzie tej w k-tym kroku wyznaczamy wektory $x^{(k)}$ i $r^{(k)}$ za pomocą wzorów

$$\begin{aligned} r_i^{(k)} &= b_i - \sum_{j=1}^{n} a_{ij} x_j^{(k)} \\ x_i^{(k)} &= x_i^{(k-1)} + r_i^{(k)} \end{aligned} \tag{121}$$

Przykładowa implementacja w Octave:

```
function x = iter_rozw_rich(A, b, x0, k = 10)
  [h, w] = size(A)
  x = x0;
  for i = 1:k
    r = b - A * x
    x = x + r;
  end
end
```

2.6.2 Metoda Jacobiego

Macierz Q jest w niej macierzą przekątniową, a elementy na jej głównej przekątnej są równe elementom głównej przekątnej macierzy A

W metodzie tej wyznaczamy i-tą składową kolejnego przybliżenia rozwiązania z i-tego równania, w który w miejsce pozostałych składowych wstawione zostały ich wartości z poprzedniego przybliżenia rozwiązania. Wzór służący wyznaczaniu składowych kolejnych przybliżeń ma zatem postać

$$x_i^{(k)} = \frac{b_i - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij} x_j}{a_{ii}} \tag{122}$$

2025-04-15

Twierdzenie 2.6

Jeśli macierz A jest dominująca przekątniowo (tzn. o dominującej głównej przekątnej), to ciąg $x^{(k)}$ zdefiniowany tak jak w metodzie Jacobiego jest zbieżny do rozwiązania układu równań $Ax = b$ bez względu na sposób doboru pierwszego przybliżenia rozwiązania $x^{(0)}$.

Achtung! 4

Ponieważ w każdym kroku tej metody następuje dzielenie wektora b oraz elementu macierzy A w i-tym wierszu przez element z głównej przekątnej (a_{ii}), dlatego w praktycznym zastosowaniu tej metody wygodnie jest przed przejściem do kolejnych kroków wyznaczania przybliżenia rozwiązania, podzielić wiersze macierzy A, oraz odpowiadające im składowe wektora b przez odpowiednie wyrazy a_{ii} z głównej przekątnej macierzy A

2.6.3 Metoda Gaussa-Seidela

Macierz Q jest w niej macierzą dolnie trójkątną, a jej elementy są równe elementom macierzy A poniżej głównej przekątnej (wraz z tą przekątną), czyli $a_{ij}\ i \geq j$

Metoda ta działa podobnie jak metoda Jacobiego. W każdym kroku metody wyznaczamy i-tą składową kolejnego przybliżenia rozwiązania z i-tego równania w którym w miejsce składowych o indeksach od $i+1$ do n wstawione są ich wartości z poprzedniego przybliżenia rozwiązania, a w miejsce składowych o indeksach od 1 do $i-1$ wstawione są ich wartości z obecnie wyznaczanego przybliżenia rozwiązania (w momencie wyznaczania i-tej składowej wcześniejsze składowe są już wyznaczone). Wzór służący wyznaczaniu składowych kolejnych przybliżeń ma zatem postać:

$$x_i^{(k)} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)}}{a_{ii}} \qquad (123)$$

Twierdzenie 2.7

Jeśli macierz A jest dominująca przekątniowo, to ciąg $x^{(k)}$ zdefiniowany tak jak w metodzie Gaussa-Seidela, jest zbieżny do rozwiązania układu równań $Ax = b$, bez względu na sposób doboru pierwszego przybliżenia rozwiązania $x^{(0)}$.

Przykładowa implementacja w Octave:

```
function x = iter_rozw_gauss_seidel(A, b, x0, k=10)
  [h, w] = size(A);
  x = x0;
  for i = 1:k
    for j = 1:h
      % Zamiast odejmować dwie sumy, odejmujemy jedną i dodajemy
      % składnik A(j, j) * x(j), którego nie powinniśmy byli odejmować
      %
      %                  Suma       składnik kompensujący
```

```
      %               vvvvvvvvvvvv   vvvvvvvvvvvvvv
      x(j) = (b(j) - A(j,1:h) * x + A(j, j) * x(j)) / A(j, j);
    end
  end
end
```

W metodach iteracyjnych można poprawić ich zbieżność poprzez zastosowanie dodatkowych technik polegających na ustalaniu wartości nowego przybliżenia rozwiązania poprzez wykorzystanie także poprzednich rozwiązań:

1) Ekstrapolacja – dla ustalonej wartości parametru $0 \leq \gamma \leq 1$ w k-tym kroku wyznaczamy najpierw k-tą wartość wektora przybliżeń rozwiązania $x^{(k)}$ zgodnie z jedną z podanych metod, a następnie ustalamy ostateczną wartość k-tego przybliżenia jako

$$x^{(k)} = \gamma x^{(k)} + (1-\gamma)x^{(k-1)} \qquad (124)$$

2) Metoda Czebyszewa – dla ustalonych wartości parametru $0 \leq \gamma \leq 1$ i ciągu $0 \leq \varrho_k \leq 1$ dla $k \geq 2$ w k-tym kroku wyznaczamy najpierw k-tą wartość wektora przybliżeń rozwiązań $x^{(k)}$ zgodnie z jedną z podanych metod a następnie ustalamy ostateczną wartość k-tego przybliżenia jako

$$x^{(k)} = \varrho_k(\gamma x^{(k)} + (1-\gamma)x^{(k-1)}) + (1-\varrho_k)x^{(k-2)} \qquad (125)$$

2.6.4 Metody gradientowe

Nieco inną grupą metod iteracyjnych są metody gradientowe. W metodach tych kolejne przybliżenie rozwiązania wyraża się wzorem

$$x^{(k)} = x^{(k-1)} + t_{k-1}v^{(k)} \qquad (126)$$

gdzie

- ciąg $v^{(k)}$ jest pewnym ciągiem wektorów kierunkowych
- t_k są pewnymi liczbami rzeczywistymi

Metoda najszybszego spadku Przykładem metody gradientowej jest metoda najszybszego spadku. W metodzie rolę ciągu wektorów $v^{(k)}$ pełni ciąg wektorów rezydualnych a współczynniki t_k wyznaczone są ze wzoru

$$t_k = \frac{v^{(k)} \circ v^{(k)}}{v^{(k)} \circ (Av^{(k)})} \qquad (127)$$

gdzie $\circ$ oznacza iloczyn skalarny. Poszczególne ciągi w tej metodzie wyrażają się zatem wzorami:

$$\begin{aligned} v_i^{(k)} &= b_i - \sum_{j=1}^{n} a_{ij} x_j^{(k-1)} \\ t_k &= \frac{\sum_{i=1}^{n} (v_i^{(k)})^2}{\sum_{i=1}^{n} v_i^{(k)} \cdot (\sum_{j=1}^{n} a_{ij} v_i^{(k)})} \\ x_i^{(k)} &= x_i^{(k-1)} + t_k \cdot v_i^{(k)} \end{aligned} \qquad (128)$$

Metody gradientowe można stosować jedynie do macierzy rzeczywistych symetrycznych i dodatnio określonych

Przykład 2.7

Wykonaj po 2 kroki iteracyjnych metod Jacobiego i Gaussa-Seidela dla układu równań

$$\begin{bmatrix} 2 & -1 & 0 \\ 1 & 6 & -2 \\ 4 & -3 & 8 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ -4 \\ 5 \end{bmatrix} \qquad (129)$$

rozpoczynając od wektora $x^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$

Wyznaczając z kolejnych równań poszczególne składowe wektora $x^{(k)}$ mamy

Metoda Jacobiego	Metoda Gaussa-Seidela
$x_1^{(k)} = \frac{2+x_2^{(k-1)}}{2} = 1 + \frac{x_2^{(k-1)}}{2}$	$x_1^{(k)} = 1 + \frac{x_2^{(k-1)}}{2}$
$x_2^{(k)} = -\frac{1}{6}x_1^{(k-1)} + \frac{1}{3}x_3^{k-1} - \frac{2}{3}$	$x_2^{(k)} = -\frac{1}{6}x_1^{(k)} + \frac{1}{3}x_3^{(k-1)} - \frac{2}{3}$
$x_3^{(k)} = -\frac{1}{2}x_1^{(k-1)} + \frac{3}{8}x_2^{(k-1)} + \frac{5}{8}$	$x_3^{(k)} = -\frac{1}{2}x_1^{(k)} + \frac{3}{8}x_2^{(k)} + \frac{5}{8}$

Dla metody Jacobiego mamy zatem:

$$\begin{aligned} x_1^{(1)} &= 1 \\ x_2^{(1)} &= -\frac{2}{3} \\ x_3^{(1)} &= \frac{5}{8} \end{aligned} \qquad x^{(1)} = \begin{bmatrix} 1 \\ -\frac{2}{3} \\ \frac{5}{8} \end{bmatrix} \tag{130}$$

$$\begin{aligned} x_1^{(2)} &= 1 + \frac{1}{2} \cdot (-\frac{2}{3}) = \frac{2}{3} \\ x_2^{(2)} &= -\frac{1}{6} \cdot 1 + \frac{1}{3} \cdot \frac{5}{8} - \frac{2}{3} = -\frac{15}{24} \\ x_3^{(2)} &= -\frac{1}{2} \cdot 1 + \frac{3}{8} \cdot (-\frac{2}{3}) + \frac{5}{8} = -\frac{1}{8} \end{aligned} \qquad x^{(2)} = \begin{bmatrix} \frac{2}{3} \\ -\frac{15}{24} \\ -\frac{1}{8} \end{bmatrix} \tag{131}$$

Dla metody Gaussa-Seidela:

$$\begin{aligned} x_1^{(1)} &= 1 - \frac{1}{2} \cdot 0 = 1 \\ x_2^{(1)} &= -\frac{1}{6} \cdot 1 + \frac{1}{3} \cdot 0 - \frac{2}{3} = -\frac{5}{6} \\ x_3^{(1)} &= -\frac{1}{2} \cdot 1 + \frac{3}{8}(-\frac{5}{6}) + \frac{5}{8} = -\frac{3}{16} \end{aligned} \qquad x^{(1)} = \begin{bmatrix} 1 \\ -\frac{5}{6} \\ -\frac{3}{16} \end{bmatrix} \tag{132}$$

$$\begin{aligned} x_1^{(2)} &= 1 + \frac{1}{2}(-\frac{5}{6}) = \frac{7}{12} \\ x_2^{(2)} &= -\frac{1}{6} \cdot \frac{7}{12} + \frac{1}{3}(-\frac{3}{16}) - \frac{2}{3} = -\frac{119}{144} \\ x_3^{(2)} &= -\frac{1}{2} \cdot \frac{7}{12} + \frac{3}{8}(-\frac{119}{144}) + \frac{5}{8} = \frac{27}{1152} \end{aligned} \qquad x^{(2)} = \begin{bmatrix} \frac{7}{12} \\ -\frac{119}{144} \\ \frac{27}{1152} \end{bmatrix} \tag{133}$$

3 Rozwiązywanie równań nieliniowych

Niech $g(x)$ i $h(x)$ będą dwoma funkcjami rzeczywistymi zmiennej $x \in \mathbb{R}$, czyli $g, h : \mathbb{R} \to \mathbb{R}$

Równanie $g(x) = h(x)$ nazywamy nieliniowym jeśli funkcja $f(x) = g(x) - h(x)$ nie jest funkcją liniową. Przynosząc funkcję $h(x)$ na lewą stronę powyższego równania otrzymujemy równoważne mu równanie postaci

$$f(x) = g(x) - h(x) = 0 \tag{134}$$

Zatem aby znaleźć rozwiązanie równania $g(x) = h(x)$ wystarczy odnaleźć miejsce zerowe funkcji $f(x) = g(x) - h(x)$. Problem rozwiązywania równania, jest zatem równoważny problemowi znajdowania miejsca zerowego funkcji.

