Metody numeryczne dla fizyków

Krzysztof Golec–Biernat

(1 maja 2015)

Wersja robocza nie do dystrybucji

Kraków
2007-15

Spis treści

Wykład 1

Liczby i ich reprezentacje

1.1 Systemy liczbowe

Przypomnijmy zapis liczby całkowitej w systemie dziesiętnym, na przykład

$$(1234)_{10} = 4\cdot 10^0 + 3\cdot 10^1 + 2\cdot 10^3 + 1\cdot 10^4 .$$

Do zapisu używamy dziesięć cyfr $\{0,1,2,\ldots,9\}$, a 10 to **podstawa** systemu liczbowego. Wartość cyfry zależy od pozycji w liczbie.

W ogólności, dowolną liczbę całkowitą można zapisać w systemie pozycyjnym o podstawie p przy pomocy cyfr $c_i \in \{0,1,\ldots,p-1\}$ w następujacy sposób

$$\boxed{(c_k \ldots c_1 c_0)_p = c_0\cdot p^0 + c_1\cdot p + \ldots + c_k\cdot p^k} \tag{1.1}$$

Podobnie, dowolną liczbę ułamkową z przedziału $(0,1)$ możemy zapisać przy pomocy (na ogół nieskończonego) rozwinięcia w ujemnych potęgach podstawy p,

$$\boxed{(0.c_{-1}c_{-2}\ldots c_{-n})_p = c_{-1}\cdot p^{-1} + c_{-2}\cdot p^{-2} + \ldots + c_{-n}\cdot p^{-n}} \tag{1.2}$$

1.2 System dwójkowy

Szczególnie ważną rolę odgrywa **system dwójkowy** o podstawie $p=2$ i cyfrach $c_i \in \{0,1\}$. Znaczenie tego systemu dla komputerów wynika z faktu, że podstawową jednostką informacji jest **bit** przyjmujący jedną z dwóch możliwych wartości: 0 lub 1. Przyjęło się nazywać

1 bajt = 8 bitów

Przykładowy zapis liczby całkowitej w systemie dwójkowym to

$$(11010)_2 = 0\cdot 2^0 + 1\cdot 2^1 + 0\cdot 2^2 + 1\cdot 2^3 + 1\cdot 2^4 = 2+8+16 = (26)_{10} .$$

Podobnie dla liczby ułamkowej

$$(0.1101)_2 = 1\cdot\tfrac{1}{2}+1\cdot\tfrac{1}{4}+0\cdot\tfrac{1}{8}+1\cdot\tfrac{1}{16} = \left(\tfrac{13}{16}\right)_{10}.$$

Załóżmy, że dysponujemy n bitami do zapisu liczby całkowitej. Pozwalają one zapisać 2^n liczb. Najmniejszą liczbą całkowitą dodatnią zapisaną przy pomocy n bitów jest zero

$$(00\ldots00)_2 = 0\,, \tag{1.3}$$

natomiast największą liczbą jest

$$(11\ldots11)_2 = 1\cdot2^0+1\cdot2^1+\ldots+1\cdot2^{n-1} = 2^n-1\,. \tag{1.4}$$

Wykorzystaliśmy tutaj wzór na sumę skończonej liczby wyrazów szeregu geometrycznego o ilorazie $a\neq1$:

$$\boxed{1+a+a^2+\ldots+a^{N-1} = \frac{1-a^N}{1-a}} \tag{1.5}$$

Podsumowując, dysponując n bitami możemy zapisać 2^n liczb całkowitych z przedziału

$$[0,\,2^n-1]\,. \tag{1.6}$$

Podobnie, przy pomocy n bitów możemy zapisać 2^n-1 liczb ułamkowych z przedziału $(0,1)$. Najmniejszą z nich jest

$$(0.00\ldots01)_2 = 1\cdot2^{-n} = \frac{1}{2^n} \tag{1.7}$$

natomiast największa to

$$(0.11\ldots11)_2 \;=\; 1\cdot2^{-1}+1\cdot2^{-2}+\ldots+1\cdot2^{-n} = 1-\frac{1}{2^n}\,. \tag{1.8}$$

Podsumowując, przy pomocy n bitów zapiszemy (2^n-1) (brak zera) liczb z przedziału

$$\left[\frac{1}{2^n},\,1-\frac{1}{2^n}\right], \tag{1.9}$$

Wygodnie jest zapamiętać następująca tabelkę ułatwiającą zmianę zapisu liczb w systemie dwójkowym na system dziesiętny.

n	2^n	2^{-n}
0	1	1
1	2	1/2
2	4	1/4
3	8	1/8
4	16	1/16
5	32	1/32
6	64	1/64
7	128	1/128
8	256	1/256
9	512	1/512
10	1024	1/1024
11	2048	1/2048
12	5096	1/5096

1.3 Konwersja do systemu dwójkowego

1.3.1 Konwersja liczby całkowitej

Konwersję odwrotną z systemu dziesiętnego na dwójkowy dla liczb całkowitych realizuje się zauważając, że zachodzi

$$x = c_0 + c_1 \cdot 2^1 + \ldots + c_k \cdot 2^k = c_0 + 2 \cdot (c_1 + \ldots + c_k \cdot 2^{k-1}) \tag{1.10}$$

Tak więc c_0 to reszta z dzielenia x przez 2. Podobnie c_1 jest resztą z dzielenia $(c_1 + \ldots + c_k \cdot 2^{k-1})$ przez 2, itd.

Przykładowo, konwertując liczbę 30 otrzymamy

$$\begin{array}{r|l} 30 & = 2 \cdot 15 + 0 \\ 15 & = 2 \cdot 7 + 1 \\ 7 & = 2 \cdot 3 + 1 \\ 3 & = 2 \cdot 1 + 1 \\ 1 & = 2 \cdot 0 + 1 \end{array}$$

Odczytując reszty **od dołu do góry** otrzymujemy:

$$30 = (11110)_2 \,. \tag{1.11}$$

1.3.2 Konwersja liczby ułamkowej

Dla liczby ułamkowej,

$$x = c_{-1} \cdot 2^{-1} + c_{-2} \cdot 2^{-2} + \ldots + c_{-n} \cdot 2^{-n} \tag{1.12}$$

po pomnożeniu przez dwa, otrzymujemy

$$2x = c_{-1} + \underbrace{(c_{-2}\cdot 2^{-1} + \ldots + c_{-n}\cdot 2^{-n+1})}_{r} \tag{1.13}$$

Jak łatwo zauważyć c_{-1} jest częścią całkowitą liczby $2x$ (równą 0 lub 1) natomiast $r < 1$ jest częścią ułamkową. Aby wydobyć c_{-2} mnożymy wtedy r przez dwa, itd.

Przykładowo, dla 0.375 otrzymujemy

$$\begin{array}{r|l} 0.375 & \times 2 = 0.750 \\ 0.750 & \times 2 = 1.500 \\ 0.500 & \times 2 = 1.000 \end{array}$$

Odczytując części całkowite liczb po prawej stronie równości **od góry do dołu**, dostajemy

$$0.375 = (0.011)_2\,. \tag{1.14}$$

Otrzymana reprezentacja dwójkowa jest skończona. Rozwinięcie dwójkowe ułamków może być jednak nieskończone, podobnie jak w systemie dziesiętnym.

Łącząc dwie przedstawione metody konwersji, możemy znaleźć reprezentację dwójkową dowolnej liczby rzeczywistej, na przykład:

$$30.375 = 30 + 0.375 = (11110.011)_2 \tag{1.15}$$

1.4 Reprezentacja liczb całkowitych

Załóżmy, że przeznaczamy n bitów do zapisu liczby całkowitej w komputerze. Pierwszy z nich przeznaczamy na znak liczby:

$$(0) = (+) \qquad\qquad (1) = (-) \tag{1.16}$$

Pozostałe $(n-1)$ bitów posłuży do zapisu liczb całkowitych C z przedziału

$$C \in [\,-2^{n-1}+1\,,\,2^{n-1}-1\,]\,. \tag{1.17}$$

Zauważmy, że postępując w ten sposób zero jest reprezentowane dwukrotnie ze względu na znak:

$$0 = (0)\underbrace{0\ldots0}_{n-1} \qquad \text{oraz} \qquad 0 = (1)\underbrace{0\ldots0}_{n-1} \tag{1.18}$$

Przyjmując tylko pierwszy sposób zapisu unikamy tej niejednozaczności zwalniając jednocześnie miejsce dla dodatkowej liczby, na przykład 2^{n-1}. Tak więc, przy pomocy n bitów można zapisać 2^n liczb całkowitych z przedziału

$$\boxed{C \in [\,-2^{n-1}+1\,,\,2^{n-1}\,]} \tag{1.19}$$

Podkreślmy, że liczby całkowite są reprezentowane *dokładnie*.

W praktyce, jednoznaczną reprezentację zera można zrealizować w następujący sposób. Przeznaczmy n bitów na zapis 2^n nieujemnych liczb całkowitych C' z przedziału

$$C' \in [\,0\,, 2^n - 1\,]\,. \tag{1.20}$$

Przesuńmy następnie wszystkie liczby całkowite odejmując od każdej z nich liczbę $E = 2^{n-1} - 1$. Otrzymujemy liczby

$$C = C' - E \tag{1.21}$$

z przedziału (1.19). Zauważmy, że zero nie jest już reprezentowane przez pierwszy ciąg bitów w reprezentacji (1.18). Jego reprezentacja odpowiada teraz dwójkowej reprezentacji liczby E.

Współczesne procesory przeznaczają w pojedynczej precyzji **4 bajty czyli 32 bity** do zapisu liczby całkowitej lub **8 bajtów czyli 64 bity** w podwójnej precyzji. Poniższa tabelka pokazuje zakres liczb całkowitych w tych przypadkach

Precyzja	Liczba bitów	Zakres	Uwagi
Pojedyncza	32	$[-2^{31}+1\,, 2^{31}]$	$2^{31} \sim 10^9$
Podwójna	64	$[-2^{63}+1\,, 2^{63}]$	$2^{63} \sim 10^{19}$

1.5 Reprezentacja zmiennoprzecinkowa

Przedyskutujemy teraz zapis liczb rzeczywistych w komputerze. Najbardziej efektywny zapis wykorzystuje reprezentację **zmiennoprzecinkową**. W reprezentacji tej nie ustala się maksymalnej liczby cyfr po przecinku przy zapisie liczby.

Przykładowo, rozważmy liczbę $x = 0.00045$. Możemy ja zapisać w następujący sposób

$$x = 45 \cdot 10^{-5} = 4.5 \cdot 10^{-4} = 0.45 \cdot 10^{-3}\,.$$

Liczba stojąca przez potęgą dziesiątki to **mantysa** M, natomiast wykładnik potęgi to **cecha** C. Dowolną liczbę rzeczywistą można więc zapisać w postaci

$$x = \pm M \cdot 10^C\,. \tag{1.22}$$

Przyjmujemy konwencję, że mantysa należy do przedziału

$$M \in [1,10)\,, \tag{1.23}$$

natomiast cecha C jest *liczbą całkowitą* (dodatnią ujemna lub równą zeru). Dla przykładowej liczby x: $M = 4.5$ i $C = -4$.

Przyjmując za podstawę $p = 2$ dostajemy analogiczny zapis w systemie dwójkowym

$$\boxed{x = \pm M \cdot 2^C,} \tag{1.24}$$

gdzie tym razem dla mantysy zachodzi:

$$\boxed{M \in [1,2)} \tag{1.25}$$

1.5.1 Podział bitów

Reprezentacja (1.24) jest podstawą zapisu liczb rzeczywistych w komputerze. W tym celu dzielimy n **bitów** przeznaczonych do zapisu liczby rzeczywistej,

$$x = S \times M \times 2^C, \tag{1.26}$$

w następujący sposób

$$n = 1 + n_M + n_C. \tag{1.27}$$

W powyższej sumie

- pierwszy bit przeznaczamy na znak liczby $S = \pm 1$,
- następne n_M bitów służą do zapisu *części ułamkowej* F mantysy

  $$M = 1 + F. \tag{1.28}$$

  Ze względu na zawsze występująca jedynkę nie musimy jej kodować. Część ułamkowa mantysy należy więc do przedziału

  $$F \in \left[\frac{1}{2^{n_M}}, 1 - \frac{1}{2^{n_M}}\right] \tag{1.29}$$

- pozostałe n_C bitów przeznaczamy na zapis *cechy C wraz ze znakiem.* Zgodnie z warunkiem (1.19)

  $$C \in [-2^{n_C-1} + 1\,, 2^{n_C-1}]. \tag{1.30}$$

  Mantysę C można zapisać wykorzystując przesuniecie $C = C' - E$, gdzie $E = 2^{n_C-1} - 1$. Kodujemy wtedy liczbę całkowitą $C' \in [0\,, 2^{n_C} - 1]$.

Oczywistym jest, że prawie wszystkie liczby rzeczywiste są reprezentowane w *przybliżeniu.*

Współczesne procesory reprezentują liczby zmiennoprzecinkowe w konwencji zgodnej ze standardem IEEE Standard 754-1985. Podział bitów dla cechy i mantysy dla tego standardu podany jest w tabelce.

Precyzja	Liczba bitów	n_M	n_C	Zakres C
Pojedyncza	32	23	8	[-127 , 128]
Podwójna	64	52	11	[-1023 , 1024]

Poniżej podajemy kilka zapisów liczb w pojedynczej precyzji.

$$0(\underbrace{0\ldots0}_{23\ zera})(00000000) \;=\; +1.0\times 2^0 = 1.0$$

$$0(\underbrace{0\ldots0}_{23\ zera})(00000001) \;=\; +1.0\times 2^1 = 2.0$$

$$1(1\,\underbrace{0\ldots0}_{22\ zera})(10000001) \;=\; -(1.0+\tfrac{1}{2})\times 2^{-1} = -0.75$$

Wykład 2

Błędy i ich źródła

2.1 Dokładność zapisu

Miarą przybliżonego zapisu liczby rzeczywistej jest epsilon maszynowy, zdefiniowany w następujący sposób.

Rozważmy *najmniejszą* liczbę możliwą do zapisu w komputerze *większa od* 1 i oznaczny ją przez x. **Epsilon maszynowy** to różnica:

$$\epsilon_m = x - 1\,. \tag{2.1}$$

W przykładzie z podziałem $8 = 1+3+4$ bitów najmniejszą zapisaną liczbą większą od 1 jest

$$(0)001(0)000 = (1 + 0\cdot 2^{-1} + 0\cdot 2^{-2} + 1\cdot 2^{-3})\cdot 2^0 = 1 + \tfrac{1}{8} \tag{2.2}$$

i stąd $\epsilon_m = 2^{-3} = 0.125$.

W ogólności, przeznaczając n_M bitów do zapisu mantysy (z wyłączeniem znaku), zachodzi

$$\boxed{\epsilon_m = \frac{1}{2^{n_M}}} \tag{2.3}$$

Liczba

$$p = n_M \tag{2.4}$$

nazywana jest **precyzją** i jest liczbą bitów przeznaczonych do zapisu mantysy z wyłaczeniem znaku.

Zwiększając liczbę bitów mantysy n_M zwiększamy precyzję. Odbywa się to kosztem zmniejszenia maksymalnej liczby możliwej do zapisania, gdyż przy ustalonej całkowitej liczbie bitów n, mniej bitów można przeznaczyć do zapisu cechy:

$$n_C = n - 1 - n_M\,. \tag{2.5}$$

Jest to ilustracja wymienności pomiędzy *dokładnością* a *zakresem* reprezentowanych liczb zmiennoprzecinkowych w komputerze.

Tabelka poniżej precyzję i epsilon maszynowy dla używanych powszechnie precyzji zapisu liczby rzeczywistej.

Precyzja	Liczba bitów	n_M	p	ϵ_m
Pojedyncza	32	23	23	$2^{-23} \sim 10^{-7}$
Podwójna	64	52	52	$2^{-52} \sim 10^{-15}$

2.2 Błąd bezwględny i względny

Jak widzimy, na ogół dowolona liczba zmiennoprzecinkowa x nie może być reprezentowana w komputerze dokładnie.

Jeżeli najbliższą liczbą reprezentującą x w komputerze jest x', to **błąd bezwględny** zapisu liczby x jest zdefiniowany jako różnica

$$\boxed{\delta x = x' - x} \tag{2.6}$$

Błąd względny to stosunek błędu bezwzględnego do wartości dokładnej liczby

$$\boxed{\epsilon = \frac{x' - x}{x}} \tag{2.7}$$

Z powyższego wzoru wynika. że liczbę przybliżoną x' można traktować jako zaburzenie liczby dokładnej x przez czynnik multiplikatywny zawierający wartość błędu względnego ϵ,

$$x' = x\,(1+\epsilon). \tag{2.8}$$

Można pokazć, że moduł błędu względnego zapisu liczby zmiennoprzecinkowej jest mniejszy od epsilona maszynowego

$$\boxed{|\epsilon| \leqslant \epsilon_m} \tag{2.9}$$

2.2.1 Przyklad

Zapiszmy liczbę $0.48 = 12/25$ przy pomocy $8 = 1+3+4$ bitów. Epsilon maszynowy w tym przypadku to

$$\epsilon_m = \tfrac{1}{8} = 0.125\,. \tag{2.10}$$

W reprezentacji cecha-mantysa otrzymujemy

$$0.48 = 1.92/4 = (1.92)\cdot 2^{-2}.$$

Mantysa zapisana w systemie dwójkowym to

$$1.92 = (1.111010\ldots)_2\,. \tag{2.11}$$

W przyjętej reprezentacji pozostawiamy pierwsze trzy bity w rozwinięciu dwójkowym mantysy, stąd

$$0.48 \approx (0)111(1)010 = (1+1\cdot 2^{-1}+2^{-2}+2^{-3})\cdot 2^{-2} = \tfrac{15}{32} = 0.46875\,.$$

Jest to ilustracja *zaokrąglania*, polegającego na *odrzuceniu* cyfr w rozwinięciu dwójkowym mantysy wykraczających poza przyjętą liczbę bitów (w tym przypadku trzech). Powstały w ten sposób błąd względny to

$$|\epsilon| = \left|\frac{\frac{15}{32}-\frac{12}{25}}{\frac{12}{25}}\right| = \tfrac{4}{96} \approx 0.04\,.$$

Jak widać $|\epsilon| < \epsilon_m$, zgodnie z warunkiem (2.9).

2.3 Źródła błędów

Przy obliczeniach komputerowych mamy do czynienia z trzema rodzajami błędów.

2.3.1 Błędy wejściowe.

Są one spowodowane tym, że dane wprowadzane do komputera są obarczone błędem. Mogą to być dane eksperymentalne, które ze swej natury są obarczone błędem. Najczęściej jednak błędy te wynikają z zaokrąglania liczb rzeczywistych, reprezentowanych w komputerze przez słowa binarne o skończonej długości.

2.3.2 Błędy obcięcia.

Błędy obcięcia powstają podczas obliczeń wymagających nieskończonej liczby działań, np. podczas nieskończonych sumowań lub ścisłych przejść granicznych.

Dla przykładu, obliczając wartość funkcji przy pomocy rozwinięcia Taylora,

$$\boxed{f(x) = f(x_0) + \frac{f'(x_0)}{1!}\,h + \frac{f''(x_0)}{2!}\,h^2 + \ldots + \frac{f^{(N)}(x_0)}{N!}\,h^N + R_{N+1}(\xi)} \tag{2.12}$$

gdzie $h = x - x_0$, popełniamy błąd zadany przez odrzucaną resztę

$$R_{N+1}(\xi) = \frac{f^{(N+1)}(\xi)}{(N+1)!}\,h^{N+1}\,, \tag{2.13}$$

gdzie punkt $\xi \in [min\{x_0, x\}, max\{x_0, x\}]$.

W szczególności, dla kilku podstawowych funkcji używamy następujących przybliżonych rozwinięć wokół $x_0 = 0$:

$$\exp x \approx 1 + x + \frac{x^2}{2!} + \ldots + \frac{x^N}{N!} \qquad (-\infty < x < \infty) \quad (2.14)$$

$$\sin x \approx x - \frac{x^3}{3!} + \frac{x^5}{5!} + \ldots + (-1)^N \frac{x^{2N+1}}{(2N+1)!} \qquad (-\infty < x < \infty) \quad (2.15)$$

$$\cos x \approx 1 - \frac{x^2}{2!} + \frac{x^4}{4!} + \ldots + (-1)^N \frac{x^{2N}}{(2N)!} \qquad (-\infty < x < \infty) \quad (2.16)$$

$$\frac{1}{1-x} \approx 1 + x + x^2 + \ldots + x^N, \qquad (-1 < x < 1) \quad (2.17)$$

Ponadto, do obliczenia logarytmu dowolnej dodatniej liczby rzeczywistej dodatniej stosujemy wzór

$$\ln \frac{1+x}{1-x} \approx 2\left(x + \frac{x^3}{3} + \frac{x^5}{5} + \ldots + \frac{x^{2N+1}}{2N+1}\right), \qquad (-1 < x < 1) \quad (2.18)$$

Zauważmy bowiem, że podstawiając $x \in (-1, 1)$, liczba

$$w = \frac{1+x}{1-x} \quad (2.19)$$

przebiega zakres wartości dziedziny logarytmu: $w \in (0, +\infty)$. Stąd metoda by licząc $\ln w$ najpierw odwrócić powyższą relację:

$$x = \frac{(w-1)}{(w+1)}, \quad (2.20)$$

a następnie podstawić ją po prawej stronie wzoru (2.18).

2.3.3 Błędy zaokrągleń.

To błędy wynikające z zaokrągleń wykonywanych w trakcie wielokrotnych działań arytmetycznych.

Oznaczmy przez $\mathrm{fl}(x)$ wynik obliczeń numerycznych różniący się od dokładnej wartości x na skutek błędów zaokrągleń. Zgodnie z (2.7) dla pojedynczej liczby mamy:

$$\mathrm{fl}(x) = x\,(1+\epsilon)\,. \quad (2.21)$$

W przypadku działań arytmetycznych obowiązuje:

Lemat Wilkinsona
Błędy zaokrągleń powstające przy wykonywaniu działań zmiennoprzecinkowych

są równoważne zastępczemu zaburzaniu liczb, na których wykonujemy działania. W przypadku pojedynczych działań zachodzi

$$\mathrm{fl}(x_1 \pm x_2) \;=\; x_1(1+\epsilon_1) \pm x_2(1+\epsilon_2) \tag{2.22}$$

$$\mathrm{fl}(x_1 \cdot x_2) \;=\; x_1(1+\epsilon_3)\cdot x_2 = x_1 \cdot x_2(1+\epsilon_3) \tag{2.23}$$

$$\mathrm{fl}(x_1/x_2) \;=\; x_1(1+\epsilon_4)/x_2 = x_1/(x_2(1+\epsilon_5))\,, \tag{2.24}$$

Wszystkie zaburzenia można wybrać tak by ich moduł był mniejszy od epsilona maszynowego

$$|\epsilon_i| < \epsilon_m\,. \tag{2.25}$$

2.3.4 Przyklad

Zilustrujmy lemat Wilkinsona dodając do siebie dwie liczby w reprezentacji jednobajtowej z podziałem $8 = 1+3+4$,

$$0.48 + 0.24 = \tfrac{12}{25} + \tfrac{12}{50} = 0.72\,.$$

Dostajemy

$$\begin{aligned} 0.48 = 1.92\cdot 2^{-2} &\;\approx\; (0)111(1)010 = (1+1\cdot 2^{-1}+2^{-2}+2^{-3})\cdot 2^{-2} = \tfrac{15}{32} \\ 0.24 = 1.92\cdot 2^{-3} &\;\approx\; (0)111(1)110 = (1+1\cdot 2^{-1}+2^{-2}+2^{-3})\cdot 2^{-3} = \tfrac{15}{64}\,. \end{aligned}$$

gdyż

$$1.92 = (1.111010\ldots)_2 \approx (1.111)_2\,.$$

Po dodaniu numerycznym otrzymujemy

$$\mathrm{fl}(0.48+0.24) = \tfrac{15}{32} + \tfrac{15}{64} = \tfrac{45}{64} = 1.40625\cdot 2^{-1} \approx (0)011(1)100 = \tfrac{11}{16} = 0.6875\,,$$

gdyż

$$1.40625 = (1.01101)_2 \approx (1.011)_2\,.$$

Zgodnie z lematem Wilkinsona, ten sam wynik można otrzymać zaburzając dodawane do siebie liczby

$$\tfrac{12}{25}(1+\epsilon_1) + \tfrac{12}{50}(1+\epsilon_2) = \tfrac{11}{16}\,,$$

skąd wynika relacja

$$2\epsilon_1 + \epsilon_2 = -\tfrac{13}{96}\,.$$

Wybór liczb ϵ_i nie więc jest jednoznaczny. Można je jednak wybrać tak by zachodziło: $|\epsilon_i| \leqslant \epsilon_m = 1/8$. Na przykład: $\epsilon_1 = -1/16$ oraz $\epsilon_2 = -1/96$.

2.4 Odejmowanie małych liczb

Obliczmy błąd względny różnicy dwóch liczb

$$|\epsilon| = \left| \frac{\mathrm{fl}(x_1 - x_2) - (x_1 - x_2)}{x_1 - x_2} . \right| \tag{2.26}$$

Zgodnie z lematem Wilkinsona

$$\mathrm{fl}(x_1 - x_2) = x_1(1+\epsilon_1) - x_2(1+\epsilon_2) . \tag{2.27}$$

Podstawiając do pierwszego wzoru, otrzymujemy

$$|\epsilon| = \left| \frac{x_1 \cdot \epsilon_1 - x_2 \cdot \epsilon_2}{x_1 - x_2} . \right| . \tag{2.28}$$

Załóżmy, ze dwie liczby różnią się mało od siebie

$$x_1 = x_2(1+\delta) , \qquad \delta \to 0 \tag{2.29}$$

Podstawiając powyżej, znajdujemy

$$|\epsilon| \simeq \frac{|\epsilon_1 - \epsilon_2|}{|\delta|} \to \infty \qquad \text{dla} \qquad \delta \to 0 . \tag{2.30}$$

Błąd względny odejmowania może być bardzo duży dla liczb niewiele różniących się od siebie. *Należy więc unikać odejmowania mało różniących się od siebie liczb.*

Nieograniczony wzrost (2.30) nie jest sprzeczny z warunkiem (2.25) w lemacie Wilkinsona, gdyż epsilony w tym lemacie to zaburzenia indywidualnych liczb, a nie błąd względny (2.26) wyniku działań arytmetycznych.

Jako przykład, obliczmy jeden z pierwiastków równania kwadratowego dla $b^2 \gg 4ac$,

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a} . \tag{2.31}$$

W tym celu należy wykorzystać równoważną postać

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \cdot \frac{(-b - \sqrt{b^2 - 4ac})}{(-b - \sqrt{b^2 - 4ac})} = \frac{-2c}{b + \sqrt{b^2 - 4ac}} , \tag{2.32}$$

gdyz dodajemy wtedy dwie wielkości tego samego rzędu w mianowniku, zamiast odejmować dwie bliskie liczby w liczniku.

Wykład 3

Algorytmy

Omówimy po krótce problem stabilności i efektywności algorytmów. Zagadnienie to dotyczy wpływu błędów zaokrągleń lub błędów obcięć na stabilność algorytmów numerycznych. Wynikające stąd problemy przedstawimy na przykładzie algorytmów realizowanych przy pomocy *relacji rekurencyjnych*.

3.1 Algorytmy stabilne

Przykładem algorytmu stabilnego jest relacja rekurencyjna służąca do obliczenia kolejnych przybliżeń liczby $\sqrt{2}$:

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{2}{x_n}\right), \qquad n \geqslant 0 \tag{3.1}$$

Zaczynają z $x_0 = 2$ otrzymujemy kolejne przybliżenia

$$x_1 = 1.5\,, \qquad x_2 = 1.416667\,, \qquad x_3 = 1.414216$$

Zakładając, że istnieje granicą tego ciągu otrzymujemy jako granicę $x_* = \sqrt{2}$, gdyż z relacji (3.1) wynika

$$x_* = \frac{x_*}{2} + \frac{1}{x_*} \qquad => \qquad x_* = \sqrt{2}\,. \tag{3.2}$$

Istnienie granicy wynika z faktu, że bład względny kolejnych przybliżeń szybko zmierza do zera. Zaczynając od $n = 1$, otrzymujmy

$$x_1 = \sqrt{2}(1+\epsilon)\,. \tag{3.3}$$

gdzie $\epsilon < 1$. W kolejnym przybliżeniu otrzymujemy

$$\begin{aligned} x_2 &= \frac{x_1}{2} + \frac{1}{x_1} = \frac{1}{\sqrt{2}}\left\{1+\epsilon+\frac{1}{1+\epsilon}\right\} \\ &= \frac{1}{\sqrt{2}}\left\{1+\epsilon+(1-\epsilon+\epsilon^2+\ldots)\right\} \approx \sqrt{2}\left\{1+\frac{\epsilon^2}{2}\right\} \end{aligned} \tag{3.4}$$

Teraz błąd względny jest kwadratem poprzedniego. W następnych krokach blad ten maleje jak kolejne potęgi ϵ^{2k} tak, że wystarczy niewiele iteracji do osiągniecia z góry założonej dokładności Algorytm znajdowania pierwiastka jest więc szybko zbieżny (efektywny).

Przykładem mało efektywnego algorytmu jest sposób obliczania liczby $\ln 2$ przy pomocy rozwinięcia w szereg Taylora

$$\ln(1+x) \approx x - \frac{x^2}{2} + \frac{x^3}{3} + \ldots + (-1)^{N-1}\frac{x^N}{N}\,, \tag{3.5}$$

dla $x = 1$. Błąd bezwględny to $|R_{N+1}| < 1/(N+1)$ i chcąc na przykład osiągnąc dokładność równą 10^{-8}, musimy wysumować $N \approx 10^8$ wyrazów. Znacznie lepiej jest tutaj skorzystać ze wzoru (2.18)

$$\ln\frac{1+x}{1-x} \approx 2\left(x + \frac{x^3}{3} + \frac{x^5}{5} + \ldots + \frac{x^{2N+1}}{2N+1}\right) \qquad (-1 < x < 1)$$

dla $x = 1/3$, który jest dużo szybciej zbieżny ze względu na szybko malejące kolejne nieparzyste potęgi tej liczby.

3.2 Algorytmy niestabilne

Rozważmy trójczłonową relację rekurencyjną

$$x_{n+2} = x_n - x_{n+1}\,, \qquad n \geqslant 0 \tag{3.6}$$

z warunkiem początkowym

$$x_0 = 1\,, \qquad x_1 = \frac{\sqrt{5}-1}{2}\,. \tag{3.7}$$

Łatwo sie przekonać, że rozwiązaniem tej rekurencji jest

$$x_n = \left(\frac{\sqrt{5}-1}{2}\right)^n\,. \tag{3.8}$$

W oczywisty sposób $x_n \to 0$ dla $n \to \infty$, gdyż $(\sqrt{5}-1)/2 \approx 0.618 < 1$ Obliczając jednak kolejne wartości x_n z warunku (3.6) dostajemy od pewnego n szybko rosnące wartości. Jest to spowodowane nieuniknionym błędem zaokrąglenia liczby niewymiernej $(\sqrt{5}-1)/2$.