3.1 Metody znajdowania miejsca zerowego

3.1.1 Metoda bisekcji

Metodę bisekcji można stosować do funkcji ciągłej i takiej, że na końcach badanego przedziału przyjmuje ona wartości przeciwnych znaków. Dla takiej funkcji można zastosować własność Darboux.

Twierdzenie 3.1: Własność Darboux

Funkcja ciągła f w przedziale $[a, b]$ przyjmuje w tym przedziale wszystkie wartości pomiędzy $f(a)$ i $f(b)$. W szczególności jeśli $f(a) \cdot f(b) < 0$, to w przedziale (a, b) znajduje się miejsce zerowe tej funkcji.
Zatem warunek $f(a) \cdot f(b) < 0$ gwarantuje istnienie wewnątrz przedziału $[a, b]$ przynajmniej jednego miejsca zerowego

---

2025-04-29

W metodzie tej dla ustalonych końców przedziału $a = a_0$ i $b = b_0$ w których rozpatrywana funkcja $f(x)$ przyjmuje wartości przeciwnych znaków, znajdujemy środek tego przedziału (punkt $c = \frac{a+b}{2}$) i wyznaczamy dla niego wartość funkcji $f(c)$.

Jeśli wartość funkcji w tym punkcie ma taki sam znak jak $f(a)$, to jako nowy początek przeszukiwanego przedziału przyjmujemy punkt $c = a_1$ (koniec tego przedziału pozostaje niezmieniony $b_1 = b_0$). W przeciwnym razie początek przedziału pozostawiamy niezmieniony ($a_1 = a_0$), zaś jako nowy koniec przeszukiwanego przedziału przyjmujemy punkt $c = b_1$. Postępowanie to kontynuujemy rekurencyjnie aż do osiągnięcia zadowalającego przybliżenia miejsca zerowego.

Ponieważ działania te wykonywane są w arytmetyce zmiennopozycyjnej, jako kryterium zatrzymania algorytmu, należy poza maksymalną liczbą kroków algorytmu M przyjąć także parametry β i ε określające odpowiednio zadowalającą długość przedziału w którym znajduje się miejsce zerowe (odległość faktycznego miejsca zerowego od uzyskanego przybliżenia na osi OX), oraz minimalną wartość bezwzględną wartości funkcji f. Algorytm należy zakończyć przy spełnieniu co najmniej jednego z powyższych warunków.

W metodzie bisekcji w każdym kroku otrzymujemy przedział dwa razy krótszy niż w poprzednim kroku:

$$\begin{aligned} b_n - a_n &= \frac{1}{2}(b_{n-1} - a_{n-1}) \\ &= \frac{1}{2^n}(b_0 - a_0) \end{aligned} \tag{135}$$

Ponieważ długości przedziałów w kolejnych krokach dążą do 0, ciąg lewych końców przedziałów a_i jest niemalejący i ograniczony od góry, a ciąg prawych końców przedziałów b_i jest nierosnący i ograniczony od dołu. Zatem oba te ciągi mają granicę r. Przechodząc zatem do granicy $n \to \infty$ w nierówności $0 > f(a_n)f(b_n)$ otrzymujemy $0 \geq f(r)^2$, czyli $f(r) = 0$.

Mimo, że metoda bisekcji jest skuteczna w znajdowaniu miejsca zerowego funkcji, to jest ona stosunkowo wolno zbieżna. Jej zaletą jest jednak, że zawsze prowadzi do odnalezienia przybliżenia miejsca zerowego niezależnie od początkowego doboru przedziału $[a, b]$, jeśli tylko $f(a)f(b) < 0$

3.1.2 Metoda Newtona (stycznych)

Znacznie szybszą (choć nie zawsze zbieżną) od metody bisekcji jest metoda Newtona.

Metoda ta jest szczególnie szybko zbieżna w bliskim otoczeniu dokładnego miejsca zerowego, dlatego wygodna jest do wykonania końcowych etapów wyznaczania miejsca zerowego w połączeniu z jakąś wolniejszą ale zawsze zbieżną metodą (np. metodą bisekcji, przy pomocy której wykonujemy początkowe kroki)

Niech r będzie dokładnym miejscem zerowym, a $x = r - h$ jego przybliżeniem. Ze wzoru Taylora mamy wówczas:

$$0 = f(r) = f(x + h) = f(x) + \frac{f'(x)}{1!}h + O(h^2) \qquad (136)$$

dla dostatecznie małego h mamy:

$$0 \approx f(x) + f'(x) \cdot h \qquad (137)$$

czyli

$$h \approx -\frac{f(x)}{f'(x)} \qquad (138)$$

Zatem lepszym przybliżeniem miejsca zerowego r jest wartość

$$r \approx x - \frac{f(x)}{f'(x)} \qquad (139)$$

Otrzymujemy stąd rekurencyjny wzór dla metody Newtona:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \qquad (140)$$

Metoda Newtona polega na przybliżeniu funkcji f za pomocą funkcji liniowej o tej samej wartości pochodnej w aktualnie rozważanym punkcie. Zatem graficznie metodę tę można zinterpretować jako wyznaczenie stycznej do wykresu funkcji f w rozważanym punkcie, a następnie wyznaczenie punktu przecięcia tej stycznej z osią OX, który stanowi kolejne przybliżenie miejsca zerowego.

Metoda Newtona jest kwadratowo zbieżna do dokładnej wartości miejsca zerowego, jeśli r jest dokładną wartością miejsca zerowego, to

$$|x_{n+1} - r| \leq \left|C(x_n - r)^2\right| \qquad (141)$$

gdzie C jest pewną stałą

Zastosowanie uogólnionej metody Newtona do rozwiązywania układów równań nieliniowych
Metodę Newtona można uogólnić tak, aby mogla służyć do rozwiązywania układów równań nieliniowych. Oznaczmy przez $f\colon (x_1, x_2, \dots x_k)$ kolejne funkcje (równania) których miejsca zerowego poszukujemy. Stosując do pojedynczej funkcji wzór Taylora dla funkcji wielu zmiennych otrzymujemy:

$$0 = f_i(x_1 + h_1, x_2 + h_2, \dots x_k + h_k) \approx f(x_1, x_2, \dots, x_k + h_1\frac{\partial f_i}{\partial x_1} + h_2\frac{\partial f_i}{\partial x_2} + \dots + h_k\frac{\partial f_i}{\partial x_k}) \qquad (142)$$

Przenosząc na lewą stronę wartości funkcji $f_i(x_1, x_2, \dots, x_k)$ i postępując analogicznie z pozostałymi funkcjami f_i otrzymujemy układ równań:

$$\begin{cases} f_1(x_1, x_2, \dots, x_k) = f_1(x_1, x_2, \dots, x_k + h_1\frac{\partial f_1}{\partial x_1} + h_2\frac{\partial f_1}{\partial x_2} + \dots + h_k\frac{\partial f_1}{\partial x_k}) \\ f_2(x_1, x_2, \dots, x_k) = f_2(x_1, x_2, \dots, x_k + h_1\frac{\partial f_2}{\partial x_1} + h_2\frac{\partial f_2}{\partial x_2} + \dots + h_k\frac{\partial f_2}{\partial x_k}) \\ \vdots \\ f_k(x_1, x_2, \dots, x_k) = f_k(x_1, x_2, \dots, x_k + h_1\frac{\partial f_k}{\partial x_1} + h_2\frac{\partial f_k}{\partial x_2} + \dots + h_k\frac{\partial f_k}{\partial x_k}) \end{cases} \qquad (143)$$

Powyższy układ równań jest układem równań liniowych który można rozwiązać względem wektora $H = [h_1, h_2, \dots h_k]^T$

Stosując jedną z poznanych metod dla układów równań liniowych, np. metodę eliminacji Gaussa. Macierz współczynników tego układu równań A oraz wektor wyrazów wolnych b mają postać

$$A = F'(X) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_k} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_k} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_k}{\partial x_1} & \frac{\partial f_k}{\partial x_2} & \dots & \frac{\partial f_k}{\partial x_k} \end{bmatrix} \qquad (144)$$

$$b = F(X) = \begin{bmatrix} -f_1(x_1, x_2, \dots, x_k) \\ -f_2(x_1, x_2, \dots, x_k) \\ \vdots \\ -f_k(x_1, x_2, \dots, x_k) \end{bmatrix} \qquad (145)$$

Kolejne przybliżenie wektora miejsc zerowych X uzyskujemy dodając do poprzedniego wektora X wyznaczony wektor H. Zatem oznaczając: $X^{(i)} = \left[x_1^{(i)}, x_2^{(i)}, ..., x_k^{(i)}\right]$ możemy opisać tę metodę jako metodę składającą się z dwóch poniższych kroków powtarzanych iteracyjnie:

1) Wyznaczamy wektor nowych poprawek $H^{(i)}$, rozwiązując względem $H^{(i)}$ układ równań liniowych

$$F'(X^{(i)})H^{(i)} = F(X^{(i)}) \qquad (146)$$

2) Wyznaczamy nowy wektor przybliżający miejsca zerowe

$$X^{(i+1)} = X^{(i)} + H^{(i)} \qquad (147)$$

2025-05-06

3.1.3 Metoda siecznych

Jedną z głównych trudności pojawiających się w praktycznych zastosowaniach metody Newtona jest konieczność wyznaczenia pochodnej $f'(x_n)$. Wstawiając do wzoru Newtona:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \qquad (148)$$

przybliżoną wartość pochodnej otrzymujemy inne metody rozwiązywania równań nieliniowych. Przypomnijmy, że dokładna wartość pochodnej wyraża się wzorem:

$$f'(x_n) = \lim_{h \to 0} \frac{f(x_n + h) - f(x_n)}{h} \qquad (149)$$

W zależności od wyboru postaci przybliżenia pochodnej uzyskujemy różne metody:

1. Biorąc $h = f(x_n)$ bez przechodzenia z h do granicy otrzymujemy metodę Steffensena:

$$\begin{aligned} x_{n+1} &= x_n - \frac{f(x_n)}{f'(x_n)} \\ &= x_n - \frac{f(x_n) \cdot h}{f(x_n + h) - f(x_n)} \\ &= x_n - \frac{[f(x_n)]^2}{f(x_n + f(x_n)) - f(x_n)} \end{aligned} \qquad (150)$$

2. Biorąc przybliżenie pochodnej ilorazami różnicowymi:

$$f'(x_n) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}} \qquad (151)$$

   otrzymujemy metodę siecznych:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} = x_n - f(x_n) \cdot \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} \qquad (152)$$

W metodzie siecznych do wyznaczenia kolejnego przybliżenia konieczne jest pamiętanie 2 poprzednich przybliżeń. Graficznie metodę tę można zinterpretować jako wyznaczanie siecznej do wykresu funkcji f, przechodzącego przez dwa poprzednio wyznaczone punkty, a następnie wyznaczenie punktu przecięcia tej siecznej z osią OX, który stanowi nowe przybliżenie miejsca zerowego.