Aby to zrozumieć, rozważmy relację rekurencyjną (3.6) podstawiając rozwiązanie próbne w postaci

$$x_n = \lambda^n\,, \tag{3.9}$$

Otrzymujemy wtedy równanie

$$\lambda^2 + \lambda - 1 = 0\,. \tag{3.10}$$

Ma ono dwa rozwiązania

$$\lambda_1 = \frac{\sqrt{5}-1}{2}, \qquad \lambda_2 = -\frac{\sqrt{5}+1}{2}, \tag{3.11}$$

spełniające warunki: $\lambda_1 < 1$ oraz $|\lambda_2| \approx 1.618 > 1$. Stąd ogólne rozwiązanie rekurencji w postaci

$$x_n = A\,(\lambda_1)^n + B\,(\lambda_2)^n\,. \tag{3.12}$$

Współczynniki A i B wyznaczymy korzytając z warunków początkowych (3.7)

$$\begin{aligned} A + B &= 1 \\ A\,\lambda_1 + B\,\lambda_2 &= \lambda_1 + \epsilon\,, \end{aligned} \tag{3.13}$$

gdzie dodaliśmy mały czynnik $\epsilon \neq 0$ po prawej stronie. Zdaje on sprawę z tego, że liczba λ_1 jest zawsze reprezentowana w komputerze w sposób przybliżony ze względu na błąd zaokrąglenia nieskończonego rozwinięcia dwójkowego tej liczby. Rozwiązaniem układu równań (3.13) jest

$$A = 1 + \frac{\epsilon}{\sqrt{5}}, \qquad B = -\frac{\epsilon}{\sqrt{5}}\,. \tag{3.14}$$

Stąd ostateczne rozwiązanie rekurencji

$$x_n = \left(1 + \frac{\epsilon}{\sqrt{5}}\right)(\lambda_1)^n - \frac{\epsilon}{\sqrt{5}}\,(\lambda_2)^n\,. \tag{3.15}$$

Tak wiec, niezależnie od tego jak mały jest błąd ϵ, drugi wyraz będzie dominował w sumie od pewnej wartości n, gdyż $|(\lambda_2)^n| \to \infty$ dla $n \to \infty$. Relacja rekurencyjna (3.6) z warunkiem początkowym (3.7) jest więc niestabilna ze względu na małe zaburzenia wartości początkowych.

Łatwo sprawdzić, że relacja rekurencyjna (3.6) z warunkiem

$$x_0 = 1\,, \qquad x_1 = \lambda_2\,, \tag{3.16}$$

który prowadzi do rozwiązania $x_n = (\lambda_2)^n$, jest stabilna ze względu na małe zaburzenia warunku początkowego.

3.3 Złożoność obliczeniowa

Złożoność obliczeniowa jest określona przez liczbę kroków (działań), które należy wykonać w danym algorytmie by osiągnąć zamierzony wynik.

Jako przykład, rozważmy problem obliczenia wartości wielomianu stopnia n,

$$W_n(x) = a_0 + a_1\,x + a_2\,x^2 + \ldots + a_n\,x^n\,, \tag{3.17}$$

dla pewnej wartości x. Licząc "naiwnie" mamy do wykonania n dodawań oraz następującą liczbę mnożeń

$$1 + 2 + \ldots + (n-1) + n = \frac{n(n+1)}{2} \tag{3.18}$$

W sumie liczba działań zależy *kwadratowo* od stopnia wielomianu n.

Dla wielomianu stopnia 10 wykonujemy więc $55 \sim O(10^2)$ działań, natomiast dla wielomianu stopnia 100 to liczba $5050 \sim O(10^4)$. Dobrze by było znaleźć algorytm, w którym liczba wykonanych działań byłaby znacząco mniejsza, chociażby ze względu ma ryzyko kumulacji błędów zaokrągleń.

Takim algorytmem jest *algorytm Hornera*. Obliczmy wartość wielomianu zapisując go w następujący sposób:

$$W_n(x) = (\ldots((a_n x + a_{n-1})\, x + a_{n-2})\, x + a_{n-3})\, x + \ldots + a_1)\, x + a_0)\,. \tag{3.19}$$

Tym razem występuje tu n mnożeń oraz n dodawań. Liczba działań zależy więc *liniowo* od n. Na przykład, dla wielomianu stopnia 100 wykonujemy tylko 200 działań!

Algorytmy, w których liczba kroków do wykonania by osiągnąc cel rośnie szybciej niż potęgowo z wymiarem problemu n (w naszym przykładzie określonym przez stopień wielomianu) **nie jest** *praktycznie obliczalny*. Takim wzrostem jest wzrost wykładniczy e^n. Już dla $n = 10$ mamy $\mathrm{e}^{10} \approx 10^4$ kroków, podczas gdy dla $n = 100$ liczba kroków to $\mathrm{e}^{100} \approx 10^{43}$!

Przykładem takiego problemu jest rozkład danej liczby naturalnej na iloczyn liczb pierwszych. Wszystkie algorytmy implementowane na komputerze klasycznym zależą wykładniczo od długości liczby n, która określa wymiar problemu. Dopiero *kwantowy algorytm Schora*, implementowany na komputrze kwantowym, pozwala rozwiązać ten problem w przy pomocy około $(\ln n)^3$ liczby kroków. Problem tylko w tym, że jak dotąd nie skonstruowano stabilnie działającego komputera kwantowego.

Wykład 4

Macierze

4.1 Układ równań liniowych

Macierze pojawiają się naturalnie w problemie rozwiązywania układów równań liniowych

$$\begin{aligned} a_{11}\,x_1 + a_{12}\,x_2 + ... + a_{1n}\,x_n &= y_1 \\ a_{21}\,x_1 + a_{22}\,x_2 + ... + a_{2n}\,x_n &= y_2 \\ &... \\ a_{n1}\,x_1 + a_{n2}\,x_2 + ... + a_{nn}\,x_n &= y_n\,. \end{aligned} \tag{4.1}$$

Układ ten można zapisać w postaci macierzowej

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}. \tag{4.2}$$

lub w skróconej formie.

$$\mathbf{A}\,\mathbf{x} = \mathbf{y}\,. \tag{4.3}$$

Tak więc $\mathbf{A}$ jest macierzą kwadratową o wymiarze $n \times n$. Rozwiązanie układu (4.1) istnieje i jest ono jednoznaczne jeśli wyznacznik

$$\det \mathbf{A} \neq 0\,. \tag{4.4}$$

Istnieje wtedy macierz odwrotna $\mathbf{A}^{-1}$ i rozwiązanie jest zadane przez

$$\mathbf{x} = \mathbf{A}^{-1}\,\mathbf{y}\,. \tag{4.5}$$

Metoda ta wymaga więc znajomości metod odwracania macierzy $\mathbf{A}$.

Alternatywnie, można skorzystać ze wzorów Cramera

$$x_1 = \frac{\det \mathbf{A}_1}{\det \mathbf{A}}\,, \qquad x_2 = \frac{\det \mathbf{A}_2}{\det \mathbf{A}}\,, \qquad \dots \qquad x_n = \frac{\det \mathbf{A}_n}{\det \mathbf{A}}\,. \tag{4.6}$$

Macierze $\mathbf{A}_i$ w liczniku powstały poprzez zamianę i-tej kolumny w macierzy $\mathbf{A}$ przez wektor $\mathbf{y}$. Przykładowo

$$\mathbf{A}_1 = \begin{pmatrix} y_1 & a_{12} & \dots & a_{1n} \\ y_2 & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ y_n & a_{n2} & \dots & a_{nn} \end{pmatrix}. \tag{4.7}$$

Z punktu widzenia metod numerycznych wzory Cramera można zastosować jedynie dla macierzy o małych wymiarach. Obliczenie wyznacznika macierzy o wymiarze $n \times n$ wymaga bowiem wykonanie $n!$ mnożeń oraz $n!$ dodawań. We wzorach (4.6) musimy policzyń $(n+1)$ wyznaczników macierzy, stąd całkowita liczba działań potrzebnych do znalezienia rozwiązania wynosi $2\,(n+1)!$. Korzystając ze wzoru Stirlinga, słusznego dla dużych n,

$$n! \approx n^n \, \mathrm{e}^{-n} \sqrt{2\pi\, n} \;\sim\; \mathrm{e}^{n \ln n - n}\,, \tag{4.8}$$

widzimy, że liczba działań w metodzie Cramera rośnie wykładniczo z n i metoda szybko staje się bezużyteczna. Przykładowo, rozwiązując układ złożony z 15 równań musimy wykonać

$$2 \cdot 16! \approx 10^{13} \tag{4.9}$$

działań.

Przykład ten ilustruje zagadnienie **złożoności obliczeniowej**. Wszystkie algorytmy, w których liczba działań do wykonania rośnie *wykładniczo* z wymiarem problemu szybko stają się bezużyteczne. Aby rozwiązać problem musimy znaleźć bardziej wydajne algorytmy, w których liczba działań zależy na przykład potęgowo od n.

4.2 Metoda eliminacji Gaussa

Metoda eliminacji Gaussa pozwala znaleźć rozwiązanie układu równań liniowych przy potęgowo zależnej od n liczbie wykonanych działań. Pozwala ona sprowadzić układ (4.1) do *równoważnej* postaci z macierzą trójkątną, w której wszystkie składowe poniżej (lub powyżej) diagonali są równe zeru

$$\begin{pmatrix} a'_{11} & a'_{12} & \dots & a'_{1(n-1)} & a'_{1n} \\ 0 & a'_{22} & \dots & a'_{2(n-1)} & a'_{2n} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \dots & a'_{(n-1)(n-1)} & a'_{(n-1)n} \\ 0 & 0 & \dots & 0 & a'_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_{n-1} \\ y'_n \end{pmatrix}. \tag{4.10}$$

Wyznacznik macierzy trójkątnej jest iloczynem liczb na diagonali

$$\det \mathbf{A}' = \prod_{i=1}^{n} a'_{ii}\,. \tag{4.11}$$

Warunkiem istnienia rozwiązania układu (4.10) jest by $\det \mathbf{A}' \neq 0$. Stąd wszystkie wyrazy na diagonali są rózne od zera. Łatwo już rowiązać taki układ. Zaczynając od ostatniego równania dostajemy

$$\begin{aligned} x_n &= \frac{1}{a'_{nn}} y'_n \\ x_{n-1} &= \frac{1}{a'_{(n-1)(n-1)}} \left(y'_{n-1} - a'_{(n-1)n} x_n \right) \\ &\dots \\ x_i &= \frac{1}{a'_{ii}} \left(y'_i - \sum_{k=i+1}^{n} a'_{ik} x_k \right). \end{aligned} \tag{4.12}$$

Rozważmy równanie (4.1):

$$\mathbf{A}\,\mathbf{x} = \mathbf{y}\,.$$

W pierwszym kroku metody eliminacji Gaussa mnożymy pierwsze równanie w tym układzie przez a_{i1}/a_{11} i odejmujemy kolejno od i-tych równań, gdzie $i = 2, 3, \dots, n$. W wyniku tego otrzymamy układ równań:

$$\mathbf{A}^{(1)}\,\mathbf{x} = \mathbf{y}^{(1)}\,, \tag{4.13}$$

lub w jawnej postaci

$$\begin{aligned} a^{(1)}_{11} x_1 + a^{(1)}_{12} x_2 + ... + a^{(1)}_{1n} x_n &= y^{(1)}_1 \\ a^{(1)}_{22} x_2 + ... + a^{(1)}_{2n} x_n &= y^{(1)}_2 \\ ... \\ a^{(1)}_{n2} x_2 + ... + a^{(1)}_{nn} x_n &= y^{(1)}_n\,. \end{aligned} \tag{4.14}$$

Wyeliminowaliśmy w ten sposób pierwszą niewiadomą w równaniach o numerach $i \geqslant 2$.

Mnożymy następnie drugie równanie w układzie (4.14) przez $a^{(1)}_{i2}/a^{(1)}_{22}$ i odejmujemy od i-tego równania dla $i = 3, 4, \dots, n$. Eliminujemy więc drugą niewiadomą x_2 z równań o numerach $i \geqslant 3$, otrzymując nowy układ:

$$\mathbf{A}^{(2)}\,\mathbf{x} = \mathbf{y}^{(2)}\,, \tag{4.15}$$

lub

$$\begin{aligned} a^{(2)}_{11} x_1 + a^{(2)}_{12} x_2 + a^{(2)}_{13} x_3 + ... + a^{(2)}_{1n} x_n &= y^{(2)}_1 \\ a^{(2)}_{22} x_2 + a^{(2)}_{23} x_3 + ... + a^{(2)}_{2n} x_n &= y^{(2)}_2 \\ ... \\ a^{(2)}_{n2} x_3 + ... + a^{(2)}_{nn} x_n &= y^{(2)}_n\,. \end{aligned} \tag{4.16}$$

Procedurę tę kontynuujemy aż do chwili otrzymania układu równań z macierza trójkątną:

$$\mathbf{A}^{(n-1)}\,\mathbf{x} = \mathbf{y}^{(n-1)},, \tag{4.17}$$

lub

$$\begin{aligned} a_{11}^{(n-1)} x_1 + a_{12}^{(n-1)} x_2 + a_{13}^{(n-1)} x_3 + ... + a_{1n}^{(n-1)} x_n &= y_1^{(n-1)} \\ a_{22}^{(n-1)} x_2 + a_{23}^{(n-1)} x_3 + ... + a_{2n}^{(n-1)} x_n &= y_2^{(n-1)} \\ ... & \\ a_{nn}^{(n-1)} x_n &= y_n^{(n-1)} \end{aligned} \tag{4.18}$$

Następnie rozwiązujemy ten układ korzystając ze wzorów (4.12).

Do wyznaczenia tą metodą rozwiązania wykonujemy, odpowiednio

$$\tfrac{1}{3}n^3 + n^2 - \tfrac{1}{3}n\,, \qquad\qquad \tfrac{1}{3}n^3 + \tfrac{1}{2}n^2 - \tfrac{5}{6}n \tag{4.19}$$

mnożeń i dzieleń oraz dodawań. Całkowita liczba działań zależy więc *potęgowo* od n.

4.3 Rozkład LU macierzy

Bardzo pożytecznym dla zastosowań numerycznych jest rozkład danej macierzy $\mathbf{A}$ na iloczyn dwóch macierzy trójkątnych $\mathbf{L}$ i $\mathbf{U}$:

$$A = \mathbf{L}\,\mathbf{U} \tag{4.20}$$

takich, że

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ l_{21} & 1 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ l_{n1} & l_{n2} & \dots & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ 0 & u_{22} & \dots & u_{2n} \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & u_{nn} \end{pmatrix}. \tag{4.21}$$

W takim przypadku wyznaczenie rozwiązania układu równań

$$\mathbf{A}\,\mathbf{x} = \mathbf{L}(\mathbf{U}\,\mathbf{x}) = \mathbf{y} \tag{4.22}$$

polega na rozwiązaniu dwóch układów równań liniowych z macierzami trójkątnymi

$$\mathbf{L}\,\mathbf{w} = \mathbf{y}\,, \qquad\qquad \mathbf{U}\,\mathbf{x} = \mathbf{w}\,, \tag{4.23}$$

stosując metodę z poprzedniego rozdziału.

Metoda eliminacji Gaussa prowadzi do rozkładu $\mathbf{LU}$, gdyż kolejne jej kroki są równoważne operacji mnożenia przez odpowiednie macierze $\mathbf{L}^{(i)}$:

$$\mathbf{A}^{(n-1)} = \mathbf{L}^{(n-1)}\,\mathbf{L}^{(n-2)}\,\dots\,\mathbf{L}^{(1)}\,\mathbf{A}\,, \tag{4.24}$$

gdzie $\mathbf{A}^{(n-1)}$ jest macierzą trójkątną typu $\mathbf{U}$. Podobnie dla wektora $\mathbf{y}$:

$$\mathbf{y}^{(n-1)} = \mathbf{L}^{(n-1)}\,\mathbf{L}^{(n-2)}\,\dots\,\mathbf{L}^{(1)}\,\mathbf{y}\,. \tag{4.25}$$

Kolejne macierze $\mathbf{L}$ to

$$\mathbf{L}^{(1)} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ -l_{21} & 1 & 0 & \dots & 0 \\ -l_{31} & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ -l_{n1} & 0 & 0 & \dots & 1 \end{pmatrix} \qquad l_{i1} = a_{i1}/a_{11}\,, \qquad i > 1 \tag{4.26}$$

następnie

$$\mathbf{L}^{(2)} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & -l_{32} & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & -l_{n2} & 0 & \dots & 1 \end{pmatrix} \qquad l_{i2} = a^{(1)}_{i2}/a^{(1)}_{12}\,, \qquad i > 2 \tag{4.27}$$

gdzie $a^{(1)}_{ik}$ to współczynniki macierzy

$$\mathbf{A}^{(1)} = \mathbf{L}^{(1)}\,\mathbf{A}\,, \tag{4.28}$$

w której wyzerowane są elementy pierwszej kolumny z wyjątkiem $a^{(1)}_{11}$. Postępując w ten sposób aż do macierzy $\mathbf{L}^{(n-1)}$ otrzymujemy wzór (4.24).

Macierze $\mathbf{L}^{(i)}$ są nieosobliwe, istnieje więc dla każdej z nich macierz odwrotna $(\mathbf{L}^{(i)})^{-1}$, która powstaje poprzez zamianę występujących w $\mathbf{L}^{(i)}$ minusów na plusy. Odwracając więc wzór (4.24), znajdujemy

$$\mathbf{A} = \underbrace{(\mathbf{L}^{(1)})^{-1}(\mathbf{L}^{(2)})^{-1}\dots(\mathbf{L}^{(n-1)})^{-1}}_{\mathbf{L}}\,\underbrace{(\mathbf{A}^{(n-1)})}_{\mathbf{U}}\,. \tag{4.29}$$

Otrzymujemy w ten sposób rozkład (4.21), gdyż $\mathbf{A}^{(n-1)}$ jest macierzą trójkątną typu U, natomiast iloczyn kolejnych macierzy odwrotnych to macierz typu L:

$$L = \begin{pmatrix} 1 & 0 & \dots & 0 \\ l_{21} & 1 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ l_{n1} & l_{n2} & \dots & 1 \end{pmatrix}. \tag{4.30}$$

4.4 Metoda Doolittle'a

Współczynniki macierzy $\mathbf{L}$ i $\mathbf{U}$ można także znaleźć bezpośrednio traktując (4.21) jako układ n^2 równań na n^2 poszukiwanych współczynników l_{ij} $(i > j)$ oraz u_{ij} $(i \leqslant j)$.

Rozważmy dla uproszczenia $n = 3$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{pmatrix}. \tag{4.31}$$

Mnożąc pierwszy wiersz macierzy **L** przez kolumny macierzy **U**, otrzymujemy

$$\begin{aligned} u_{11} &= a_{11} \\ u_{12} &= a_{12} \\ u_{13} &= a_{13}\,. \end{aligned} \tag{4.32}$$

Mnożąc drugi wiersz macierzy **L** przez kolumny macierzy **U**, znajdujemy

$$l_{21}\,u_{11} = a_{21}\,, \qquad l_{21}\,u_{12} + u_{22} = a_{22}\,, \qquad l_{21}\,u_{13} + u_{23} = a_{23}\,,$$

skąd przy założeniu $u_{11} \neq 0$ można wyznaczyć

$$\begin{aligned} l_{21} &= a_{21}/u_{11} \\ u_{22} &= a_{22} - l_{21}\,u_{12} \\ u_{23} &= a_{23} - l_{21}\,u_{13}\,. \end{aligned} \tag{4.33}$$

Mnożąc trzeci wiersz macierzy **L** przez kolumny macierzy **U**, dostajemy

$$l_{31}\,u_{11} = a_{31}\,, \qquad l_{31}\,u_{12} + l_{32}\,u_{22} = a_{32}\,, \qquad l_{31}\,u_{13} + l_{32}\,u_{23} + u_{33} = a_{33}\,,$$

co przy założeniu $u_{22} \neq 0$ prowadzi do

$$\begin{aligned} l_{31} &= a_{31}/u_{11} \\ l_{32} &= (a_{32} - l_{31}\,u_{12})/u_{22} \\ u_{33} &= a_{33} - l_{31}\,u_{13} - l_{32}\,u_{23}\,. \end{aligned} \tag{4.34}$$

Łatwo stąd wydedukować wzór dla macierzy o dowolnym wymiarze n.

$$u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik}\,u_{kj}\,, \qquad i \leqslant j \tag{4.35}$$

$$l_{ij} = \frac{1}{u_{jj}} \left(a_{ij} - \sum_{k=1}^{j-1} l_{ik}\,u_{kj} \right), \qquad i > j \tag{4.36}$$

Zauważmy, że obliczając l_{ij} wykorzystujemy elementy macierzy L poprzedzające ten element w wierszu oraz elementy macierzy U w wierszach powyżej. Podobnie, dla obliczenia u_{ij} wykorzystujemy obliczone już elementy macierzy L w wierszu oraz elementy macierzy U w wierszach powyżej. W ten sposób powyższe wzory definiują algorytm rozkładu LU, w którym obliczamy kolejne elementy macierzy L i U poruszając się wzdłuż wierszy od lewa do prawa z góry na dół.

Pozostaje do wyjaśnienia co należy zrobić jeśli w trakcie obliczania pojawi się $u_{jj} = 0$ dla $j < n$. Należy wtedy zamieniać wiersz j z wierszem $j+1$ lub wyższym do chwili gdy nowe u_{jj} jest różne od zera, a następnie powtórzyć algorytm od początku. Znaleziony w ten sposób rozkład LU będzie odpowiadał macierzy A z przestawionymi wierszami. Pojawienie się natomiast $u_{nn} = 0$ nie stanowi przeszkody, gdyż oznacza to, że wyznacznik macierzy A wynosi 0.

Przedstawiona metoda nazywa się metodą Doolittle'a. Rozwiązując przy jej pomocy układ równań (4.1) wykonuje się tyle samo działań algebraicznych co w metodzie eliminacji Gaussa. Zależność od n jest więc *potęgowa*.

4.5 Wyznacznik macierzy

Rozkładu **LU** umożliwia także policzenie wyznacznika macierzy **A**, korzystając z własności wyznacznika iloczynu macierzy:

$$\det \mathbf{A} = \det \mathbf{L} \cdot \det \mathbf{U} \tag{4.37}$$

oraz z faktu, że wyznacznik macierzy trójkątnej jest iloczynem liczb na diagonali. Tak więc ostatecznie otrzymujemy

$$\boxed{\det \mathbf{A} = \prod_{i=1}^{n} u_{ii}} \tag{4.38}$$

Wykład 5

Interpolacja

5.1 Wielomian interpolacyjny Lagrange'a

Poszukujemy ogólnej zależności funkcyjnej w sytuacji gdy znamy tę zależność w $(n+1)$ punktach:

$$\begin{array}{cccc} x_0 & x_1 & ... & x_n \\ y_0 & y_1 & ... & y_n\,. \end{array} \tag{5.1}$$

Punkty x_i nazywamy *węzłami* interpolacji, przy czym $x_i < x_j$ dla $i < j$. Problem ten ma jednoznaczne rozwiązanie jeśli poszukiwaną funkcją jest *wielomian stopnia n*

$$W_n(x) = a_0 + a_1\,x + ... + a_n\,x^n\,. \tag{5.2}$$

Współczynniki wielomianu a_i muszą być tak dobrane by wielomian W_n przechodził przez każdy punkt (x_i, y_i):

$$y_i = W_n(x_i) \qquad\qquad i = 0, 1, \ldots, n\,. \tag{5.3}$$

Warunek ten prowadzi do układu liniowego $(n+1)$ równań na $(n+1)$ niewiadomych współczynników wielomianu a_i:

$$\begin{aligned} a_0 + a_1\,x_0 + ... + a_n\,x_0^n &= y_0 \\ a_0 + a_1\,x_1 + ... + a_n\,x_1^n &= y_1 \\ &... \\ a_0 + a_1\,x_n + ... + a_n\,x_n^n &= y_n\,, \end{aligned} \tag{5.4}$$

lub zapisując w postaci macierzowej

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \ldots & x_0^n \\ 1 & x_1 & x_1^2 & \ldots & x_1^n \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \ldots & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}. \tag{5.5}$$

Jednoznaczne rozwiąznie powyższego układu równań istnieje gdyż wyznacznik macierzy głównej (zwany wyznacznikiem Vandermonde'a) jest różny od zera

$$\det = \prod_{(i,j)\, i>j} (x_i - x_j) \neq 0\,. \tag{5.6}$$

Istnieje więc dokładnie jeden wielomian interpolujący (5.2) przechodzący przez punkty (5.1), zwany *wielomianem Lagrange'a*.

Aby znaleźć jawną postać wielomianu Lagrange'a należy rozwiązać układ równań (5.5). Dzięki warunkowi jednoznaczności alternatywną metodą jest poszukiwanie wielomianu w postaci

$$\boxed{W_n(x) = y_0\, L_0(x) + y_1\, L_1(x) + ... + y_n\, L_n(x)\,,} \tag{5.7}$$

gdzie $L_i(x)$ są wielomianami stopnia n spełniającymi następujący warunek

$$L_i(x_j) = \delta_{ij} = \begin{cases} 1 & \text{dla } i = j \\ 0 & \text{dla } i \neq j\,. \end{cases} \tag{5.8}$$

Spełniony jest wtedy warunek (5.3) nałożony na wielomian Lagrange'a:

$$W_n(x_j) = \sum_{i=0}^{n} y_i\, L_i(x_j) = \sum_{i=0}^{n} y_i\, \delta_{ij} = y_j\,. \tag{5.9}$$

Zgodnie z warunkiem (5.8) każdy wielomian $L_i(x)$ zeruje się we wszystkich węzłach z wyjątkiem $x = x_i$. Jest więc proporcjonalny do iloczynu wszystkich jednomianów $(x - x_j)$ z wyłączeniem czynnika $(x - x_i)$:

$$L_i(x) = a\,(x - x_0)...(x - x_{i-1})\,(x - x_{i+1})...(x - x_n)\,. \tag{5.10}$$

Współczynnik a wyznaczamy z warunku $L_i(x_i) = 1$, co prowadzi do wzoru

$$\boxed{L_i(x) = \frac{(x - x_0)...(x - x_{i-1})\,(x - x_{i+1})...(x - x_n)}{(x_i - x_0)...(x_i - x_{i-1})\,(x_i - x_{i+1})...(x_i - x_n)}} \tag{5.11}$$

Wyrażenia (5.7) i (5.11) tworzą ostateczną postać wielomianu interpolacyjnego Lagrange'a.

Przyklad

Rozważmy interpolację zależności funkcyjnej określonej w trzech węzłach x_0, x_1, x_2. Wielomian Lagrange'a przyjmuje postać

$$W_2(x) = y_0\, \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} + y_1\, \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} + y_2\, \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}$$

Podsumowując, interpolacja przy pomocy wielomianu Lagrange'a jest jednoznacznym rozwiązaniem problem znalezienia funkcji, przechodzącej przez zadane punkty. Ponieważ prawdziwa zależność funkcyjna nie jest znana, pytanie o błąd interpolacji nie ma sensu.

5.2 Interpolacja Lagrange'a znanej funkcji

Często w zastosowaniach numerycznych mamy do czynienia z problemem przybliżenia *znanej* funkcji $y = f(x)$ w przedziale $[a,b]$ przy pomocy wyrażenia wykorzystującego znajomość tej funkcji w skończonej liczbie punktów z tego przedziału. Wybierzmy w tym celu $(n+1)$ węzłów

$$a = x_0 < x_1 < \ldots < x_n = b \tag{5.12}$$

i znajdźmy odpowiadające im wartości funkcji

$$f(x_0) \quad f(x_1) \quad \ldots \quad f(x_n)\,. \tag{5.13}$$

Przybliżmy następnie funkcję przy pomocy wielomianu Lagrange'a (5.7):

$$W_n(x) = \sum_{i=0}^{n} f(x_i)\, L_i(x)\,. \tag{5.14}$$

Zakładając, że istnieje $(n+1)$ pochodna f można pokazać (dowód w [1]), że

$$f(x) - W_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\, \omega_{n+1}(x) \tag{5.15}$$

gdzie $\xi \in [a,b]$, natomiast $\omega_{n+1}(x)$ to wielomian stopnia $(n+1)$:

$$\omega_{n+1}(x) = (x-x_0)(x-x_1)\ldots(x-x_n)\,. \tag{5.16}$$

Ze wzoru tego wynika, że maksymalny błąd interpolacji Lagrange'a jest ograniczony przez

$$\boxed{|f(x) - W_n(x)| \leqslant \frac{M_{n+1}}{(n+1)!}\, |\omega_{n+1}(x)|} \tag{5.17}$$

gdzie M_{n+1} to kres górny

$$M_{n+1} = \sup_{x\in[a,b]} |f^{(n+1)}(x)| \tag{5.18}$$

Zauważmy, że dla wielomianów stopnia n pochodne rzędu $\geqslant (n+1)$ znikają. Stąd $M_{n+1} = 0$ i błąd interpolacji Lagrange'a wynosi zero. Prawdziwe są zatem

Twierdzenia
Dla każdego wielomianu stopnia n, interpolacja Lagrange'a z liczbą węzłów $\geqslant (n+1)$ *jest dokładna.*

Interpolacja Lagrange'a przy pomocy $(n+1)$ *węzłów jest dokładna dla każdego wielomianu stopnia* $\leqslant n$.

Powstaje pytanie, jak dobrać węzły interpolacji by prawa strona wyrażenia (5.17) była jak najmniejsza. Należy w tym celu znaleźć najmniejsze oszacowanie kresu górnego wartości $|\omega_{(n+1)}(x)|$ w przedziale $[a,b]$. Odpowiedź na to pytanie otrzymujemy rozważając wielomiany Czebyszewa.

5.3 Interpolacja przy pomocy funkcji sklejanych

Dla każdego układu $(n+1)$ węzłów w przedziale $[a,b]$ istnieje funkcja ciągła w tym przedziale, dla której metoda interpolacyjna nie jest zbieżna. Dla takiej funkcji zwiększenie liczby węzłów pogarsza dokładność interpolacji (zwłaszcza bliżej końców przedziału). Przykładem jest funkcja $f(x) = |x|$ interpolowana przy pomocy równoodległych węzłów.

W interpolacji przy pomocy funkcji sklejanych można osiągnąć bardzo dobrą dokładność przy pomocy małej liczby węzłów. Rozważmy $(n+1)$ węzłów oraz n odpowiadających im przedziałów:

$$[x_0, x_1] \ \cup \ [x_1, x_2] \ \cup \ \ldots \ \cup \ [x_{n-1}, x_n], \tag{5.19}$$

numerowanych indeksem prawego końca przedziału.

W każdym przedziale funkcja $f(x)$ jest interpolowana przy pomocy wielomianu trzeciego stopnia:

$$P_i(x) = a_{0\,i} + a_{1\,i}\,x + a_{2\,i}\,x^2 + a_{3\,i}\,x^3, \tag{5.20}$$

gdzie $i = 1, 2, \ldots n$. Otrzymujemy więc n wielomianów o $4n$ współczynnikach do wyznaczenia. Nakładamy na nie następujące $4n$ warunków, które prowadzą do jednoznacznego rozwiązania.

1. Na brzegach każdego przedziału spełnione jest $2n$ warunków interpolacji:

$$\begin{aligned} P_i(x_{i-1}) &= f(x_{i-1}) \\ P_i(x_i) &= f(x_i), \end{aligned} \tag{5.21}$$

   gdzie $i = 1, 2, \ldots n$.