Metoda siecznych jest wolniej zbieżna do dokładnej wartości miejsca zerowego niż metoda Newtona, ale szybciej niż metoda bisekcji.

Jeżeli r jest dokładną wartością miejsca zerowego, to:

$$|x_{n+1} - r| \leq |C(x_n - r)^\alpha| \qquad (153)$$

gdzie:

- C jest pewną stałą
- $\alpha = \frac{1+\sqrt{5}}{2} \approx 1.62$

3.2 Pierwiastki wielomianów

Szczególnym rodzajem funkcji nieliniowych są wielomiany

Wielomianem nazywamy funkcję:

$$w(x) = a_n x^n + a_{n-1}x^{n-1} + \ldots + a_1 x + a_0 \qquad (154)$$

gdzie

- $a_n \neq 0$
- a_i są liczbami rzeczywistymi (lub zespolonymi) nazywanymi współczynnikami wielomianu

Liczbę naturalną n będącą indeksem współczynnika przy najwyższej potędze x w wielomianie $w(x)$, nazywamy stopniem tego wielomianu.

W określeniu liczby i przedziałów w których znajdują się miejsca zerowe wielomianów przydatne są następujące twierdzenie:

Twierdzenie 3.2: Zasadnicze twierdzenie algebry

Wielomian $w(x)$ stopnia n ma n pierwiastków zespolonych i może być przedstawiony w postaci:

$$w(x) = a_n(x - z_1)(x - z_2)\ldots(x - z_n) \qquad (155)$$

gdzie z_i są jego pierwiastkami zespolonymi.
Wielomian o współczynnikach rzeczywistych może być przedstawiony w postaci iloczynu czynników liniowych postaci $(x - r_i)$ odpowiadających jego pierwiastkom rzeczywistym r_i oraz czynników kwadratowych nierozkładalnych $(x^2 + p_i x + q_i) = (x - z_i)(x - \overline{z}_i)$ odpowiadających jego pierwiastkom zespolonym z_i i $\overline{z}_i$ czyli

$$w(x) = a_n(x - r_1)(x - r_2)\ldots(x - r_k)(x^2 + p_1 x + q_1)\ldots(x^2 + p_s x + q_s) \qquad (156)$$

Twierdzenie 3.3

Wszystkie pierwiastki wielomianu $w(x)$ leżą w kole o promieniu

$$\varrho = 1 + \frac{\max\limits_{0 \leq k \leq n} |a_k|}{|a_n|} \qquad (157)$$

Oznaczmy przez x_0 dowolne miejsce zerowe wielomianu

$$\begin{aligned} s(x) &= x^n w\left(\frac{1}{x}\right) \\ &= x^n \left[a_n \frac{1}{x^n} + a_{n-1}\frac{1}{x^{n-1}} + \ldots + a_1 \frac{1}{x} + a_0\right] \\ &= a_0 x^n + a_1 x^{n-1} + \ldots a_{n-1} x + a_n \end{aligned} \qquad (158)$$

Z definicji tej wynika, że jeśli $x_0 \neq 0$, to $\frac{1}{x_0}$ jest miejscem zerowym wielomianu $w(x)$.

Stosując powyższe twierdzenie do pierwiastków wielomianu $s(x)$ otrzymujemy, że:

$$|x_0| \leq \varrho_1 = 1 + \frac{\max\limits_{0<k\leq n} |a_k|}{|a_0|} \qquad (159)$$

Ponieważ $\frac{1}{x_0}$ jest pierwiastkiem wielomianu $w(x)$, a $|x_0| \leq \varrho_1$, to $\frac{1}{x_0} \geq \frac{1}{\varrho_1}$. Otrzymujemy stąd następującą uwagę:

Achtung! 5

Wszystkie niezerowe pierwiastki wielomianu $w(x)$ leżą na zewnątrz koła o promieniu

$$\frac{1}{\varrho_1} = \frac{1}{1 + \frac{\max\limits_{0<k\leq n} |a_k|}{|a_0|}} \qquad (160)$$

Przykład 3.1

Znajdź przedziały w których znajdują się rzeczywiste miejsca zerowe wielomianu:

$$w(x) = x^4 - 4x^3 + 7x^2 - 5x - 2 \qquad (161)$$

Dla wielomianu $w(x)$ mamy:

$$\varrho = 1 + \frac{\max\limits_{0\leq k<n} |a_k|}{|a_n|} = 1 + \frac{7}{1} = 8 \qquad (162)$$

$$\frac{1}{\varrho} = \frac{1}{1 + \frac{\max\limits_{0<k\leq n} |a_k|}{|a_0|}} = \frac{1}{1 + \frac{7}{2}} = \frac{1}{\frac{9}{2}} = \frac{2}{9} \qquad (163)$$

Zatem dla każdego miejsca zerowego x_0 zachodzi nierówność:

$$\frac{2}{9} \leq |x_0| \leq 8 \qquad (164)$$

Czyli pierwiastki rzeczywiste znajdują się w zbiorze

$$A = \left[-8; -\frac{2}{9}\right] \cup \left[\frac{2}{9}; 8\right] \qquad (165)$$

3.2.1 Schemat Hornera

Zastosowania Schematu Hornera:

Wyznaczanie wartości wielomianu Zauważmy, że obliczając wartość wielomianu

$$w(x_0) = a_n x_0^n + a_{n-1} x_0^{n-1} + \ldots + a_1 x + a_0 \qquad (166)$$

w podanej we wzorze kolejności wymaga to n dodawań i $n + (n-1) + \ldots + 1 = \frac{n(n+1)}{2}$ mnożeń.

Znacznie efektywniejsze jest obliczanie wartości tego wielomianu w kolejności zgodnej z poniższym wzorem Hornera

$$w(x_0) = ((\ldots((a_n x_0 + a_{n-1})x_0 + a_{n-2})x_0 \ldots + a_1)x_0 + a_0 \qquad (167)$$

W pierwszym kroku wyznaczania wartości wielomianu za pomocą algorytmu Hornera do sumy reprezentującej wartość wielomianu bierzemy współczynnik a_n, a następie w każdym kolejnym kroku przemnażamy dotychczasową sumę przez wartość x_0 oraz dodajemy kolejny współczynnik wielomianu a_i. Takie wyznaczanie wartości wielomianu wymaga jedynie n dodawań i n mnożeń.

Dzielenie wielomianów przez czynnik liniowy Wielomian $w(x)$ możemy zapisać w postaci:

$$w(x) = (x - x_0)q(x) + r(x_0) \qquad (168)$$

gdzie:

- $q(x)$ jest ilorazem
- $r(x_0)$ jest resztą z dzielenia tego wielomianu przez czynnik $x - x_0$

Wstawiając $x = x_0$ otrzymujemy:

$$w(x_0) = r(x_0) \qquad (169)$$

Stopień wielomianu $q(x)$ jest o 1 mniejszy od stopnia wielomianu $w(x)$, zatem może on być przedstawiony w postaci:

$$q(x) = b_{n-1}x^{n-1} + b_{n-2}x^{n-2} + \ldots + b_1 x + b_0 \qquad (170)$$

Wstawiając $q(x)$ do równania i uwzględniając że $r(x_0) = w(x_0)$ jest wartością wielomianu w punkcie x_0 mamy:

$$\begin{aligned} a_n x^n + & a_{n-1}x^{n-1} + \ldots + a_1 x + a_0 \\ &= (x - x_0)(b_{n-1}x^{n-1} + b_{n-2}x^{n-2} + \ldots + b_1 x + b_0) + w(x_0) \\ &= b_{n-1}x^n + (b_{n-2} - b_{n-1}x_0)x^{n-1} + \ldots + (b_0 - b_1 x_0)x - b_0 x_0 + w(x_0) \end{aligned} \qquad (171)$$

Porównując współczynniki przy tych samych potęgach po obu stronach równania otrzymujemy:

$$\begin{cases} b_{n-1} = a_n \\ b_{n-2} = a_{n-1} + b_{n-1}x_0 \\ \vdots \\ b_0 = a_1 + b_1 x_0 \\ w(x_0) = a_0 + b_0 x_0 \end{cases} \qquad (172)$$

Zauważmy, że wykonywane są tutaj identyczne działania jak w przypadku wyznaczania wartości wielomianu $w(x)$ w punkcie x_0, a wartości sumy uzyskiwane w kolejnych krokach są wartościami kolejnych współczynników ilorazu $q(x)$. Ostatnia uzyskana wartość (wartość wielomianu w punkcie x_0) jest jednocześnie resztą z dzielenia wielomianu $w(x)$ przez czynnik $x - x_0$

2025-05-20

3.2.2 Zastosowanie – Deflacja wielomianu

Jeśli x_0 jest pierwiastkiem wielomianu $w(x)$, to jego reszta z dzielenia przez czynnik $x - x_0$ jest równa zero i czynnik $(x - x_0)$ można wyłączyć z wielomianu $w(x)$

$$w(x) = (x - x_0)q(x) \qquad (173)$$

W kolejnych krokach z uzyskiwanych ilorazów $q(x)$ można wyłączać kolejne pierwiastki wielomianu, czyli dokonywać jego deflację.

3.2.3 Zastosowanie – rozwinięcie Taylora – wyznaczanie pochodnych wielomianu

Dla dowolnego x_0 (nie koniecznie pierwiastka wielomianu $w(x)$) możemy n-krotnie (gdzie n to stopień wielomianu) wykonywać dzielenie kolejnych ilorazów przez czynnik $(x - x_0)$. Prowadzi to do przedstawienia wielomianu $w(x)$ w postaci

$$\begin{aligned} w(x) &= a_n x^n + a_{n-1}x^{n-1} + \ldots + a_1 x + a_0 \\ &= c_n(x - x_0)^n + c_{n-1}(x - x_0)^{n-1} + \ldots + c_1(x - x_0) + c_0 \end{aligned} \tag{174}$$

gdzie współczynniki c_i są kolejnymi resztami z dzielenia przez czynnik $(x - x_0)$

Taka postać wielomianu $w(x)$ jest po prostu rozwinięciem tego wielomianu w szereg Taylora. Jako że rozwinięcie wielomianu w szereg Taylora do wyrazu n-tego (numerując od zera) jest dokładne (reszta rozwinięcia jest równa zeru). Takie rozwinięcie może posłużyć do wyznaczania pochodnych wielomianu $w(x)$ w punkcie x_0. Dla rozwinięcia $w(x)$ w szereg Taylora mamy:

$$w(x) = w(x_0) + \frac{w'(x_0)}{1!}(x - x_0) + \frac{w''(x_0)}{2!}(x - x_0)^2 + \ldots + \frac{w^{(n-1)}(x_0)}{(n-1)!}(x - x_0)^{n-1} + \frac{w^{(n)}(x_0)}{n!}(x - x_0)^n \tag{175}$$

Porównując współczynniki tego rozwinięcia ze współczynnikami c_i mamy

$$\begin{cases} w^{(n)}(x_0) = c_n \cdot n! \\ w^{(n-1)}(x_0) = c_{n-1} \cdot (n-1)! \\ \vdots \\ w'(x_0) = c_1 \\ w(x_0) = c_0 \end{cases} \tag{176}$$