2. W punktach wewnętrznych $\{x_1, \ldots, x_{n-1}\}$ spełnione jest $2(n-1)$ warunków ciągłości dla pierwszych i drugich pochodnych:

$$\begin{aligned} P_i'(x_i) &= P_{i+1}'(x_i) \\ P_i''(x_i) &= P_{i+1}''(x_i), \end{aligned} \tag{5.22}$$

   gdzie $i = 1, 2, \ldots (n-1)$.

3. W punktach zewnętrznych $\{x_0, x_n\}$ żądamy dla drugich pochodnych:

$$\begin{aligned} P_1''(x_0) &= f''(x_0) \\ P_n''(x_n) &= f''(x_n). \end{aligned} \tag{5.23}$$

   Stąd dwa warunki.

W sumie otrzymujemy $\{2n + 2(n-1) + 2\} = 4n$ poszukiwanych warunków.

Wykład 6

Interpolacja optymalna

6.1 Wielomiany Czebyszewa

Wielomian Czebyszewa stopnia n jest zdefiniowany przy pomocy wzoru

$$\boxed{T_n(x) = \cos(n \arccos x)} \tag{6.1}$$

dla $n = 0, 1, 2, \ldots$ oraz $x \in [-1, 1]$. Z definicji tej wynika ważna relacja

$$|T_n(x)| \leqslant 1\,. \tag{6.2}$$

Dwa pierwsze wielomiany Czebyszewa to

$$\begin{aligned} T_0(x) &= 1 \\ T_1(x) &= x\,. \end{aligned} \tag{6.3}$$

Kolejne wielomiany można wyliczyć korzystając z relacji rekurencyjnej słusznej dla wartości stopnia wielomianu $n \geqslant 1$:

$$\boxed{T_{n+1}(x) = 2x\,T_n(x) - T_{n-1}(x)\,,} \tag{6.4}$$

Dowód
Podstawiając $\phi = \arccos x$, rozważmy

$$\begin{aligned} T_{n+1}(x) &= \cos(n+1)\phi = \cos(n\phi)\cos\phi - \sin(n\phi)\sin\phi \\ &= x\,T_n(x) - \sin(n\phi)\sin\phi\,. \end{aligned} \tag{6.5}$$

Wykorzystując następnie tożsamość trygonometryczną

$$\sin\alpha \sin\beta = \tfrac{1}{2}\{\cos(\alpha-\beta) - \cos(\alpha+\beta)\} \tag{6.6}$$

otrzymamy

$$\begin{aligned} T_{n+1} &= x\,T_n - \tfrac{1}{2}\{\cos((n-1)\phi) - \cos((n+1)\phi)\} \\ &= x\,T_n - \tfrac{1}{2}\left(T_{n-1} - T_{n+1}\right). \end{aligned} \tag{6.7}$$

Stąd wynika już relacja (6.4).

Tak więc następne wielomiany Czebyszewa to

$$\begin{aligned} T_2(x) &= 2x^2 - 1 \\ T_3(x) &= 4x^3 - 3x \\ T_4(x) &= 8x^4 - 8x^2 + 1\,. \end{aligned} \tag{6.8}$$

Widać stąd, że wielomian T_n jest funkcją o określonej parzystości, dodatniej dla parzystych n i ujemną dla nieparzystych wartości stopnia wielomianu. Ponadto, rozważając współczynnik przy najwyższej potędze zauważamy, że

$$T_n(x) = 2^{n-1}x^n + \cdots\,. \tag{6.9}$$

Każdy wielomian Czebyszewa stopnia n ma dokładnie n pierwiastków rzeczywistch. Aby je znaleźć rozwiązujemy równanie

$$T_n(x) = \cos(n \arccos x) = 0\,, \tag{6.10}$$

otrzymując następujące pierwiastki dla $k = 0, 1, \ldots, n-1$

$$\boxed{x_k = \cos\frac{\pi(k+1/2)}{n}} \tag{6.11}$$

Dowolny wielomian Czebyszewa można więc zapisać w postaci

$$T_n(x) = 2^{n-1}(x-x_0)(x-x_1)\ldots(x-x_{n-1}) = 2^{n-1}\,\omega_n(x)\,, \tag{6.12}$$

gdzie współczynnik 2^{n-1} wynika z uwagi (6.9).

6.2 Optymalny wybór węzłów interpolacji

Rozważmy najpierw funkcję $f(x)$ określoną na przedziale $[-1, 1]$. Oszacujemy maksymalną wartość kresu górnego wartości $|\omega_{(n+1)}(x)|$ w tym przedziale. W tym celu wybierzmy $(n+1)$ węzłów interpolacji będących zerami wielomianu Czebyszewa $T_{n+1}(x)$:

$$x_k = \cos\frac{\pi(k+1/2)}{(n+1)} \qquad\qquad k = 0, 1, \ldots n\,. \tag{6.13}$$

Na podstawie relacji (6.12) oraz własności (6.2) otrzymujemy dla tak wybranych węzłów relację

$$|\omega_{n+1}(x)| = \frac{1}{2^n}\,|T_{n+1}(x)| \leqslant \frac{1}{2^n}\,. \tag{6.14}$$

Wzór (5.17) przyjmuje więc teraz postać

$$\boxed{|f(x) - W_n(x)| \leqslant \frac{M_{n+1}}{2^n(n+1)!}} \tag{6.15}$$

gdzie M_{n+1} to kres górny wartości modułu $(n+1)$ pochodnej funkcji f:

$$M_{n+1} = \sup_{x\in[-1,1]} |f^{(n+1)}(x)|\,. \tag{6.16}$$

Otrzymaliśmy w ten sposób najmniejsze oszacowanie maksymalnego błędu interpolacji Lagrange'a.

6.3 Wzory dla dowolnego przedziału

Rozważmy następnie funkcję $f(y)$ określoną w dowolnym przedziale $[a,b]$. Optymalnie wybranymi węzłami są teraz obrazy zer (6.13) poprzez transformację liniową

$$y_k = \tfrac{1}{2}\left\{(b-a)\,x_k + (a+b)\right\}, \tag{6.17}$$

Łatwo sprawdzić, wszystkie obrazy węzłów Czebyszewa leżą w przedziale $[a,b]$.

Zastosujemy teraz wzory z poprzedniego rozdziału do $y \in [a,b]$

$$\begin{aligned}\omega_{n+1}(y) &= \prod_{k=0}^{n}(y-y_k) = \frac{(b-a)^{n+1}}{2^{n+1}}\prod_{k=0}^{n}(x-x_k)\\ &= \left(\frac{b-a}{2}\right)^{n+1}\omega_{n+1}(x)\,.\end{aligned} \tag{6.18}$$

Wykorzystując następnie relację (6.14) otrzymujemy

$$|\omega_{n+1}(y)| \leqslant \left(\frac{b-a}{2}\right)^{n+1}\frac{1}{2^n}. \tag{6.19}$$

Stąd najmniejsze oszacowanie błędu maksymalnego interpolacji Lagrange'a

$$\boxed{|f(y)-W_n(y)| \leqslant \frac{M_{n+1}}{2^n(n+1)!}\left(\frac{b-a}{2}\right)^{n+1}} \tag{6.20}$$

gdzie M_{n+1} to kres górny wartości modułu $(n+1)$ pochodnej funkcji f:

$$M_{n+1} = \sup_{y\in[a,b]}|f^{(n+1)}(y)|\,. \tag{6.21}$$

Wykład 7

Aproksymacja

W wielu przypadkach przybliżanie zależności funkcyjnej poprzez wielomian przechodzący przez wszystkie znane punkty nie jest dobrą metodą, w szczególności, gdy punkty są obarczone błędem lub gdy jest ich bardzo dużo. W tym ostatnim przypadku stopień wielomianu Lagrange'a byłby bardzo wysoki co prowadzić może do niestabilności numerycznych.

Znacznie lepszą metodą jest wtedy *aproksymacja* przy pomocy względnie prostej funkcji, tak by była ona jak najmniej "odległa" od aproksymowanej funkcji. Metodę tę zilustrujemy na początek w najprostszym przypadku, gdy dobrym przybliżeniem jest założenie o liniowej zależności funkcyjnej.

7.1 Regresja liniowa

Załóżmy, że mamy do czynienia z N węzłami $\{x_0, x_1 \ldots x_{N-1}\}$ i odpowiadającymi im wartościami $\{y_0, y_1 \ldots y_{N-1}\}$. Załóżmy, że dobrą poszukiwaną zależnością jest funkcja liniowa

$$y = ax + b\,, \tag{7.1}$$

gdzie parametry a i b pozostają do wyznaczenia. Utwórzmy w tym celu funkcję

$$S(a,b) = \sum_{k=0}^{N-1} \{y_k - (ax_k + b)\}^2 \tag{7.2}$$

będącą sumą kwadratów odległości punktów (x_k, y_k) od prostej (7.1), *mierzonych wzdłuż osi y*. Dobierzmy następnie parametry a i b tak, by wartość tej funkcji była jak najmniejsza. Różniczkując, otrzymamy

$$\begin{aligned} \frac{\partial S}{\partial a} &= \sum_{k=0}^{N-1} (-2x_k)(y_k - ax_k - b) = 0 \\ \frac{\partial S}{\partial b} &= \sum_{k=0}^{N-1} (-2)(y_k - ax_k - b) = 0\,, \end{aligned} \tag{7.3}$$

Stąd dostajemy układ równań na współczynniki a i b:

$$\begin{aligned} a(\sum_k x_k^2) + b(\sum_k x_k) &= \sum_k x_k y_k \\ a(\sum_k x_k) + b(\sum_k 1) &= \sum_k y_k \,. \end{aligned} \tag{7.4}$$

lub w postaci macierzowej

$$\begin{pmatrix} \sum_k x_k^2 & \sum_k x_k \\ \sum_k x_k & N \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \sum_k x_k y_k \\ \sum_k y_k \end{pmatrix} . \tag{7.5}$$

Wykorzystaliśmy przy tym obserwację $\sum_{k=0}^{n-1} 1 = n$. Rozwiązanie zadane jest poprzez wzory Cramera

$$\begin{aligned} a &= \frac{N(\sum_k x_k y_k) - (\sum_k x_k)(\sum_k y_k)}{N(\sum_k x_k^2) - (\sum_k x_k)^2} \\ b &= \frac{(\sum_k x_k^2)(\sum_k y_k) - (\sum_k x_k y_k)(\sum_k x_k)}{N(\sum_k x_k^2) - (\sum_k x_k)^2} \,. \end{aligned} \tag{7.6}$$

Uogólnienie tej metody polega na rozważeniu zależności wielomianowej

$$y = a_0 + a_1 x + \ldots + a_M x^M \tag{7.7}$$

i minimalizację ze względu na współczynniki tego wielomianu następującej różnicy

$$S(a_i) = \sum_{k=0}^{N-1} \left\{ y_k - (a_0 + a_1 x + \ldots + a_M x^M) \right\}^2 . \tag{7.8}$$

Zwróćmy uwagę, że na ogół liczba punktów N, w których znamy aproksymowaną funkcję, jest dużo wieksza od stopnia wielomianu aproksymującego M.

7.2 Aproksymacja średniokwadratowa

Przedstawiona w poprzednim rozdziale metoda jest przykładem aproksymacji średniokwadratowej. W ogólności, znając N węzłów $\{x_0, x_1 \ldots x_{N-1}\}$ oraz odpowiadających im wartości $\{f(x_0), f(x_1), \ldots f(x_{N-1})\}$, nieznanej na ogół funkcji, chcemy ją aproksymować przy pomocy wyrażenia

$$F(x) = c_0\,\phi_0(x) + c_1\,\phi_1(x) + \ldots + c_M\,\phi_M(x)\,, \tag{7.9}$$

gdzie

$$\{\phi_0(x), \phi_1(x), \ldots \phi_M(x)\} \tag{7.10}$$

jest układem dogodnie wybranych funkcji bazowych. W przykładzie (7.8) funkcjami bazowymi są jednomiany $\phi_n(x) = x^n$. Zauważmy, że liczba węzłów N jest niezależna od liczby funkcji bazowych $(M+1)$.

Współczynniki c_n są tak dobrane by zminimalizować odległość pomiędzy funkcją dokładną $f(x)$ i przybliżoną $F(x)$, zdefiniowaną poprzez:

$$\begin{aligned} S(c_i) &= \sum_{k=0}^{N-1} \{f(x_k) - F(x_k)\}^2 \\ &= \sum_{k=0}^{N-1} \{f(x_k) - c_0\,\phi_0(x_k) - c_1\,\phi_1(x_k) - \ldots - c_M\,\phi_M(x_k)\}^2 \,. \end{aligned} \tag{7.11}$$

Różniczkując po parametrach otrzymujemy dla każdego parametru c_n:

$$\frac{\partial S}{\partial c_n} = \sum_{k=0}^{N-1} (-2)\,\phi_n(x_k)\{f(x_k) - c_0\,\phi_0(x_k) - c_1\,\phi_1(x_k) - \ldots - c_M\,\phi_M(x_k)\} = 0\,.$$

Stąd następujący układ równań na poszukiwane współczynniki porozważeniu kolejnych wskaźników $n = 0, 1, \ldots M$:

$$\begin{aligned} (\phi_0\,,\phi_0)\,c_0 + (\phi_0\,,\phi_1)\,c_1 + \cdots + (\phi_0\,,\phi_M)\,c_M &= (\phi_0\,, f) \\ (\phi_1\,,\phi_0)\,c_0 + (\phi_1\,,\phi_1)\,c_1 + \cdots + (\phi_1\,,\phi_M)\,c_M &= (\phi_1\,, f) \\ \ldots & \\ (\phi_N\,,\phi_0)\,c_0 + (\phi_N\,,\phi_1)\,c_1 + \cdots + (\phi_N\,,\phi_M)\,c_M &= (\phi_N\,, f)\,, \end{aligned} \tag{7.12}$$

Wielkość $(\phi_n\,,\phi_m)$ dla $n, m = 0, 1, \ldots M$ to *peudo-iloczyn skalarny*[1], zdefiniowany przy pomocy dyskretnego zbioru węzłów $\{x_k\}$:

$$\boxed{(\phi_n\,,\phi_m) = \sum_{k=0}^{N-1} \phi_n(x_k)\,\phi_m(x_k)} \tag{7.13}$$

Rozwiązanie układu (7.12) jest szczególnie proste, gdy układ funkcji bazowych jest ortogonalny:

$$(\phi_n\,,\phi_m) \sim \delta_{nm}\,. \tag{7.14}$$

W równaniach (7.12) pozostają tylko składowe diagonalne i stąd

$$c_n = \frac{(\phi_n\,, f)}{(\phi_n\,,\phi_n)} = \frac{\sum_{k=0}^{N-1} \phi_n(x_k)\,f(x_k)}{\sum_{k=0}^{N-1} \phi_n^2(x_k)}. \tag{7.15}$$

dla $n = 0, 1, \ldots, M$. Ostatecznie funkcja aproksymująca to

$$\boxed{F(x) = \sum_{n=0}^{M} \frac{(\phi_n\,, f)}{(\phi_n\,,\phi_n)}\,\phi_n(x)} \tag{7.16}$$

W następnych rozdziałach przedstawimy przykłady dwóch szczególnie ważnych realizacji powyższego rozwiązania. Rozważymy aproksymację przy pomocy wielomianów Czebyszewa oraz funkcji trygonometrycznych.

[1] *Pseudo-*, gdyż nie jest spełniona własność iloczynu skalarnego $(\phi_n\,,\phi_n) = 0 \;\Rightarrow\; \phi_n(x) = 0$. W takim wypadku funkcje $\phi_n(x)$ muszą się zerować tylko w węzłach x_k.

Wykład 8

Przykłady aproksymacji

8.1 Aproksymacja Czebyszewa

Wybierzmy N węzłów $\{x_0, x_1 \ldots x_{N-1}\}$ będących zerami wielomianu Czebyszewa:

$$T_N(x_k) = 0 \tag{8.1}$$

Zgodnie ze wzorem (6.11) węzły

$$x_k = \cos\frac{\pi(k+1/2)}{N}, \qquad k = 0, 1, \ldots (N-1). \tag{8.2}$$

Wielomiany Chebyszewa $\{T_0, T_1, \ldots, T_{N-1}\}$ są ortogonalne względem peudo-iloczynu skalarnego (7.13) z powyższymi węzłami.

$$(T_n, T_m) = \sum_{k=0}^{N-1} T_n(x_k) T_m(x_k) = \begin{cases} N & \text{dla} \quad n = m = 0 \\ N/2 & \text{dla} \quad n = m \neq 0 \\ 0 & \text{dla} \quad n \neq m \end{cases} \tag{8.3}$$

gdzie $n, m = 0, 1, \ldots (N-1)$.

Dowód

Rozważmy peudo-iloczyn skalarny dla $0 \leqslant n, m \leqslant (N-1)$:

$$\sum_{k=0}^{N-1} T_n(x_k) T_m(x_k) = \sum_{k=0}^{N-1} \cos\left(n\frac{\pi(2k+1)}{2N}\right) \cos\left(m\frac{\pi(2k+1)}{2N}\right). \tag{8.4}$$

1. Dla $n = m = 0$ otrzymujemy sumę równą N.

2. Rozważmy następnie przypadek $n \neq m$. Korzystając ze wzoru

$$\cos\alpha \cos\beta = \tfrac{1}{2}\{\cos(\alpha+\beta) + \cos(\alpha-\beta)\} \tag{8.5}$$

otrzymujemy

$$(T_n, T_m) = \frac{1}{2} \sum_{k=0}^{N-1} \left\{ \cos\left((2k+1)\, \frac{\pi(n-m)}{2N} \right) + \cos\left((2k+1)\, \frac{\pi(n+m)}{2N} \right) \right\}. \tag{8.6}$$

Policzmy następnie dla dowolnego kąta $\phi \neq 0$:

$$\begin{aligned} \sum_{k=0}^{N-1} \cos(2k+1)\,\phi &= \mathrm{Re} \sum_{k=0}^{N-1} \mathrm{e}^{i(2k+1)\,\phi} \\ &= \mathrm{Re} \left\{ \mathrm{e}^{i\,\phi} \left(1 + \mathrm{e}^{2i\,\phi} + \ldots + \mathrm{e}^{2i(N-1)\,\phi}\right) \right\} \\ &= \mathrm{Re} \left\{ \mathrm{e}^{i\,\phi}\, \frac{1 - \mathrm{e}^{2i\,N\,\phi}}{1 - \mathrm{e}^{2i\,\phi}} \right\} = \frac{\sin(2N\,\phi)}{2 \sin\phi}\,. \end{aligned} \tag{8.7}$$

Podstawiając

$$\phi_{\pm} = \frac{\pi(n \pm m)}{2N} \neq 0\,, \tag{8.8}$$

otrzymujemy dla wzoru (8.6)

$$(T_n, T_n) = \frac{\sin \pi(n-m)}{4 \sin\, \phi_+} + \frac{\sin \pi(n+m)}{4 \sin\, \phi_-}\,, \tag{8.9}$$

Wyrażenie to znika dla $n \neq m$ ze względu na zerowanie się sinusów w liczniku. Stąd warunek ortogonalności dla wielomianów Czebyszewa.

3. Dla $n = m \neq 0$ równanie (8.6) przyjmuje postać

$$(T_n, T_n) = \frac{1}{2} \sum_{k=0}^{N-1} \left\{ 1 + \cos(2k+1)\, \frac{\pi\, n}{N} \right\} = \frac{N}{2} \tag{8.10}$$

gdyż z równania (8.7) dla $\phi = \pi n/N$ wynika, że suma cosinusów daje zero.

Możemy więc aproksymować przy pomocy wielomianów Czebyszewa funkcję $f(x)$ określoną w przedziale $[-1, 1]$:

$$f(x) \simeq \frac{1}{2}\, c_0 + \sum_{n=0}^{N-1} c_n\, T_n(x)\,. \tag{8.11}$$

Współczynniki c_n są wyliczone ze wzoru (7.15), w którym $\phi_n(x) = T_n(x)$:

$$\boxed{c_n = \frac{2}{N} \sum_{k=0}^{N-1} T_n(x_k)\, f(x_k)} \tag{8.12}$$

gdzie dla $n = 0$ podstawiliśmy dwukrotnie większą wartość c_0 niż ta wynikająca z normalizacji: $(T_0, T_0) = N$. Stąd współczynnik $1/2$ we wzorze (8.11).

W sumie (8.11) można zachować jedynie $M \leqslant (N-1)$ składników przy niezmienionych współczynnikach c_n. W większości przypadków stają się one coraz

mniejsze dla rosnących wartości wskaźnika, natomiast każdy wielomian Czebyszewa jest ograniczony warunkiem (6.2). Nie popełniamy w ten sposób dużego błędu odrzucając wyrazy z $n > M$. Stąd ostateczny wzór

$$\boxed{f(x) \simeq \frac{1}{2} c_0 + \sum_{n=0}^{M<N} c_n T_n(x)} \tag{8.13}$$

Podsumowując, kluczowym punktem w aproksymacji Czebyszewa jest znajomość funkcji $f(x)$ w węzłach Chebyszewa. Na tej podstawie konstruuje się współczynniki (8.12), a następnie przybliżenie (8.13).

8.2 Aproksymacja Czebyszewa w dowolnym przedziale

Rozważmy aproksymację funkcji $f(y)$ określoną dla $y \in [a, b]$. Niech

$$y = y(x)\,, \qquad x \in [-1, 1] \tag{8.14}$$

będzie dowolną transformacją bijektywną odwzorowującą $[-1, 1] \to [a, b]$. Transformacja odwrotna to

$$x = y^{-1}(y)\,, \qquad y \in [a, b]\,. \tag{8.15}$$

Zdefiniujmy nową funkcję określoną dla $x \in [-1, 1]$ wzorem:

$$\tilde{f}(x) = f(y(x))\,, \tag{8.16}$$

a następnie zastosujmy do niej wzór aproksymacyjny (8.13):

$$\tilde{f}(x) \approx \frac{1}{2} c_0 + \sum_{n=0}^{M<N} c_j n T_n(x)\,. \tag{8.17}$$

Podstawiając po prawej stronie relację odwrotną $x = y^{-1}(y)$, otrzymujemy aproksymację funkcji $f(y)$:

$$\boxed{f(y) \approx \frac{1}{2} c_0 + \sum_{n=0}^{M<N} c_n T_n(y^{-1}(y))} \tag{8.18}$$

gdzie współczynniki c_n dla $n = 0, 1 \ldots (N-1)$ są zadane przez

$$\boxed{c_n = \frac{2}{N} \sum_{k=0}^{N-1} T_n(x_k)\, f(y_k)} \tag{8.19}$$

Punkty $y_k = y(x_k)$ w powyższym wzorze są obrazami węzłów Czebyszewa (8.2).

Przyklad
Rozważmy transformację liniową

$$y = y(x) = \tfrac{1}{2}\{(b-a)\,x + (a+b)\} \tag{8.20}$$

przeprowadzającą

$$x \in [-1,1] \;\rightarrow\; y \in [a,b] \tag{8.21}$$

Transformacja odwrotna to

$$x = y^{-1}(y) = \frac{2y-a-b}{b-a} \tag{8.22}$$

8.3 Aproksymacja trygonometryczna

Aproksymację tę stosujemy, gdy mamy do czynienia z funkcją okresową $f(x)$ o okresie 2π. Załóżmy, że znamy tę funkcję w *parzystej* liczbie $2N$ równoodległych punktów z przedziału $[0, 2\pi]$:

$$x_k = \frac{k\pi}{N}\,, \qquad\qquad k = 0, 1, \ldots, (2N-1) \tag{8.23}$$

Właściwym układem funkcji bazowych dla aproksymacji (7.9) są wtedy funkcje trygonometryczne wraz ze stałą

$$\{\,1,\ \sin x,\ \cos x,\ \sin 2x,\ \cos 2x\ \ldots\ \sin(N-1)x,\ \cos(N-1)x\,\} \tag{8.24}$$

Zbiór tych funkcji jest ortogonalny względem peudo-iloczynu skalarnego (7.13) z punktami (8.23).

Dowód
Zachodzi bowiem dla $1 \leqslant n \leqslant (N-1)$:

$$\begin{aligned}\sum_{k=0}^{2N-1} \mathrm{e}^{inx_k} \;&=\; \sum_{k=0}^{2N-1} \{\cos(nx_k) + i\sin(mx_k)\} \\ &=\; \sum_{k=0}^{2N-1} \mathrm{e}^{(in\,\pi/N)k} = \frac{1-\mathrm{e}^{i2\pi n}}{1-\mathrm{e}^{in\,\pi/N}} = 0.\end{aligned} \tag{8.25}$$

Stąd relacje ortogonalności

$$\begin{aligned}\sum_{k=0}^{2N-1} \sin(nx_k)\cdot 1 \;&=\; 0 \\ \sum_{k=0}^{2N-1} \cos(nx_k)\cdot 1 \;&=\; 0 \\ \sum_{k=0}^{2N-1} 1\cdot 1 \;&=\; 2N\,.\end{aligned}$$

Ponadto dla $1 \leqslant n,m \leqslant (N-1)$ mamy

$$\begin{aligned}
\sum_{k=0}^{2N-1} \sin(nx_k)\cdot\sin(mx_k) &= \frac{1}{2}\sum_{k=0}^{2N-1}\{\cos(n-m)x_k - \cos(n+m)x_k\} = N\,\delta_{nm} \\
\sum_{k=0}^{2N-1} \cos(nx_k)\cdot\cos(mx_k) &= \frac{1}{2}\sum_{k=0}^{2N-1}\{\cos(n-m)x_k + \cos(n+m)x_k\} = N\,\delta_{nm} \\
\sum_{k=0}^{2N-1} \cos(nx_k)\cdot\sin(mx_k) &= \frac{1}{2}\sum_{k=0}^{2N-1}\{\sin(n-m)x_k + \sin(n+m)x_k\} = 0\,.
\end{aligned}$$

Stąd, aproksymując funkcję $f(x)$ przy pomocy funkcji trygonometrycznych, otrzymujemy

$$\boxed{f(x) \approx \frac{1}{2}a_0 + \sum_{n=1}^{N-1}\{a_n\cos(nx) + b_n\sin(nx)\}\,.} \tag{8.26}$$

Współczynniki a_n i b_n dla $n = 1\ldots(N-1)$ można wyliczyć ze wzoru (7.15):

$$\begin{aligned}
a_n &= \frac{1}{N}\sum_{k=0}^{2N-1} f(x_k)\cos(nx_k) \\
b_n &= \frac{1}{N}\sum_{k=0}^{2N-1} f(x_k)\sin(nx_k)\,,
\end{aligned} \tag{8.27}$$

natomiast

$$a_0 = \frac{1}{N}\sum_{k=0}^{2N-1} f(x_k)\,. \tag{8.28}$$

8.4 Wzory dla dowolnego okresu

Zastosujmy powyższą aproksymację do funkcji czasu o okresie T

$$f(t) = f(t+T)\,. \tag{8.29}$$

Zdefiniujmy nową funkcję $\tilde{f}(x)$ o okresie 2π:

$$\tilde{f}(x) = f(t) \tag{8.30}$$

przy pomocy transformacji:

$$t = x\,\frac{T}{2\pi}\,. \tag{8.31}$$

Aproksymujemy funkcję $\tilde{f}(x)$ za pomocą wzoru (8.26), a następnie podstawiamy po prawej stronie transformację odwrotną:

$$x = \frac{2\pi}{T}\,t\,. \tag{8.32}$$

Otrzymujemy w ten sposób dla funkcji $f(t)$:

$$\boxed{f(t) \approx \frac{1}{2}\,a_0 + \sum_{n=1}^{N-1} \left\{ a_n \cos\left(\frac{2\pi n}{T}\,t\right) + b_n \sin\left(\frac{2\pi n}{T}\,t\right) \right\}} \tag{8.33}$$

Współczynniki a_n i b_n są teraz zadane wzorami:

$$a_n \;=\; \frac{1}{N} \sum_{k=0}^{2N-1} f(t_k) \cos(n\,x_k) \tag{8.34}$$

$$b_n \;=\; \frac{1}{N} \sum_{k=0}^{2N-1} f(t_k) \sin(n\,x_k)\,, \tag{8.35}$$

oraz

$$a_0 = \frac{1}{N} \sum_{k=0}^{2N-1} f(t_k)\,, \tag{8.36}$$

w których wielkości t_n dla $n = 0, 1, \ldots (2N-1)$ są obrazami węzłów (8.23) poprzez transformację (8.31):

$$t_k = x_k\,\frac{T}{2\pi} = k\,\frac{T}{2N}\,. \tag{8.37}$$

Wykład 9

Różniczkowanie

9.1 Metoda z aproksymacją

W metodzie tej najpierw aproksymujemy funkcję, której pochodną chcemy znaleźć przy pomocy jednej z metod opisanych w poprzednim rozdziale:

$$f(x) \simeq \sum_{n=0}^{M} c_n \, \phi_n(x) \,, \tag{9.1}$$

gdzie $\phi_n(x)$ to znane i różniczkowalne funkcje bazowe, np. jednomiany x^n lub wielomiany Czebyszewa $T_n(x)$. Przyjmujemy, że pochodna tej funkcji to

$$\boxed{f'(x) \simeq \sum_{n=0}^{M} c_n \, \phi'_n(x)} \tag{9.2}$$

Podobnie postępujemy przy obliczniu wyższych pochodnych.

9.2 Metody z rozwinięciem Taylora

Ta metoda różniczkowania wykorzystują rozwinięcie Taylora funkcji. Zakładając, że rozwinięcie takie istnieje w otoczeniu punktu x, mamy

$$f(x+h) = f(x) + f'(x)\, h + f''(x)\, \frac{h^2}{2!} + f'''(x)\, \frac{h^3}{3!} + \mathcal{O}(h^4) \tag{9.3}$$

gdzie symbol $\mathcal{O}(h^4)$ oznacza resztę rzędu h^4, tzn

$$\lim_{h \to 0} \frac{\mathcal{O}(h^4)}{h^4} = \text{const} \,. \tag{9.4}$$

Aby obliczyć pierwszą pochodną wykorzystujemy wzór zapisany z dokładnością do członu liniowego w h:

$$f(x+h) = f(x) + f'(x)\, h + \mathcal{O}(h^2) \,.,$$

Stąd wynika

$$f'(x) = \frac{f(x+h)-f(x)}{h} + \mathcal{O}(h)\,. \tag{9.5}$$

Aby poprawić dokładność obliczanej w ten sposób pochodnej wykorzystujemy dwa rozwinięcia Taylora zapisane z dokładnością h^2:

$$\begin{aligned} f(x+h) &= f(x) + f'(x)\,h + f''(x)\,\frac{h^2}{2!} + \mathcal{O}(h^3) \\ f(x-h) &= f(x) - f'(x)\,h + f''(x)\,\frac{h^2}{2!} - \mathcal{O}(h^3)\,. \end{aligned} \tag{9.6}$$

Po odjęciu stronami wyrażenia z parzystymi potegami h upraszczają się i stąd dostajemy

$$\boxed{f'(x) = \frac{f(x+h)-f(x-h)}{2h} + \mathcal{O}(h^2)} \tag{9.7}$$

Znając więc wartośc funkcji w punktach sąsiednich $(x-h)$ oraz $(x+h)$, poprawiamy dokładność pierwszej pochodnej.

Dodając stronami wyrażenia (9.6) otrzymujemy wzór na drugą pochodną

$$\boxed{f''(x) = \frac{f(x+h)-2f(x)+f(x-h)}{h^2} + \mathcal{O}(h^2)} \tag{9.8}$$

Rząd reszty wynika z faktu kasowanie się wyrażeń z nieparzystymi potęgami h przy dodawaniu stronami.