Wyznaczone wartości $w(x_0)$ i $w'(x_0)$ mogą posłużyć do wykonania kroku metody Newtona w celu wyznaczenia miejsca zerowego wielomianu $w(x)$

Przykład 3.2

Stosując schemat Hornera oblicz $w(3)$ dla

$$w(x) = x^4 - 4x^3 + 7x^2 - 5x - 2 \qquad (177)$$

$$\begin{array}{c|ccccc} & 1 & -4 & 7 & -5 & -2 \\ 3 & & 3 & -3 & 12 & 21 \\ \hline & 1 & -1 & 4 & 7 & 19 \end{array} \qquad (178)$$

$$w(3) = 19 \qquad (179)$$

Przykład 3.3

Wykonaj deflację wielomianu z poprzedniego przykładu dla jego pierwiastka $x_0 = 2$

$$\begin{array}{c|ccccc} & 1 & -4 & 7 & -5 & -2 \\ 2 & & 2 & -4 & 6 & 2 \\ \hline & 1 & -2 & 3 & 1 & 0 \end{array} \qquad (180)$$

Zatem

$$w(x) = (x-2)(x^3 - 2x^2 + 3x + 1) \qquad (181)$$

Przykład 3.4

Znajdź rozwinięcie w szereg Taylora wielomianu z poprzednich przykładów w punkcie $x_0 = 3$. Wyznacz $w''(3)$

	1	-4	7	-5	-2
3		3	-3	12	21
	1	-1	4	7	$\underset{c_0}{19}$
3		3	6	30	
	1	2	10	$\underset{c_1}{37}$	
3		3	15		
	1	5	$\underset{c_2}{25}$		
3		3			
	1	$\underset{c_3}{8}$			
	$\underset{c_4}{1}$				

(182)

3.3 Metoda Bairstowa

Metoda Bairstowa służy do wyłączania z wielomianu o współczynnikach rzeczywistych czynnika kwadratowego nierozkładalnego.

Dzieląc wielomian

$$w(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0 \qquad (183)$$

przez trójmian kwadratowy $x^2 - ux - v$ uzyskamy resztę (wielomian 1 stopnia) postaci $r(x) = b_1(x - u) + b_0$, mamy zatem

$$\sum_{k=0}^{n} a_k x^k = (x^2 - ux - v)\left(\sum_{k=2}^{n} b_k x^{k-2}\right) + b_1(x - u) + b_0 \qquad (184)$$

gdzie przyjmujemy $b_{n+1} = b_{n+2} = 0$ i porównując po obu stronach współczynniki wielomianów otrzymujemy wzór rekurencyjny

$$a_i = -vb_{i+2} - ub_{i+1} + b_i \qquad \text{czyli} \qquad b_k = a_k + ub_{k+1} + vb_{k+2} \qquad (185)$$

Aby dokonać teraz deflacji wielomianu $w(x)$ trójmianem kwadratowym nierozkładalnym wystarczy znaleźć taki trójmian kwadratowy (współczynniki u i v), że $b_1(u, v) = b_0(u, v) = 0$. Można to zrobić

stosując metodę Newtona dla układów równań. Potrzebujemy jednak wówczas wartości pochodnych funkcji $b_1(u,v)$ i $b_0(u,v)$. Wyznaczyć je można ze wzoru rekurencyjnego uzyskanego i różniczkowane równanie rekurencyjne względem u i względem r. Jeśli oznaczymy sobie $c_k = \frac{\partial b_k}{\partial u}$ i $d_k = \frac{\partial b_{k-1}}{\partial v}$, to biorąc pochodną po u mamy:

$$c_k = b_{k+1} + c_{k+1} \cdot u + v c_{k+2} = b_{k+1} + u\frac{\partial b_{k+1}}{\partial u} + v\frac{\partial b_{k+2}}{\partial u} \qquad (186)$$

a biorąc pochodną po v

$$\frac{\partial b_k}{\partial v} = d_k = u\frac{\partial b_k}{\partial v} + b_{k+1} + v\frac{\partial b_{k+1}}{\partial v} = b_{k+1} + u d_{k+1} + v d_{k+2} \qquad (187)$$

Ponieważ wzory te są identyczne i $c_{n+1} = c_n = d_{n+1} = d_n = 0$, wartości tych pochodnych także są identyczne (można je wyznaczać tylko raz)

Dla danego przybliżenia (u,v), kolejne przybliżenie będzie miało postać $(u+\delta u, v+\delta v)$. Stosując metodę Newtona do $b_0(u+\delta u, v+\delta v) = 0$ mamy:

$$b_0(u,v) + \frac{\partial b_0}{\partial u}\delta u + \frac{\partial b_0}{\partial v}\delta v = 0 \qquad (188)$$

$$b_1(u,v) + \frac{\partial b_1}{\partial u}\delta u + \frac{\partial b_1}{\partial v}\delta v = 0 \qquad (189)$$

Ponieważ $\frac{\partial b_0}{\partial u} = c_0$, $\frac{\partial b_1}{\partial v} = c_1$, zatem ten układ równań można zapisać jako:

$$\begin{bmatrix} c_0 & c_1 \\ c_1 & c_2 \end{bmatrix} \begin{bmatrix} \delta u \\ \delta v \end{bmatrix} = \begin{bmatrix} -b_0(u,v) \\ -b_1(u,v) \end{bmatrix} \qquad (190)$$

I rozwiązując ten układ równań liniowych mamy: $J = c_0 c_2 - c_1^2$

$$\delta u = \frac{c_1 b_1 - c_2 b_0}{J} \quad \text{i} \quad \delta v = \frac{c_1 b_0 - c_0 b_1}{J} \qquad (191)$$

Zatem nowym przybliżeniem współczynniku trójmianu jest $u+\delta u$, $v+\delta v$ i w kolejnych krokach wyznaczamy kolejne przybliżenia, aż do uzyskania satysfakcjonującej dokładności współczynników.

3.4 Metoda Laguerre'a

Metoda Laguerre'a jest metodą iteracyjną służącą do wyznaczenia pierwiastków wielomianów. W metodzie tej dla i-tego przybliżenia pierwiastka $x^{(i)}$ wyznaczamy wartości liczbowe A, B, C na podstawie których wyznaczamy kolejne przybliżenia pierwiastka. Wartości te wyrażają się wzorami:

$$A = -\frac{w'(x^{(i)})}{w(x^{(i)})} \qquad B = A^2 - \frac{w''(x^{(i)})}{w(x^{(i)})} \qquad C = \frac{1}{n}\left(A \pm \sqrt{(n-1)(nB - A^2)}\right) \tag{192}$$

gdzie n jest stopniem wielomianu, a znak przy wyznaczaniu wartości C powinien być tak dobrany, aby $|C|$ była jak największa. Kolejne przybliżenia pierwiastka jest równe $x^{(i+1)} = x^{(i)} + \frac{1}{C}$

2025-05-27

Achtung! 6

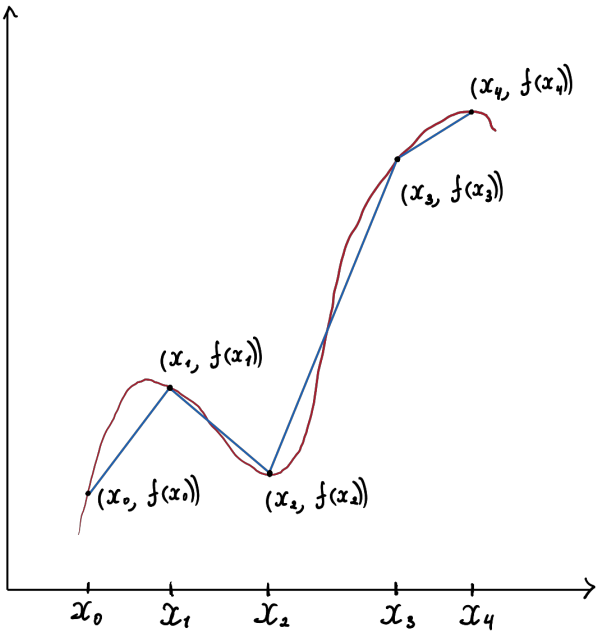
Terminy egzaminów są dostępne (czy będą dostępne) na stronie doktora

4 Interpolacja

Interpolacja przybliżenie nieznanej funkcji na podstawie przyjmowanych przez nią wartości w zadanych punktach jej dziedziny za pomocą funkcji mającej w tych punktach te same wartości co nieznana funkcja nazywamy **interpolacją**.

Funkcja interpolowana, interpolująca Nieznaną funkcję nazywamy funkcją **interpolowaną**, a funkcja która ją przybliża funkcją **interpolującą**

Węzły interpolacji Punkty dziedziny funkcji interpolowanej w których jej wartości są znane nazywamy **węzłami interpolacji**



Rysunek 2: Przykład funkcji interpolujących

4.1 Interpolacja wielomianowa

W interpolacji wielomianowej funkcją interpolującą jest wielomian możliwie najniższego stopnia. W interpolacji wielomianowej dla danych $n+1$ punktów $x_0, x_1, \ldots, x_n$, w których interpolowana funkcja przyjmuje odpowiednio wartości $y_0, y_1, \ldots, y_n$, szukamy takiego wielomianu $w(x)$ stopnia n, że

$$w(x_i) = y_i \qquad \text{dla} \quad 0 \leq i \leq n \qquad (193)$$

Twierdzenie 4.1

Jeżeli $f : \mathbb{R} \to \mathbb{R}$, a $x_0, x_1, \ldots, x_n$ są parami różnymi liczbami rzeczywistymi, to istnieje dokładnie jeden wielomian stopnia co najwyżej n spełniający warunek

$$w(x_i) = y_i \qquad \text{dla} \quad 0 \leq i \leq n \qquad (194)$$

Zauważmy, że jeśli $f(x_0) = y_0$, to wielomian $w_0(x) = y_0$ interpoluje tę funkcję. Jeżeli chcemy, aby wielomian ten interpolował także funkcję f w punkcie x_1 w którym $f(x_1) = y_1$, wystarczy dodać do niego składnik zerujący się w x_0 (aby nie zmieniał wartości w punkcie x_0) i modyfikujący wartość wielomianu $w_0(x)$ w punkcie x_1, zatem

$$w(x_1) = w_0(x) + c_1(x - x_0) = y_0 + c_1(x - x_0) \qquad (195)$$

gdzie $c_1 = \dfrac{y_1 - y_0}{x_1 - x_0}$

Postępując analogicznie otrzymujemy

$$w_2(x) = w_1(x) + c_2(x - x_0)(x - x_1) \qquad (196)$$

gdzie $c_2 = \dfrac{y_2 - w_1(x_2)}{(x_2 - x_0)(x_2 - x_1)}$

Wzór interpolacyjny Newtona Ogólny wzór na wielomian interpolujący ma zatem postać

$$w_k(x) = \sum_{i=0}^{k} c_i \prod_{j=0}^{i-1}(x - x_j) = \sum_{i=0}^{k} c_i q_i(x) \qquad (197)$$

Jest to tzw. **wzór interpolacyjny Newtona**.

Ponieważ wyznaczanie współczynników c_i z tego wzoru jest kosztowne i może powodować duże błędy zaokrągleń, dlatego wygodniejsze jest wyznaczanie tych współczynników za pomocą tzw. wzoru różnicowego.