9.2.1 Większa dokładność

Lepszą dokładność obliczanych pochodnych można uzyskać rozważając rozwinięcia Taylora zapisane z dokładnościa do piątej potęgi h dla wartości funkcji w czterech punktach: $f(x-2h)$, $f(x-h)$, $f(x+h)$, $f(x+2h)$.

Jako pożyteczne ćwiczenie należy udowodnić, że w takim przypadku dla pierwszej pochodnej otrzymujemy

$$f'(x) = \frac{f(x-2h)-8f(x-h)+8f(x+h)-f(x+2h)}{12h} + \mathcal{O}(h^4)\,. \tag{9.9}$$

Natomiast druga pochodna jest dana przez wyrażenie

$$f''(x) = \frac{-f(x-2h)+16f(x-h)-30f(x)+16f(x+h)-f(x+2h)}{12h^2} + \mathcal{O}(h^4)\,. \tag{9.10}$$

9.3 Wyższe pochodne

W podobny sposób można wyprowadzić wzory na wyższe pochodne, korzystając z wyprowadzonych wcześniej formuł dla niższych pochodnych. Na przykład, aby obliczyć trzecią pochodną odejmujemy od siebie dwa rozwinięcia

$$\begin{aligned} f(x+h) &= f(x) + f'(x)\,h + f''(x)\,\frac{h^2}{2!} + f'''(x)\,\frac{h^3}{3!} + \mathcal{O}(h^4) \\ f(x-h) &= f(x) - f'(x)\,h + f''(x)\,\frac{h^2}{2!} - f'''(x)\,\frac{h^3}{3!} + \mathcal{O}(h^4)\,, \end{aligned} \tag{9.11}$$

otrzymując

$$f'''(x) = 3\,\frac{f(x+h) - f(x-h)}{h^3} - 6\,\frac{f'(x)}{h^2} + \mathcal{O}(h^2)\,. \tag{9.12}$$

Zwróćmy uwagę, że podstawienie wzoru (9.7) w miejsce pierwszej pochodnej w powyższym wzorze daje resztę $\mathcal{O}(1)$, niezależną od odchylenia h. Jesteśmy więc zmuszeni do użycia dokładniejszego wzoru (9.9), prowadzącego do

$$\boxed{f'''(x) = \frac{-f(x-2h) + 2f(x-h) - 2f(x+h) + f(x+2h)}{2h^3} + \mathcal{O}(h^2)} \tag{9.13}$$

Wykład 10

Zera funkcji

Nie każde równanie (lub układ równań) można rozwiązać dokładnie. Na przykład, nie można podać ogólnych wzorów na rozwiązanie dowolnego równanie algebraicznego stopnia wyższego niż *cztery*.

Rozpatrzmy zagadnienie znajdowania pierwiastka równania

$$f(\bar{x}) = 0 \tag{10.1}$$

w pewnym przedziale. Zakładamy, że jest to pierwiastek jednokrotny, tzn. w jego otoczeniu

$$f(x) \sim (x - \bar{x})\,. \tag{10.2}$$

Prezentowane poniżej metody można stosować gdy znamy przedział, w którym znajduje się pierwiastek. Należy więc wcześniej określić taki przedział, na przykład rysując wykres funkcji $y = f(x)$.

10.1 Metoda połowienia

Zakładamy, że funkcja $f(x)$ jest ciągła w przedziale $[a, b]$. Zgodnie z twierdzeniem Bolzano-Cauchego, jeśli na końcach tego przedziału wartości funkcji mają przeciwne znaki,

$$f(a)\, f(b) < 0\,, \tag{10.3}$$

to wewnątrz przedziału znajduje się co najmniej jeden pierwiastek równania (10.1).

Zgodnie z uwagą z poprzedniego rozdziału zakładamy, że jest to pierwiastek jednokrotny. W pierwszym kroku dzielimy przedział $[a, b]$ na połowę

$$x_0 = \frac{a+b}{2}\,. \tag{10.4}$$

Jeżeli $f(x_0) = 0$ to x_0 jest poszukiwanym pierwiastkiem. W przeciwnym wypadku pierwiastek leży w tym z przedziałów

$$[a, x_0] \qquad \text{lub} \qquad [x_0, b]\,,$$

na końcach którego funkcja f ma przeciwny znak. Dzielimy ten przedział na połowę otrzymując x_1. Zauważmy, że długość nowego przedziału to

$$\Delta_1 = \frac{1}{2}(b-a)\,. \tag{10.5}$$

W wyniku wielokrotnego zastosowania tej procedury otrzymujemy pierwiastek $\bar{x}$ lub ciąg punktów x_n, będących środkiem przedziału o długości

$$\Delta_n = \frac{1}{2^n}(b-a) \tag{10.6}$$

Po dostatecznie dużej liczbie kroków długość takiego przedziału jest dowolnie mała. Zadając więc dokładność ϵ, przerywamy procedurę przy n takim, że

$$\Delta_n \leqslant 2\epsilon\,. \tag{10.7}$$

Poszukiwanym pierwiastkiem jest wtedy

$$\bar{x} = x_n \pm \epsilon\,, \tag{10.8}$$

Przyklad
Rozwiążemy równanie $\exp(-x) = x$. W tym celu poszukajmy zer funkcji

$$f(x) = x - \exp(-x)\,. \tag{10.9}$$

Zero znajduje się przedziale $[0,1]$ gdyż $f(0) = -1$ i $f(1) \approx 0.63$. Kolejne wartości środków przedziałow x_n wraz z długościami $\Delta_n/2$ to

n	x_n	$\Delta_n/2$
0	0.50000	0.50000
1	0.75000	0.25000
2	0.62500	0.12500
3	0.56250	0.06250
4	0.59375	0.03125
5	0.57812	0.01562
6	0.57031	0.00781
7	0.56641	0.00391
8	0.56836	0.00195

W 18 kroku otrzymujemy liczbę 0.56714, która już nie ulega zmianie przy zadanej liczbie pięciu cyfr po przecinku (dokładność $\epsilon < 10^{-5}$).

10.2 Metoda Newtona

Załóżmy, że f jest klasy C^2 w przedziale $[a,b]$ na końcach, którego zmienia znak. Ponadto, niech pochodne f' oraz f'' mają stały znak w całym przedziale.

Metoda Newtona obejmuje cztery przypadki będące kombinacją następujących warunków. Funkcja f jest

- rosnąca oraz wypukła ku dołowi ($f' > 0$, $f'' > 0$)
- rosnąca oraz wypukła ku górze ($f' > 0$), $f'' < 0$)
- malejąca oraz wypukła ku dołowi ($f' < 0$, $f'' > 0$)
- malejąca oraz wypukła ku górze ($f' < 0$, $f'' < 0$).

Przyjmując dla ustalenia uwagi, że obie pochodne są dodatnie, wystawmy styczną do wykresu funkcji w punkcie $x_0 = b$, w którym $f(x_0) > 0$, patrzy rysunek 10.1,

$$y - f(x_0) = f'(x_0)(x - x_0)\,. \tag{10.10}$$

Kładąc $y = 0$ otrzymujemy punkt przecięcia stycznej z osią x:

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}\,. \tag{10.11}$$

Udowodnimy, że $\bar{x} < x_1 < x_0$, gdzie $\bar{x}$ jest poszukiwanym pierwiastkiem.

Dowód
Górne ograniczenie wynika ze wzoru (10.11), gdyż $f(x_0)$ oraz $f'(x_0)$ są większe od zera. W dowodzie dolnego ograniczenia rozważmy rozwinięcia Taylora funkcji f wokół punktu x_0:

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2} f''(\xi)(x - x_0)^2\,,$$

gdzie $\xi \in (\bar{x}, x_0)$. Kładąc $x = \bar{x}$ i wykorzystując równość $f(\bar{x}) = 0$, wyliczymy

$$\bar{x} = \underbrace{x_0 - \frac{f(x_0)}{f'(x_0)}}_{x_1} - \frac{1}{2}\frac{f''(\xi)}{f'(x_0)}(\bar{x} - x_0)^2\,.$$

Stąd na podstawie założenia $f', f'' > 0$ otrzymujemy naszą tezę

$$\bar{x} - x_1 = -\frac{1}{2}\frac{f''(\xi)}{f'(x_0)}(\bar{x} - x_0)^2 < 0\,.$$

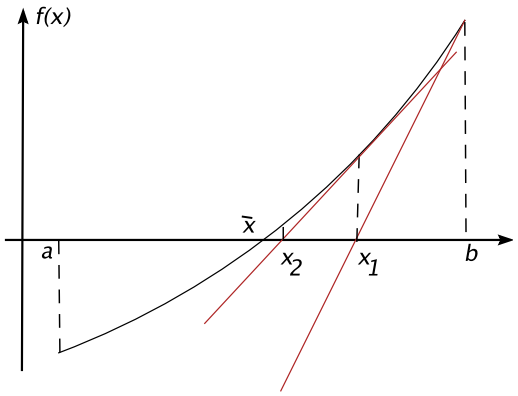
Powtórzmy procedurę, wystawiając styczną w punkcie $(x_1, f(x_1))$:

$$y - f(x_1) = f'(x_1)(x - x_1)\,. \tag{10.12}$$

Przecina ona oś x w punkcie

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}\,. \tag{10.13}$$

Punkt ten spełnia warunek $\bar{x} < x_2 < x_1$. Dolny warunek udowodnimy podobnie jak dla punktu x_1. Dowód górnego warunku polega na pokazaniu, że $f(x_1) > 0$.



Rys. 10.1: Ilustracja metody Newtona

W tym celu skorzystamy z twierdzenia Lagrange'a, które mówi, że istnieje punkt $\zeta \in (\bar{x}, x_1)$, dla którego zachodzi

$$f(x_1) - f(\bar{x}) = f'(\zeta)(x_1 - \bar{x})\,.$$

Ze względu na warunki $f(\bar{x}) = 0$ oraz $f'(\zeta) > 0$ dostajemy więc $f(x_1) > 0$.

Kolejne kroki procedury prowadzą do relacji rekurencyjnej definiującej kolejne wyrazy ciągu przybliżeń x_n poszukiwanego zera funkcji:

$$\boxed{x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}\,.} \tag{10.14}$$

Otrzymany ciąg punktów jest malejący i ograniczony od dołu. Z twierdzenia Cauchego wynika, że istnieje granica tego ciągu g. Tak więc z relacji (10.14) wynika równość

$$g = g - \frac{f(g)}{f'(g)}\,. \tag{10.15}$$

Stąd wnioski

$$f(g) = 0 \qquad => \qquad g = \bar{x}\,. \tag{10.16}$$

Procedura Newtona jest więc zbieżna do poszukiwanego zera funkcji. W praktyce procedurę przerywamy gdy

$$|x_{n+1} - x_n| < \epsilon\,. \tag{10.17}$$

Przyklad

Metoda Newtona jest znacznie szybciej zbieżna niż metoda połowienia. Dla przykładu z poprzedniego rozdziału otrzymujemy wynik po czterech krokach (pamiętamy, że $x_0 = 1$):

n	x_n	$\lvert x_n - x_{n-1}\rvert$
1	0.53788	0.46212
2	0.56699	0.02910
3	0.56714	0.00016
4	0.56714	0.00000

10.3 Metoda siecznych

W metodzie Newtona konieczna jest znajomość pochodnej f'. W metodzie siecznych unikamy tego warunku przybliżając pochodną w wyrażeniu (10.14) przez iloraz różnicowy

$$f'(x_n) \simeq \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}\,. \tag{10.18}$$

W ten sposób otrzymujemy

$$\boxed{x_{n+1} = x_n - f(x_n)\,\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}\,.} \tag{10.19}$$

Zatem do wyznaczenia $(n+1)$ przybliżenia pierwiastka $\bar{x}$ wykorzystuje się punkty $(x_n, f(x_n))$ oraz $(x_{n-1}, f(x_{n-1}))$, przez które przeprowadza się sieczną

$$y - f(x_n) = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}\,(x - x_n)\,. \tag{10.20}$$

Przecinaja ona oś x w punkcie x_{n+1} zadanym wzorem (10.19). Aby rozpocząć procedurę konieczne jest więc wybranie *dwóch* punktów startowych x_0 i x_1. W omawianej przez nas przykładzie wybieramy $x_0 = b$ oraz $x_1 = x_0 - \Delta$ z mała wartością $\Delta \ll x_0$.

Metoda siecznych może nie być zbieżna, na przykład, gdy początkowe przybliżenia nie leżą dostatecznie blisko szukanego pierwiastka. Jako dodatkowe kryterium przerwania iteracji, oprócz wartości różnic $|x_{n+1} - x_n|$, należy przyjąć wartości $|f(x_n)|$, tak by tworzyły one ciąg malejący w końcowej fazie obliczeń.

Wracając do przykładu, w metodzie siecznych otrzymujemy wynik 0.56714 z błędem mniejszym niż 10^{-5} po pięciu krokach

10.4 Metoda fałszywej prostej (falsi)

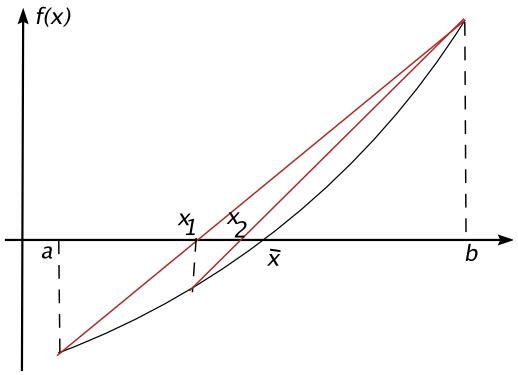
W metodzie *falsi* (fałszywej prostej) znajomość f' także nie jest potrzebna. Przy przyjętych założeniach $(f', f'' > 0)$ przeprowadzamy w pierwszą sieczną przez punkty $(b, f(b))$ oraz $(a, f(a))$:

$$y - f(a) = \frac{f(b) - f(a)}{b - a}\,(x - a)\,. \tag{10.21}$$

Przecina ona oś x w punkcie x_1, patrz rysunek 10.2,

$$x_1 = a - f(a)\,\frac{b - a}{f(b) - f(a)}\,. \tag{10.22}$$

Punkt $(b, f(b))$ jest punktem stałym wszystkich cięciw i w następnym przybliżeniu przeprowadzamy cięciwę przez punkt $(x_1, f(x_1)$, otrzymując x_2 jako



Rys. 10.2: Ilustracja metody falsi

punkt jej przecięcia z osią x. W ogólności przeprowadzamy cięciwy przez punkty $(x_n, f(x_n))$:

$$y - f(x_n) = \frac{f(b) - f(x_n)}{b - x_n}\,(x - x_n)\,. \tag{10.23}$$

Przecinają one oś x w punkcie:

$$\boxed{x_{n+1} = x_n - f(x_n)\,\frac{b - x_n}{f(b) - f(x_n)}} \tag{10.24}$$

Można pokazać, że metoda ta prowadzi do ciągu wartości x_n zbieżnych do granicy będącej jednokrotnym pierwiastkiem równania $f(\bar{x}) = 0$.

Przyklad
W naszym przykładzie otrzymujemy w metodzie falsi:

n	x_n	$\lvert x_n - x_{i-1}\rvert$
1	0.61270	0.61270
2	0.56384	0.04886
3	0.56739	0.00355
4	0.56713	0.00026
5	0.56714	0.00002
6	0.56714	0.00000

Jak widać metoda falsi jest stosunkowo wolno zbieżna.