Wprowadźmy następujące oznaczenie ilorazów różnicowych rzędu pierwszego

$$f[x_i] = f(x_i) \qquad \text{ilorazów róznicowy rzędu pierwszego} \qquad (198)$$

Ilorazy różnicowe rzędu $k+1$ definiujemy za pomocą poniższego wzoru rekurencyjnego

$$f[x_0, x_1, \ldots, x_k] = \frac{f[x_1, x_2, \ldots, x_k] - f[x_0, x_1, \ldots, x_{k-1}]}{x_k - x_0} \qquad (199)$$

Przy tak zdefiniowanych ilorazach różnicowych, możemy za ich pomocą zdefiniować wartości współczynników c_i ze wzoru interpolacyjnego Newtona za pomocą zależności

$$c_i = f[x_0, x_1, \ldots, x_i] \qquad (200)$$

Zatem wzór interpolacyjny Newtona przybiera postać

$$w(x) = \sum_{k=0}^{n} f[x_0, x_1, \ldots, x_k] \cdot \prod_{j=0}^{k-1}(x - x_j) \qquad (201)$$

Ilorazy różnicowe najwygodniej jest wyznaczać w postaci tabeli

x_0	$f[x_0]$	$f[x_0, x_1]$	$f[x_0, x_1, x_2]$	$f[x_0, x_1, x_2, x_3]$
x_1	$f[x_1]$	$f[x_1, x_2]$	$f[x_1, x_2, x_3]$	
x_2	$f[x_2]$	$f[x_2, x_3]$		
x_3	$f[x_3]$			

Wtedy wyrazy znajdujące się w pierwszym wierszu stanowią kolejne współczynniki c_i

Przykład 4.1

Wyznacz postać wielomianu interpolacyjnego Newtona dla punktów

x	y
5	1
-7	-23
-6	-54
0	-954

5	1	2	3	4
-7	-23	-31	-17	
-6	-54	-150		
0	-954			

$$f[x_0, x_1] = \frac{f[x_0] - f[x_1]}{x_1 - x_0} = \frac{-23 - 1}{-7 - 5} = 2 \qquad (202)$$

$$f[x_1, x_2] = \frac{f[x_2] - f[x_1]}{x_2 - x_1} = \frac{-54 + 23}{-6 + 7} = -31 \qquad (203)$$

$$f[x_2, x_3] = \frac{-954 + 54}{0 + 6} = -150 \qquad (204)$$

$$f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} = \frac{-31 - 2}{-6 - 5} = 3 \qquad (205)$$

$$f[x_1, x_2, x_3] = \frac{-150 + 31}{0 + 7} = -17 \qquad (206)$$

$$f[x_0, x_1, x_2, x_3] = \frac{f[x_1, x_2, x_3] - f[x_0, x_1, x_2]}{x_3 - x_0} = \frac{17 - 3}{0 - 5} = 4 \qquad (207)$$

Wielomian interpolacyjny ma postać

$$w(x) = 1 + 2 \cdot (x - 5) + 3 \cdot (x - 5)(x + 7) + 4 \cdot (x - 5)(x + 7)(x + 6) \qquad (208)$$

Z algorytmicznego punktu widzenia wygodniej jest wyznaczać ilorazy różnicowe “od dołu” nadpisując poprzednie wartości. Dzięki temu do wyznaczenia współczynników c_i wystarczy jedynie $(n + 1)$-elementowy wektor.

Postać Newtona nie jest jedyną postacią w jakiej wielomian interpolacyjny może być wyrażony. Oczywiście na mocy jednoznaczności wielomianu interpolacyjnego wszystkie inne postaci wielomianu interpolacyjnego dotyczą tego samego wielomianu, różnią się jedynie sposobem jego zapisu.

Wielomian interpolacyjny w postaci Lagrange’a wyrażamy jako następującą sumę

$$w(x) = \sum_{k=0}^{n} y_k l_k(x) \qquad (209)$$

gdzie

- y_k jest rzędną w naszych węzłach
- $l_k(x)$ są wielomianami które przyjmują wartość 1 dla $x = x_k$ i 0 dla $x = x_i$, gdzie $i \neq k$

Dzięki temu po wstawieniu do wzoru (209) w miejscu x punktu x_i uzyskamy wartość y_i (jedynym niewyzerowanym składnikiem sumy będzie $y_i l_i(x_i) = y_i$). Aby wielomiany $l_i(x)$ zerowały się dla x_j, gdzie $j \neq i$, muszą mieć postać:

$$l_i(x) = c(x - x_0)(x - x_1) \cdots (x - x_{i-1})(x - x_{i+1}) \cdots (x - x_n) \qquad (210)$$

zaś stała c jest tak dobrana aby $l_i(x_i) = 1$, czyli ostatecznie

$$l_i(x_i) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j} \qquad (0 \leq i \leq n) \qquad (211)$$

a

$$w(x) = \sum_{k=0}^{n} y_k \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j} \qquad (212)$$

Przykład 4.2

Dla punktów z poprzedniego przykładu

x	y
5	1
-7	-23
-6	-54
0	-954

wyznacz postać wielomianu interpolacyjnego Lagrange'a
Mamy:

$$l_0(x) = \frac{(x+7)(x+6)x}{(5+7)(5+6)5} = \frac{1}{660}(x+7)(x+6x) \qquad (213)$$

$$l_1(x) = \frac{(x-5)(x+6)x}{(-7-5)(-7+6)(-7)} = -\frac{1}{84}(x-5)(x+6)x \qquad (214)$$

$$l_2(x) = \frac{(x-5)(x+7)x}{(-6-5)(-6+7)(-6)} = -\frac{1}{66}(x-5)(x+7)x \qquad (215)$$

$$l_3(x) = \frac{(x-5)(x+7)(x+6)}{(0-5)(0-7)(0+6)} = -\frac{1}{210}(x-5)(x+7)(x+6) \qquad (216)$$

i wielomian interpolacyjny ma postać

$$w(x) = l_0(x) - 23l_1(x) - 54l_2(x) - 954l_3(x) \qquad (217)$$

Wielomian interpolacyjny można również przedstawić w postaci:

$$w(x) = a_n x^n + a_{n-1}x^{n-1} + \ldots + a_1 x + a_0 \qquad (218)$$

Zadanie interpolacyjne polega wówczas na wyznaczeniu współczynników $a_n, a_{n-1}, \ldots, a_1, a_0$

Dokonuje się tego rozwiązując układ równań liniowych takiej postaci:

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \qquad (219)$$

Błąd interpolacji można oszacować za pomocą poniższego twierdzenia

Twierdzenie 4.2

Jeśli $f \in C^{n+1}[a;b]$, a wielomian $w(x)$ stopnia co najwyżej n interpoluje wartości funkcji f w $n+1$ różnych punktach: $x_0, x_1, \ldots, x_n$ przedziału $[a;b]$, to dla dowolnego $x \in [a;b]$ istnieje

punkt $\xi_x \in (a;b)$ taki, że

$$f(x) - w(x) = \frac{1}{(n+1)!} \cdot f^{(n+1)}(\xi_x) \cdot \prod_{i=0}^{n}(x - x_i) \qquad (220)$$

2025-06-03

4.1.1 Wielomiany Czebyszewa

Jeśli mamy możliwość wyboru węzłów interpolacji, to można je dobrać w taki sposób, aby zminimalizować błąd interpolacji. Służą do tego miejsca zerowe wielomianów Czebyszewa.

Wielomianami Czebyszewa nazywamy wielomiany zdefiniowane za pomocą poniższego wzoru rekurencyjnego:

$$\begin{cases} T_0(x) = 1 \\ T_1(x) = x \\ T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x) \quad (n \geq 2) \end{cases} \qquad (221)$$

Twierdzenie 4.3

Jeśli $f \in C^{n+1}[-1;1]$, a wielomian $w(x)$ stopnia co najwyżej n interpoluje wartości funkcji f w $n+1$ punktach, które są miejscami zerowymi wielomianu Czebyszewa $T_{n+1}(x)$ postaci $x_i = \cos\left(\frac{(2i+1)\pi}{2n+2}\right)$ $(0 \leq i \leq n)$, to dla dowolnego $x \in [-1;1]$ mamy

$$|f(x) - w(x)| \leq \frac{1}{2^n(n+1)!} \cdot \max_{y\in[-1,1]} \left|f^{(n+1)}(y)\right| \qquad (222)$$

4.1.2 Interpolacja Hermite'a

Zadanie interpolacyjne można uogólnić na sytuację w której żądamy aby nie tylko wartości funkcji interpolowanej i wielomianu interpolującego w węzłach interpolacji były równe, ale także, aby równe były wartości pochodnych w tych węzłach, aż do pewnego rzędu. Wymagamy zatem aby wielomian interpolujący spełniał warunki $w^{(j)}(x_i) = c_{ij} \quad (0 \leq j \leq k_i - 1, 0 \leq i \leq m)$

gdzie c_{ij} jest wartością j-tej pochodnej funkcji interpolowanej f w punkcie x_i. Liczbę takich warunków oznaczamy przez $n + 1 = k_0 + k_1 + \ldots + k_m$

Twierdzenie 4.4

Istnieje dokładnie jeden wielomian $w(x)$ stopnia co najwyżej n spełniający powyższe warunki interpolacji Hermite'a.

Rozwiązanie tego problemu można uzyskać stosując wzór Newtona i uogólniając pojęcie ilorazów różnicowych.

Uporządkujmy węzły niemalejąco biorąc każdy węzeł x_i k_i-krotnie (tyle razy, ile mamy warunków na pochodne i wartość funkcji w węźle x_i). Ilorazy różnicowe $f[x_0, x_1, \ldots, x_n]$ definiujemy teraz za pomocą wzoru

$$f[\underbrace{x_i, x_i, \ldots, x_i}_{n+1}] = \frac{1}{n!} f^{(n)}(x_i) \qquad (223)$$

jeśli $x_0 = x_n$ i za pomocą wzoru

$$f[x_0, x_1, \ldots, x_n] = \frac{f[x_1, x_2, \ldots, x_n] - f[x_0, x_1, \ldots, x_{n-1}]}{x_n - x_0} \quad \text{jeśli } x_0 \neq x_n \qquad (224)$$

Wzór interpolacyjny Newtona ma taką samą postać, jak w klasycznym wzorze Newtona, ale uwzględnia dodatkowo krotność węzłów (wyrażenia $(x - x_i)$ mogą występować w wyższych niż 1 potęgach)

Przykład 4.3

Znajdź wielomian interpolacyjny $w(x)$ spełniający warunki:

$$w(1) = 2 \quad w'(1) = 3 \quad w(2) = 6 \quad w'(2) = 7 \quad w''(2) = 8 \qquad (225)$$

Tabelę ilorazów różnicowych ma wówczas postać:

x_0	$f[x_0]$	$f[x_0, x_0]$	$f[x_0, x_0, x_1]$	$f[x_0, x_0, x_1, x_1]$	$f[x_0, x_0, x_1, x_1, x_1]$
x_0	$f[x_0]$	$f[x_0, x_1]$	$f[x_0, x_1, x_1]$	$f[x_0, x_1, x_1, x_1]$	
x_1	$f[x_1]$	$f[x_1, x_1]$	$f[x_1, x_1, x_1]$		
x_1	$f[x_1]$	$f[x_1, x_1]$			
x_1	$f[x_1]$				

Tabela ilorazów różnicowych ma wówczas postać:

$$
\begin{aligned}
f[x_0, x_0] = f'(x_0) = w'(1) = 3 \qquad f[x_0, x_1] &= \frac{f[x_1] - f[x_0]}{x_1 - x_0} = \frac{6-2}{2-1} = 4 \\
f[x_1, x_1] = w'(2) = 7 \qquad f[x_0, x_0, x_1] &= \frac{f[x_0, x_1] - f[x_0, x_0]}{x_1 - x_0} = \frac{4-3}{2-1} = 1 \\
f[x_0, x_1, x_1] &= \frac{f[x_1, x_1] - f[x_0, x_1]}{x_1 - x_0} = \frac{7-4}{2-1} = 3 \\
f[x_1, x_1, x_1] &= \frac{1}{2!} f''(2) = \frac{8}{2} = 4 \qquad (226) \\
f[x_0, x_0, x_1, x_1] &= \frac{f[x_0, x_1, x_1] - f[x_0, x_0, x_1]}{x_1 - x_0} = \frac{3-1}{2-1} = 2 \\
f[x_0, x_1, x_1, x_1] &= \frac{f[x_1, x_1, x_1] - f[x_0, x_1, x_1]}{x_1 - x_0} = \frac{4-3}{2-1} = 1 \\
f[x_0, x_0, x_1, x_1, x_1] &= \frac{f[x_0, x_1, x_1, x_1] - f[x_0, x_0, x_1, x_1]}{x_1 - x_0} = \frac{1-2}{2-1} = 1
\end{aligned}
$$

1	2	3	1	2	-1
1	2	4	3	1	
2	6	7	4		
2	6	7			
2	6				

I ostatecznie

$$w(x) = 2 + 3(x-1) + (x-1)^2 + 2(x-1)^2(x-2) - (x-1)^2(x-2)^2 \qquad (227)$$

4.2 Interpolacja funkcjami sklejanymi

Zdefiniujmy pojęcie funkcji sklejanej stopnia k

Funkcja sklejana stopnia k dla $n+1$ węzłów $t_0, t_1, \ldots, t_n$ takich, że $t_0 < t_1 < \ldots < t_n$ danej liczby całkowitej k funkcją sklejaną stopnia k nazywamy taką funkcję s, która:

- W każdym z przedziałów $[t_i; t_{i+1})$ dla $(0 \leq i \leq n-1)$ jest wielomianem stopnia co najwyżej k
- Ma ciągłą $k-1$ pochodną w przedziale $[t_0; t_n]$

Szczególnie często w praktyce znajdują zastosowanie funkcje sklejane trzeciego stopnia (sześcienne). Są to zatem funkcje które między węzłami są wielomianami co najwyżej trzeciego stopnia mające wszędzie ciągłą drugą pochodną. Aby jednoznacznie określić taką funkcję musimy wyznaczyć wielomiany S_i stopnia 3 w każdym z przedziałów $[t_i; t_{i+1}]$. Każda z takich funkcji posiada 4 współczynniki, zatem należy wyznaczyć $4n$ współczynników. Warunki ciągłości funkcji S w węzłach wewnętrznych

$$S_{i-1}(t_i) = y_i = S_i(t_i) \qquad (228)$$

w połączeniu z warunkami na wartość funkcji S w węzłach brzegowych

$$S_0(t_0) = y_0 \qquad S_{n-1}(t_n) = y_n \qquad (229)$$

dają łącznie $2n$ równań wiążących współczynniki wielomianów S_i.

Ponadto mamy $n-1$ warunków na ciągłość pierwszej pochodnej i tyle samo warunków na ciągłość drugiej pochodnej w węzłach wewnętrznych

$$\begin{aligned} S'_{i-1}(t_i) &= S'_i(t_i) \qquad 1 \le i \le n-1 \\ S''_{i-1}(t_i) &= S''_i(t_i) \qquad 1 \le i \le n-1 \end{aligned} \qquad (230)$$

Mamy zatem $4n-2$ równań z $4n$ niewiadomymi. Najczęściej przyjmuje się dodatkowo, że $S''(t_0) = 0$ oraz $S''(t_n) = 0$ – otrzymujemy wówczas tzw. naturalną funkcję sklejaną.

Wprowadźmy oznaczenie $S''(t_i) = z_i$ oraz $h_i = t_{i+1} - t_i$. Ponieważ funkcje S_i są wielomianami trzeciego stopnia, zatem ich drugie pochodne są funkcjami liniowymi, na końcach przedziału $[t_i, t_{i+1}]$ przyjmujące odpowiednio wartości z_i i z_{i+1} zatem pochodne te mają postać:

$$S''_i(x) = \frac{z_i}{h_i}(t_{i+1} - x) + \frac{2i+1}{h_i}(x - t_i) \qquad (231)$$

Całkując dwukrotnie tę równość otrzymujemy

$$S_i(x) = \frac{z_i}{6h_i}(t_{i+1} - x)^3 + \frac{z_i + 1}{6h_i}(x - t_i)^3 + C(x - t_i) + D(t_{i+1} - x) \qquad (232)$$

Wstawiając do tego równania $x = t_i$ i pamiętając, że $S_i(t_i) = y_i$, wyznaczamy $D = (\frac{y_i}{h_i} - \frac{z_i h_i}{6})$ i podobnie wstawiając $x = h_{i+1}$ do równania powyższego i pamiętając, że $S_i(t_{i+1}) = y_{i+1}$ wyznaczamy $C = (\frac{y_{i+1}}{h_i} - \frac{z_{i+1} h_i}{6})$, czyli ostatecznie

$$S_i(x) = \frac{z_i}{6h_i}(t_{i+1} - x)^3 + \frac{z_{i+1}}{6h_i}(x - h_i)^3 + (\frac{y_{i+1}}{h_i} - \frac{z_{i+1}h_i}{6})(x - t_i) + (\frac{y_i}{h_i} - \frac{z_i h_i}{6})(t_{i+1} - x) \qquad (233)$$

W celu wyznaczenia wartości z_i, korzystamy z warunków na ciągłość pierwszej pochodnej w węzłach wewnętrznych. Mamy

$$S_i'(x) = \frac{z_i}{2h_i}(t_{i+1} - x)^2 + \frac{z_{i+1}}{2h_i}(x - t_i)^2 + (\frac{y_{i+1}}{h_i} - \frac{z_{i+1}}{6}) - (\frac{y_i}{h_i} - \frac{z_i h_i}{6}) \qquad (234)$$

Czyli

$$S_i'(t_i) = -\frac{h_i z_i}{3} - \frac{h_i z_{i+1}}{6} - \frac{y_i}{h_i} + \frac{y_{i+1}}{h_i} \qquad (235)$$

i

$$S_{i-1}'(t_i) = \frac{h_{i-1} z_{i-1}}{6} + \frac{h_{i-1} z_i}{3} - \frac{y_{i-1}}{h_{i-1}} + \frac{y_i}{h_{i-1}} \qquad (236)$$

Przyrównując do siebie prawe strony tych równości (jest to warunek ciągłości pierwszej pochodnej w punkcie t_i) uzyskujemy równania

$$h_{i-1} z_{i-1} + 2(h_{i-1} + h_i) z_i + h_i z_{i+1} = \frac{6}{h_i}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1}) \qquad (237)$$

dla $1 \leq i \leq n-1$ z dodatkowymi warunkami $z_0 = z_n = 0$

2025-06-10

Oznaczając $u_i = 2(h_{i-1} + h_i)$, $b_i = \frac{6}{h_i}(y_{i+1} - y_i)$, $v_i = b_i - b_{i-1}$, otrzymujemy trójprzekątniowy układ równań liniowych postaci:

$$\begin{bmatrix} u_1 & h_1 & 0 & 0 & 0 & \cdots & 0 \\ h_1 & u_2 & h_2 & 0 & 0 & \cdots & 0 \\ 0 & h_2 & u_3 & h_3 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & h_{n-2} & u_{n-1} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_{n-1} \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{n-1} \end{bmatrix} \qquad (238)$$

z którego możemy wyznaczyć wartości z_i. W ten sposób wszystkie parametry funkcji $S_i(x)$ we wzorach (233) zostały wyznaczone i można z tych wzorów korzystać do wyznaczania wartości funkcji sklejanej. W tym celu należy najpierw wyznaczyć przedział $[t_i, t_{i+1})$ w którym znajduje się wartość argumentu x. Wzór (233) można zastąpić jego lepszą z punktu widzenia arytmetyki zmiennopozycyjnej wersją:

$$f_i(x) = y_i + (x - t_i)\{C_i + (x - t_i)[B_i + (x - t_i)A_i]\} \qquad (239)$$

gdzie

- $A_i = \dfrac{z_{i+1} - z_i}{6}$
- $B_i = \dfrac{z_i}{2}$
- $C_i = -\dfrac{h_i}{6}(z_{i+1} + 2z_i) + \dfrac{1}{h_i}(y_{i+1} - y_i)$

Pojęcie naturalnej funkcji sklejanej można uogólnić na funkcję sklejaną dowolnego nieparzystego stopnia $2m + 1$. Taka funkcja jest wielomianem stopnia co najwyżej $2m + 1$ w przedziałach $[t_i; t_{i+1})$, w węzłach wewnętrznych mającą ciągłe wszystkie pochodne, aż do pochodnej rzędu $2m$ włącznie

Zdefiniujmy

$$x_f^n = \begin{cases} x^n, & x \geq 0 \\ 0, & x < 0 \end{cases} \qquad (240)$$

Ogólna postać funkcji sklejanej stopnia $2m + 1$ może być wyrażona za pomocą wzoru:

$$S(x) = \sum_{i=0}^{m} a_i x^i + \sum_{j=0}^{n} b_j (x - t_j)^{2m+1} \qquad (241)$$

gdzie współczynniki b_j można wyznaczyć z układu równań liniowych.

$$\sum_{j=0}^{n} b_j t_j^i = 0, \quad \text{dla } 0 \leq i \leq m \qquad (242)$$

Aby wyznaczyć wartości współczynników a_i, we wzorze na $S(x)$ w miejsce x należy wstawić wartości pierwszych współrzędnych węzłów t_i (czyli odcięte tych węzłów) pamiętając, że $S(t_i) = y_i$. Uzyskujemy w ten sposób $m+n+2$ niewiadomymi (współczynniki a_i i b_i), zatem wzór na $S(x)$ może być wyznaczony jednoznacznie. Można pokazać, że istnieje dokładnie jedna funkcja sklejana $2m + 1$ dla danego układu $n + 1$ węzłów (t_i, y_i)

4.3 Interpolacja trygonometryczna

Interpolacja wielomianowa nie daje zadowalających efektów w zastosowaniach do interpolacji funkcji okresowych. Do tego celu stosuje się interpolację trygonometryczną. Dla ustalenia uwagi w interpolacji trygonometrycznej przyjmuje się, że okresem podstawowym funkcji interpolowanej jest 2π. Jeżeli tak nie jest, a interpolowana funkcja $g(y)$ ma okres podstawowy równy t, to dokonując zamiany zmiennej zgodnie ze wzorem $y = \frac{2\pi x}{t}$ otrzymujemy funkcję:

$$f(x) = g(\frac{tx}{2\pi}) \qquad (243)$$

której okres jest równy 2π.

Każdą funkcję o okresie 2π można zapisać w postaci $\frac{1}{2}a_0 + \sum_{k=1}^{\infty}(a_k \cos kx + b_k \sin kx)$, gdzie współczynniki a_k i b_k wyrażają się za pomocą wzorów

$$a_k = \frac{1}{\pi}\int_{-\pi}^{\pi} f(t)\cos kt\ dt \quad b_k = \frac{1}{\pi}\int_{-\pi}^{\pi} f(t)\sin kt\ dt \qquad (244)$$

Jest to tzw. rozwinięcie funkcji okresowej w szereg Fouriera.

Analiza Fouriera przyjmuje o wiele wygodniejszą postać, jeśli uwzględni się w niej liczby zespolone w postaci wykładniczej, którą z postacią trygonometryczną wiąże wzór Eulera (jeśli rozważamy funkcję rzeczywistą to istotna będzie jedynie część rzeczywista transformaty Fouriera).

$$e^{i\phi} = \cos\phi + i\sin\phi \qquad (245)$$

Istotą interpolacji trygonometrycznej jest znalezienie współczynników c_j tzw. wielomianu wykładniczego stopnia n postaci

$$P_n(x) = \sum_{j=0}^{n} c_j e^{ijx} = \sum_{j=0}^{n} c_j (e^{ix})^j \qquad (246)$$

który w węzłach interpolacji przyjmuje te same wartości co interpolowana funkcja f.

Można pokazać, że istnieje dokładnie jeden wielomian wykładniczy stopnia co najwyżej n interpolujący funkcję w $n+1$ węzłach.

W zastosowaniach praktycznych rzadko pojawia się potrzeba wyznaczenia wielomianu interpolacyjnego wykładniczego w dowolnie dobranych węzłach. Najczęściej stosuje się tę metodę do równoległych węzłów interpolacji postaci $x_k = \frac{2k\pi}{n+1}$ i do takich węzłów się ograniczymy.

Współczynniki c_j wykładniczego wielomianu interpolacyjnego wyrażają się wzorem

$$c_j = \frac{1}{n+1}\sum_{k=0}^{n} f(x_k)e^{-ijx_k} = \frac{1}{n+1}\sum_{k=0}^{n} f(x_k)(e^{\frac{-2\pi ij}{n+1}})^k \qquad (247)$$

Wielomian interpolacyjny $P_n(x)$ można także przedstawić w postaci trygonometrycznej:

$$P_n(x) = \frac{1}{2}a_0 + \sum_{j=1}^{n}(a_j \cos jx + b_j \sin jx) + \frac{\delta}{2}a_{m+1}\cos(m+1)x \qquad (248)$$

gdzie

$$\begin{cases} \delta = 0, m = \frac{n}{2}, & \text{dla } n \text{ parzystych} \\ \delta = 1, m = \frac{n-1}{2}, & \text{dla } n \text{ nieparzystych} \end{cases} \quad \text{oraz} \quad \begin{aligned} a_j &= \frac{2}{n+1}\sum_{k=0}^{n} f(x_k)\cos jx_k \\ b_j &= \frac{2}{n+1}\sum_{k=0}^{n} f(x_k)\sin jx_k \end{aligned} \qquad (249)$$

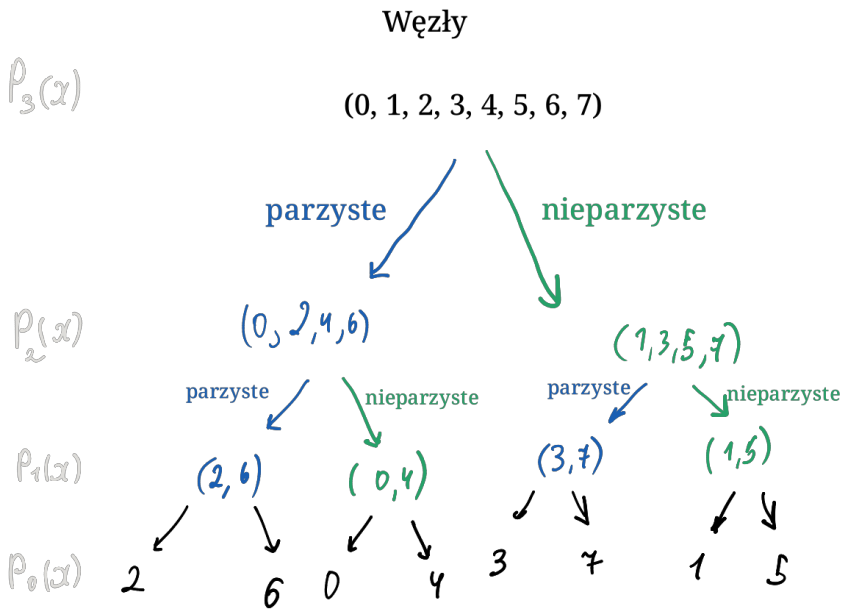
4.3.1 Szybka transformata Fouriera (FFT)

Wyznaczanie współczynników wykładniczego wielomianu interpolacyjnego c_j ze wzoru (247) wymaga n wymagań i n mnożeń na każdy współczynnik. Zatem złożoność obliczeniowa wyznaczania wszystkich współczynników jest rzędu $O(n^2)$. Złożoność tę można jednak zmniejszyć do rzędu $O(n \log n)$ stosując algorytm szybkiej transformaty Fouriera (**FFT** – **F**ast **F**ourier **T**ransform). Szczególnie wygodną formę przyjmuje algorytm FFT dla liczby węzłów będącej potęgą dwójki. Istotę algorytmu jest podział w każdym kroku węzłów na parzyste i nieparzyste i tworzenie osobno wykładniczych wielomianów interpolacyjnych dla obu tych grup węzłów. Współczynniki wykładniczego wielomianu interpolacyjnego łączącego obie te grupy węzłów wyrażają się łatwo za pomocą współczynników wielomianów interpolujących każdą z tych grup z osobna.

Kresem tych podziałów na grupy parzystych i nieparzystych węzłów jest otrzymanie pojedynczego węzła, który jest oczywiście interpolowany przez wartość funkcji w tym węźle, czyli jest postaci:

$$P_0(x) = c_0(e^{ix})^0 = c_0 = f(x_0) \qquad (250)$$

gdzie $x_{(0)}$ jest tym pojedynczym węzłem interpolacji



Rysunek 3: rysunek

Zależność rekurencyjną pomiędzy współczynnikami tych wielomianów interpolacyjnych określa poniższe twierdzenie

Twierdzenie 4.5

Niech $p(x), q(x)$ będą wykładniczymi wielomianami interpolacyjnymi interpolującymi odpowiednio parzyste i nieparzyste węzły postaci

$$p(x) = \sum_{j=0}^{n-1} \alpha_j (e^{ix})^j \qquad q(x) = \sum_{j=0}^{n-1} \beta_j (e^{ix})^j \tag{251}$$

Wówczas wielomian wykładniczy interpolujący wszystkie te węzły ma postać:

$$P(x) = \sum_{j=0}^{2n-1} \gamma_j (e^{ix})^j \tag{252}$$

gdzie

$$\begin{aligned} \text{dla } 0 \le j \le n-1 \quad \gamma_j &= \frac{1}{2}\alpha_j + \frac{1}{2} e^{-\frac{\pi i j}{n}} \beta_j \\ \gamma_{n+j} &= \frac{1}{2}\gamma_j - \frac{1}{2} e^{-\frac{\pi i j}{n}} \beta_j \end{aligned} \tag{253}$$

Przykład 4.4

Wyznaczyć postać wielomianu interpolacyjnego dla funkcji f interpolującą ją w dwóch punktach ($n = 1$)
Mamy - $x_0 = \frac{2 \cdot 0 \cdot \pi}{2} = 0$, - $x_1 = \frac{2 \cdot 1 \cdot \pi}{2} = \pi$
Wielomian $p(x) = f(x_0) = f(0) = \alpha_0$ interpoluje zatem funkcję f w węźle x_0, a wielomian $q(x) = f(x_1) = f(\pi) = \beta_0$ interpoluje funkcję f w węźle x_1.
Na podstawie tych dwóch wielomianów możemy skonstruować wielomian P interpolujący funkcję f w obu tych węzłach. Mamy:o

$$\gamma_0 = \frac{1}{2}\alpha_0 + \frac{1}{2}e^{-\frac{\pi i 0}{n}}\beta_0 = \frac{1}{2}(f(0) + f(\pi)) \qquad (254)$$

$$\gamma_1 = \gamma_{1+0} = \frac{1}{2}\alpha_0 - \frac{1}{2}e^{-\frac{\pi i 1}{n}}\beta_0 = \frac{1}{2}\alpha_0 - \frac{1}{2}\beta_0 = \frac{1}{2}(f(0) - f(\pi)) \qquad (255)$$

A wielomian interpolacyjny ma zatem postać

$$P(x) = \frac{1}{2}(f(0) + f(\pi)) + \frac{1}{2}(f(0) - f(\pi))e^{ix} \qquad (256)$$

2025-06-17

4.4 Aproksymacja średniokwadratowa

Niech $p(x) \geq 0$ będzie funkcją wagową taką, że

$$\int_a^b p(x)\, dx < \infty. \qquad (257)$$

Możemy wówczas zdefiniować iloczyn skalarny funkcji f i g za pomocą wzoru

$$(f, g) = \int_a^b p(x)f(x)g(x)\, dx \qquad (258)$$

Oraz normę funkcji f jako $\|f\| = \sqrt{(f, f)}$

Zadaniem aproksymacji jest znalezienie takiej funkcji $h^* \in U$, (gdzie U jest pewnym podzbiorem funkcji wśród których poszukujemy aproksymacji funkcji f), że $\|f - h^*\| = \min_{h \in U} \|f - h\|$.

Funkcję h^* nazywamy elementem optymalnym (najlepszym przybliżeniem) dla funkcji f względem zbioru U. Dla danej funkcji f istnieje dokładnie jeden element optymalny.

Wielkość $\varepsilon_U(f) = \|f - h^*\|$ nazywamy błędem aproksymacji.

Niech ciąg f_i będzie bazą funkcji z U. Wtedy element optymalny $h^* \in U$ dla funkcji f można zapisać w postaci

$$h^* = \sum_{i=0}^{n} \alpha_i f_i \qquad (259)$$

Współczynniki α_i elementu optymalnego znajdujemy rozwiązując układ równań liniowych postaci:

$$\sum_{i=0}^{n} \alpha_i (f_i, f_j) = (f, f_j) \quad \text{dla } j = 0, 1, \ldots, n \qquad (260)$$

Macierz A tego układu równań jest symetryczna bo $(f_i, f_j) = (f_j, f_i)$

Przykład 4.5

Znajdź element optymalny dla $f(x) = |x|$ w przedziale $[-1, 1]$ dla U – wielomianów co najwyżej 3-go stopnia z bazą $f_0(x) = 1, f_1(x) = x, f_2(x) = x^2$ i $p(x) \equiv 1$.
Mamy wtedy

$$(f_i, f_j) = \int_{-1}^{1} x^{i+j}\, dx = \frac{x^{i+j+1}}{i+j+1}\Big|_{-1}^{1} = \begin{cases} 0, & i+j \text{ nieparzyste} \\ \frac{2}{i+j+1}, & i+j \text{ parzyste} \end{cases} \qquad (261)$$

$$(f, f_i) = \int_{-1}^{1} |x| x^i\, dx = \begin{cases} 0, & i \text{ nieparzyste} \\ \frac{2}{i+2}, & i \text{ parzyste} \end{cases} \qquad (262)$$

Układ równań do wyznaczenia α_i ma postać

$$\begin{cases} \alpha_0(f_0, f_0) + \alpha_1(f_1, f_0) + \alpha_2(f_2, f_0) = (f, f_0) \\ \alpha_0(f_0, f_1) + \alpha_1(f_1, f_1) + \alpha_2(f_2, f_1) = (f, f_1) \\ \alpha_0(f_0, f_2) + \alpha_1(f_1, f_2) + \alpha_2(f_2, f_2) = (f, f_2) \end{cases} \qquad (263)$$

$$\begin{cases} 2\alpha_0 + \frac{2}{3}\alpha_2 = 1 \\ \frac{2}{3}\alpha_1 = 0 \\ \frac{2}{3}\alpha_0 + \frac{2}{5}\alpha_2 = \frac{1}{2} \end{cases} \qquad (264)$$

Rozwiązując ten układ równań otrzymujemy $\alpha_0 = \frac{3}{16}, \alpha_1 = 0, \alpha_2 = \frac{15}{16}$
czyli elementem optymalnym dla $f(x) = |x|$ jest $h^*(x) = \frac{3}{16} + \frac{15}{16}x^2$

4.5 Metoda najmniejszych kwadratów

Zdarza się, iż nie dysponujemy postacią aproksymowanej funkcji f, a jedynie jej wartościami w pewnych punktach x_i.

Poszukujemy wówczas funkcji $f \in U$, gdzie $f(x) = \sum_{j=0}^{n} \alpha_j f_j(x)$ takiej, że $\varphi(\alpha_0, \alpha_1, \ldots, \alpha_n) = \sum_{i=0}^{m} (f(x_i) - y_i)^2$ osiąga minimum.

Stosujemy wówczas metodę najmniejszych kwadratów w której postępujemy analogicznie jak w zwykłej aproksymacji, czyli znajdujemy współczynniki α_i z układu równań liniowych

$$\sum_{j=0}^{n} \alpha_j (f_j, f_i) = (y, f_i) \quad \text{dla } i = 0, 1, \ldots, n \tag{265}$$

przy czym iloczyny skalarne są zdefiniowane jako:

$$(f_i, f_j) = \sum_{k=0}^{m} f_i(x_k) f_j(x_k) \tag{266}$$

$$(y, f_i) = \sum_{k=0}^{m} y_k f_i(x_k) \tag{267}$$

Metoda najmniejszych kwadratów ma dokładnie jedno rozwiązanie.

Przykład 4.6

Znajdź funkcję liniową najlepiej dopasowaną do danych (w sensie metody najmniejszych kwadratów)

x_i	1	3	4	6	7
y_i	$-2{,}1$	$-0{,}9$	$-0{,}6$	$0{,}6$	$0{,}9$

(268)

Mamy $f_0(x) = 1$ i $f_1(x) = x$ i poszukujemy funkcji $f(x) = \alpha_0 + \alpha_1 x$, gdzie

$$\begin{cases} \alpha_0(f_0, f_0) + \alpha_1(f_1, f_0) = (y, f_0) \\ \alpha_0(f_0, f_1) + \alpha_1(f_1, f_1) = (y, f_1) \end{cases} \tag{269}$$

x	y	$f_0(x_i)$	$f_1(x_i)$	$f_1(x_i)f_1(x_i)$	$yf_1(x_i)$
1	-2,1	1	1	1	-2,1

3	-0,9	1	3	3	-2,7
4	-0,6	1	4	16	-2,4
6	0,6	1	6	36	3,6
7	0,9	1	7	49	6,3
$\sum$	-2,1	5	21	111	2,7
=	(y, f_0)	(f_0, f_0)	(f_0, f_1)	(f_1, f_1)	(y, f_1)

Czyli układ równań ma postać

$$\begin{cases} 5\alpha_0 + 21\alpha_1 = -2{,}1 \\ 21\alpha_0 + 111\alpha_1 = 2{,}7 \end{cases} \tag{270}$$

Rozwiązaniem tego układu równań są $\alpha_0 = -2{,}542, \alpha_1 = 0{,}5053$
Zatem elementem optymalnym jest funkcja

$$f(x) = -2{,}542 + 0{,}5053x \tag{271}$$

5 Różniczkowanie i całkowanie numeryczne

5.1 Różniczkowanie numeryczne

pomijamy

5.2 Całkowanie numeryczne

W przypadku całkowania numerycznego postępujemy analogicznie jak w przypadku różniczkowania. Stosujemy najpierw interpolację Lagrange'a a następnie całkujemy otrzymany wielomian interpolacyjny. Dostajemy wówczas:

$$\int_a^b f(x)\,dx \approx \int_a^b w(x)\,dx = \sum_{i=0}^{n} \left(f(x_i) \cdot \int_a^b l_i(x)\,dx \right) = \sum_{i=0}^{n} A_i f(x_i) \tag{272}$$

Wzory w których wartość całki przybliżamy poprzez sumę iloczynów wartości funkcji w węzłach pomnożone przez pewne współczynniki A_i nazywamy kwadraturami. Jeśli współczynniki A_i mają taką

postać, jak we wzorze powyżej, to taką kwadraturę nazywamy kwadraturą Newtona-Cotesa. Wielkości błędów w kwadraturach Newtona-Cotesa wyrażają się wzorem $\frac{f^{(n+1)}(\xi)}{(n+1)!}\int_a^b \prod_{i=0}^{n}(x-x_i)\,dx$.

dla pewnego $\xi \in (a;b)$

Biorąc we wzorze (5.2.5) $n=1$ (czyli 2 węzły interpolacji) mamy:

$$A_0 = \int_a^b l_0(x)\,dx = \int_a^b \frac{x-b}{a-b}\,dA = \frac{(x-b)^2}{2(a-b)}\Big|_a^b = \frac{1}{2}(b-a) \qquad (273)$$

$$A_1 = \int_a^b l_1(x)\,dx = \int_a^b \frac{x-a}{b-a}\,dx = \frac{(x-a)^2}{2(b-a)}\Big|_a^b = \frac{1}{2}(b-a) \qquad (274)$$

$$\frac{1}{2}(b-a)(fa) + \frac{1}{2}(b-a)f(b) = \frac{1}{2}(b-a)(f(a)+f(b)) \qquad (275)$$

i otrzymujemy tzw. wzór trapezów

$$\int_a^b f(x)\,dx \approx \frac{1}{2}(b-a)(f(a)+f(b)) \qquad (276)$$

Błędem tej kwadratury jest wartość $-\frac{1}{12}(b-a)^3 f''(\xi)$ dla pewnego $\xi \in (a;b)$

Aby poprawić dokładność wzoru trapezów cały rozważany przedział $(a;b)$ można podzielić na podprzedziały punktami x_i, tak że $a = x_0 < x_1 < \ldots < x_{n-1} < x_n = b$ a następnie do każdego z tak tworzonych podprzedziałów zastosować wzór trapezów (i zsumować obliczone całki na każdym z podprzedziałów). Otrzymujemy wówczas tzw. złożony wzór trapezów

$$\int_a^b f(x)\,dx = \sum_{i=1}^{n}\int_{x_{i-1}}^{x_i} f(x)\,dx \approx \frac{1}{2}\sum_{i=1}^{n}(x_i - x_{i-1})[f(x_{i-1}) + f(x_i)] \qquad (277)$$

Jeśli przyjmiemy, że punkty x_i są równo rozłożone w przedziale (a,b), czyli podprzedziały są jednakowej długości równej $h = \frac{b-a}{n}$, a $x_i = a + ih$, to otrzymujemy następującą postać złożonego wzoru trapezów.

$$\int_a^b f(x)\,dx \approx \frac{1}{2}h[f(a) + 2\sum_{i=1}^{n-1} f(a+ih) + f(b)] \qquad (278)$$

I błąd będzie równy

$$-\frac{1}{12n^2}(b-a)^3 f''(\xi) \quad \text{dla } \xi \in (a,b) \qquad (279)$$

Biorąc we wzorze (5.2.5) $n = 2$ (czyli 3 równoodległe węzły interpolacji $x_0 = 0, x_1 = \frac{a+b}{2}, x_2 = b$), mamy

$$A_0 = \int_a^b l_0(x)\ dx = \frac{1}{6}(b-a) \qquad (280)$$

$$A_1 = \int_a^b l_1(x)\ dx = \frac{4}{6}(b-a) \qquad (281)$$

$$A_2 = \int_a^b l_2(x)\ dx = \frac{1}{6}(b-a) \qquad (282)$$

i otrzymujemy tzw. wzór Simpsona:

$$\int_a^b f(x)\ dx \approx \frac{1}{6}(b-a)[f(a) + 4f(\frac{a+b}{2}) + f(b)] \qquad (283)$$

Błędem tej kwadratury jest wartość $-\frac{1}{2880}(b-a)^5 f^{(4)}(\xi)$ dla pewnego $\xi \in (a;b)$

Podobnie jak poprzednio, aby poprawić dokładność wzoru Simpsona cały rozważamy przedział $(a;b)$ można podzielić na podprzedziały punktami x_i, a następnie do każdej z trójek punktów x_i zastosować wzór Simpsona (i zsumować otrzymane całki). Otrzymujemy wówczas tzw. złożony wzór Simpsona. Jeśli przyjmiemy, że punkty x_i są równo rozłożone w przedziale (a, b), czyli podprzedziały są jednakowej długości $h = \frac{b-a}{2n}$, a $x_i = a + ih$, to otrzymamy następującą postać złożonego wzoru Simpsona:

$$\int_a^b f(x)\ dx \approx \frac{1}{3}h\left[f(x_0) + 4\sum_{i=1}^{\infty} f(x_{2i-1}) + 2\sum_{i=1}^{n-1} f(x_{2i}) + f(x_{2n})\right] \qquad (284)$$

i błąd $-\frac{1}{2880n^4}(b-a)^5 f^{(4)}(\xi)$ dla $\xi \in (a;b)$

Współczynniki A_i w kwadraturach Newtona-Cotesa nie muszą być wyznaczane poprzez całkowanie wielomianów $l_i(x)$. Można je także wyznaczyć rozwiązując następujący układ równań liniowych dla węzłów x_i i

$$\int_a^b x^j\ dx = \sum_{i=0}^{n} A_i x_i^j, \quad 0 \le j \le n \qquad (285)$$

Achtung! 7

Egzamin na godzinę, 3 pytania. Zasadniczo pytania 3 typów:

- O konkretną metodę (np. opisz metodę Richardsona)
    1. Powinniśmy napisać do czego służy (służy do rozwiązania układów równań liniowych)

2. Umiejscowienie metody wśród innych metod (należy do metod iteracyjnych, można też napisać krótko na czym polegają iteracyjne)
3. Wzór metody
4. Podsumowanie (najwolniejsza)

- Cała grupa metod:

1. Z czego powstała (geneza metod)
2. Wyszczególnienie metod
3. Wzory bez szczegółów
4. Uszeregować: szybkość, …

- Większe zagadnienia:

1. Reprezentacja liczb w komputerze:
2. system dwójkowy
3. bit znaku cecha mantysa
4. wiąże się z błędami reprezentacji, na czym one polegają

Na pierwszym terminie nie będzie materiału z dzisiaj