Wykład 11

Całkowanie

11.1 Kwadratury

Rozważmy jednowymiarową całkę na przedziale $[a,b]$

$$I(f) = \int_a^b f(x)\,dx \tag{11.1}$$

Podzielmy przedział całkowania na N *równych* odcinków o długości

$$h = \frac{b-a}{N} \tag{11.2}$$

i wyznaczonych przez kolejne $N+1$ punktów

$$x_k = a + kh \qquad\qquad k = 0, 1, \ldots, N\,. \tag{11.3}$$

Zauważmy, że $x_0 = a$ oraz $x_N = b$ są końcami przedziałów. Wtedy

$$I(f) = \sum_{i=1}^{N} \int_{x_{i-1}}^{x_i} f(x)\,dx\,. \tag{11.4}$$

Należy więc przybliżyć całkę w każdym z podprzedziałów $[x_{i-1}, x_i]$. Otrzymany wynik nazywa się kwadraturą i ma ogólną postać:

$$\boxed{I(f) \simeq \sum_{i=0}^{N} A_i\, f(x_i)} \tag{11.5}$$

A_i to współczynniki (wagi) kwadratury, natomiast punkty x_i to jej węzły.

11.2 Metoda prostokatów

W metodzie prostokątów

$$\int_{x_{i-1}}^{x_i} f(x)\,dx \simeq h\,f(x_i) \tag{11.6}$$

i wtedy, wprowadzając oznacznie $f_i \equiv f(x_i)$,

$$\boxed{\int_a^b f(x)\,dx \simeq h\,(f_1 + f_1 + \ldots + f_N)} \tag{11.7}$$

Otrzymany wynik odpowiada przybliżeniu funkcji f w każdym z podprzedziałow poprzez funkcję stałą będącą wartością funkcji w prawym końcu każdego podprzedziału.

Możemy również wykorzystać wartości funkcji w lewych końcach i wtedy

$$\int_a^b f(x)\,dx \simeq h\,(f_0 + f_1 + \ldots + f_{N-1}) \tag{11.8}$$

Wartości funkcji f_i mogą być również wzięte w środkach podprzedziałów

$$f_i = f\left(\frac{x_i + x_{i+1}}{2}\right). \tag{11.9}$$

11.3 Metoda trapezów

W metodzie trapezów otrzymujemy

$$\int_{x_{i-1}}^{x_i} f(x)\,dx \simeq \frac{h}{2}\,\{f_{i-1} + f_i\} \tag{11.10}$$

i wtedy

$$\boxed{\int_a^b f(x)\,dx \simeq \frac{h}{2}\left\{f_0 + 2(f_1 + \ldots + f_{N-1}) + f_N\right\}} \tag{11.11}$$

Prawa strona wzoru (11.10) to pole trapezu gdy obie wartości funkcji f_{i-1} i f_i są dodatnie, lub minus pole, gdy obie są ujemne. W przypadku, gdy wartości różnią się znakiem przedstawiona interpretacja geometryczna nie jest już prawdziwa. Można jednak sformułować problem ogólnie korzystając z liniowej interpolacji Lagrange'a w każdym z podprzedziałow (x_{i-1}, x_i):

$$f(x) \simeq f_{i-1}\,\frac{x - x_i}{x_{i-1} - x_i} + f_i\,\frac{x - x_{i-1}}{x_i - x_{i-1}}. \tag{11.12}$$

Całkując bowiem (11.12) i pamiętając, że $x_i - x_{i-1} = h$, otrzymujemy wzór (11.10) niezależnie od znaku wartości funkcji w krańcach podprzedziałów:

$$\frac{1}{h}\int_{x_{i-1}}^{x_i} \{f_i\,(x-x_{i-1}) - f_{i-1}\,(x-x_i)\}\,dx = \frac{h}{2}\,\{f_{i-1}+f_i\}\,.$$

Łatwo uogólnić tę metodę, przybliżając funkcję podcałkową przy pomocy większej liczby punktów. Przykładem jest metoda Simpsona.

11.4 Metoda parabol Simpsona

W metodzie tej dzielimy przedział $[a,b]$ na parzystą liczbę $2N$ przedziałow o równej długości:

$$h = \frac{b-a}{2N}. \tag{11.13}$$

W każdej sąsiedniej parze przedziałów wyznaczonych przez $(x_{2i-2}, x_{2i-1}, x_{2i})$ stosujemy interpolację Lagrange'a:

$$\begin{aligned} f(x) \;&\simeq\; f_{2i-2}\,\frac{(x-x_{2i-1})(x-x_{2i})}{(x_{2i-2}-x_{2i-1})(x_{2i-2}-x_{2i})} \\ &+\; f_{2i-1}\,\frac{(x-x_{2i-2})(x-x_{2i})}{(x_{2i-1}-x_{2i-2})(x_{2i-1}-x_{2i})} \\ &+\; f_{2i}\,\frac{(x-x_{2i-2})(x-x_{2i-1})}{(x_{2i}-x_{2i-2})(x_{2i}-x_{2i-1})}\,. \end{aligned} \tag{11.14}$$

Pamiętając, że

$$(x_{2i}-x_{2i-1}) = (x_{2i-1}-x_{2i-2}) = h \tag{11.15}$$

otrzymujemy po wykonaniu całkowania

$$\int_{x_{2i-2}}^{x_{2i}} f(x)\,dx \simeq \frac{h}{3}\,\{f_{2i-2}+4f_{2i-1}+f_{2i}\}\,. \tag{11.16}$$

Stąd przybliżony wzór dla wartości całki:

$$\boxed{\int_a^b f(x)\,dx \simeq \frac{h}{3}\left\{f_0+4(f_1+\ldots+f_{2N-1})+2(f_2+\ldots+f_{2N-2})+f_{2N}\right\}} \tag{11.17}$$

11.5 Błąd przybliżeń

Zdefiniujmy błąd obliczenia całki, rozumiany jako różnicę pomiędzy wartością dokładną a przybliżoną

$$\epsilon \equiv I(f) - \sum_{i=0}^{N} A_i\,f(x_i)\,. \tag{11.18}$$

Można pokazać (podręcznik [1]), że błąd ten jest ograniczony od góry.

W metodzie prostokątów:

$$|\epsilon| < (b-a)\frac{h}{2}\sup_{x\in[a,b]}|f'(x)| \tag{11.19}$$

Całkowanie tą metodą jest więc dokładne dla funkcji stałej – wielomianu stopnia zerowego.

W metodzie trapezów:

$$|\epsilon| < (b-a)\frac{h^2}{12}\sup_{x\in[a,b]}|f^{(2)}(x)| \tag{11.20}$$

Tym razem całkowanie jest dokładne dla wszystkich wielomianów stopnia $\leqslant 1$, dla których znika druga pochodna.

W metodzie parabol:

$$|\epsilon| < (b-a)\frac{h^4}{180}\sup_{x\in[a,b]}|f^{(4)}(x)| \tag{11.21}$$

Metoda ta jest dokładna dla każdego wielomianu stopnia $\leqslant 3$, dla którego znika czwarta pochodna.

Najwyższy stopień wielomianu, dla którego metoda całkowania *nie jest dokładna* nazywamy *rzędem metody*. Tak więc rząd przedstawionych kwadratur wynosi odpowiednio: $1, 2, 4$. Pojęcie to odgrywa podstawową rolę przy konstrukcji *kwadratur Gaussa*.

Zauważmy na koniec, że przy tej samej długości podprzedziałów $h < 1$, najlepsza parametrycznie jest metoda parabol Simpsona.

Wykład 12

Kwadratury Gaussa

Naszym celem jest znalezienie ogólnej metody przybliżonego obliczania całek

$$I(f) = \int_a^b f(x)\, w(x)\, dx \tag{12.1}$$

Granice całkowania mogą być równe $\pm\,\infty$. Dodatnia funkcja $w(x) \geqslant 0$ jest nazywana *wagą*. Może ona zawierać całkowalne osobliwości, które wyłączyliśmy z funkcji podcałkowej. Na przykład,

$$w(x) = \frac{1}{\sqrt{1-x^2}}, \tag{12.2}$$

dla całek w przedziale $[-1, 1]$ lub

$$w(x) = \mathrm{e}^{-x^2}\,. \tag{12.3}$$

dla całek niewłaściwych z granicami całkowania $\pm\,\infty$. Ostatnia waga zapewnia zbieżność całki dla szerokiej klasy funkcji f.

Wybierzmy N węzłów w przedziale całkowania:

$$\{x_0, x_1, \ldots x_{N-1}\}\,. \tag{12.4}$$

Posłużą one do skonstruowania interpolacji Lagrange'a funkcji f:

$$f(x) \simeq \sum_{k=0}^{N-1} f(x_k)\, L_k(x)\,, \qquad\qquad L_k(x) = \prod_{i=0}^{N-1}{}' \left(\frac{x - x_i}{x_k - x_i}\right), \tag{12.5}$$

gdzie symbol $(\,)'$ oznacza, że w iloczynie pomijamy składnik z $i = k$. Podstawiając do całki otrzymamy wzór na kwadraturę:

$$I(f) \simeq \sum_{k=0}^{N-1} A_k\, f(x_k)\,, \qquad\qquad A_k = \int_a^b L_k(x)\, w(x)\, dx\,. \tag{12.6}$$

Współczynniki kwadratury A_k zależą od wyboru węzłów poprzez wielomiany $L_k(x)$.

Zauważmy, że kwadratura (12.6) jest dokładna dla wielomianów stopnia $< N$, gdyż interpolacja Lagrange'a jest dokładna dla każdego z tych wielomianów.

12.1 Rząd kwadratury

Rząd kwadratury jest miarą jej dokładności.

Definicja
Mówimy, że kwadratura jest rzędu $r = N$ jeśli jest dokładna dla wszystkich wielomianów stopnia $< N$ oraz istnieje wielomian stopnia N, dla którego kwadratura nie jest dokładna.

Innymi słowy, rząd kwadratury jest określony przez najniższy stopień wielomianu, dla którego kwadratura nie jest dokładna.

Tak więc dla kwadratury (12.6) zachodzi: $r \geqslant N$. Łatwo udowodnić, że dla każdej kwadratury z N węzłami $r \leqslant 2N$. Istnieje bowiem wielomian stopnia $2N$:

$$W(x) = [(x-x_0)(x-x_1)\ldots(x-x_{N-1})]^2 \geqslant 0, \tag{12.7}$$

dla którego każda kwadratura nie jest dokładna. Zachodzi bowiem

$$I(W) = \int_a^b W(x)\, w(x)\, dx > 0\,, \qquad \sum_{k=0}^{N-1} A_k\, W(x_k) = 0\,. \tag{12.8}$$

Ostatecznie, rząd kwadratury (12.6) zawarty jest w przedziale

$$\boxed{N \leqslant r \leqslant 2N} \tag{12.9}$$

Węzły $\{x_k\}$ w kwadraturze Gaussa są tak wybrane by jej rząd był równy maksymalnej wartości $2N$. Można to osiągnąć przy pomocy wielomianów ortogonalnych.

12.2 Wielomiany ortogonalne

Wielomiany określone na odcinku $[a, b]$

$$\{P_0(x)\,,\ P_1(x), \ldots P_N(x), \ldots\} \tag{12.10}$$

tworzą układ ortogonalny z wagą $w(x) \geqslant 0$, jeśli dla $i \neq j$ zachodzi

$$\boxed{\langle P_i\,|P_j\rangle \equiv \int_a^b P_i(x)\, P_j(x)\, w(x)\, dx = 0} \tag{12.11}$$

Wielomiany ortogonalne mają bardzo ważną własność wyrażoną w następującym twierdzeniu (dowód w podręczniku [1]).

Wielomiany	P_n	$w(x)$	$[a,b]$	Rekurencja
Legendre'a	P_n	1	$[-1,1]$	$(j+1)P_{j+1} = (2j+1)xP_j - jP_{j-1}$
Czebyszewa	T_n	$1/\sqrt{1-x^2}$	$[-1,1]$	$T_{j+1} = 2xT_j - T_{j-1}$
Hermite'a	H_n	e^{-x^2}	$[-\infty,\infty]$	$H_{j+1} = 2xH_j - 2jH_{j-1}$
Laguerre'a	L_n	e^{-x}	$[0,\infty]$	$jL_{j+1} = (2j+1-x))L_j - jL_{j-1}^{(\alpha)}$

Twierdzenie
Każdy wielomian ortogonalny $P_n(x)$ ma dokładnie n jednokrotnych pierwiastków w przedziale $[a,b]$.

Poniżej w tabelce podajemy przykłady wielomianów ortogonalnych, podając wagę $w(x)$ oraz przedział $[a,b]$ w iloczynie skalarnym (12.11), a także relację rekurencyjną, przy pomocy której można obliczyć kolejne wielomiany.

Przyklad
Dla wielomianów Legendre'a zachodzi

$$\int_{-1}^{1} P_n(x)\, P_m(x)\, dx = \frac{2}{2n+1}\, \delta_{nm}\,, \tag{12.12}$$

a kolejne wielomiany konstruowane przy pomocy relacji rekurencyjnej to

$$P_0(x) = 1\,, \qquad P_1(x) = x\,, \qquad P_2(x) = \tfrac{1}{2}(3x^2-1)\,. \tag{12.13}$$

12.3 Kwadratura Gaussa

Rozważmy kwadraturę (12.6):

$$I(f) \simeq \sum_{k=0}^{N-1} A_k\, f(x_k)\,, \qquad\qquad A_k = \int_a^b L_k(x)\, w(x)\, dx.$$

Zależy ona od dwóch elementów, wagi $w(x)$ oraz wyboru N punktów interpolacyjnych $\{x_k\}$. Problem polega na takim wybraniu tych punktów przy ustalonej wadze by rząd kwadratury był maksymalny, równy $2N$. Znajdując je, otrzymujemy kwadraturę Gaussa.

Twierdzenie
N węzłów $\{x_k\}$ w kwadraturze Gaussa jest zadane przez zera wielomianu $P_N(x)$, należącego do zbioru wielomianów ortogonalnych względem wagi $w(x)$, tzn.

$$\boxed{P_N(x_k) = 0} \tag{12.14}$$

dla $k = 0, 1, \ldots (N-1)$. Dla tych węzłów rząd kwadratury wynosi $2N$.

Dowód
Pokażemy, że kwadratura Gaussa jest dokładna dla dowolnego wielomianu $W(x)$ stopnia $< 2N$. Dzieląc go przez $P_N(x)$ otrzymamy

$$W(x) = W_1(x)\,P_N(x) + W_2(x)\,, \tag{12.15}$$

gdzie $W_1(x)$, $W_2(x)$ są wielomianami stopnia $< N$. Możemy wtedy zapisać $W_1(x)$ jako kombinację liniową wielomianów ortogonalnych $P_0, P_1, \ldots, P_{N-1}$

$$W_1(x) = \sum_{i=0}^{N-1} c_i\,P_i(x) \tag{12.16}$$

i wtedy

$$W(x) = \sum_{i=0}^{N-1} c_i\,P_i(x)\,P_N(x) + W_2(x) \tag{12.17}$$

Całkując z wagą $w(x)$, otrzymamy

$$\int_a^b W(x)\,w(x)\,dx = \sum_{i=0}^{N-1} c_i \int_a^b P_i(x)\,P_N(x)\,w(x)\,dx + \int_a^b W_2(x)\,w(x)\,dx\,.$$

Suma całek znika na mocy ortogonalności układu wielomianów P_i, tak więc

$$\int_a^b W(x)\,w(x)\,dx = \int_a^b W_2(x)\,w(x)\,dx\,. \tag{12.18}$$

Jak wiemy, kwadratura (12.6) jest dokładna dla wielomianów stopnia $< N$, stąd

$$\int_a^b W_2(x)\,w(x)\,dx = \sum_{k=0}^{N-1} A_k\,W_2(x_k) \tag{12.19}$$

Wyliczając $W_2(x)$ z relacji (12.15)

$$W_2(x) = W(x) - W_1(x)\,P_N(x)\,, \tag{12.20}$$

a następnie podstawiając po prawej stronie (12.19), otrzymamy

$$\sum_{k=0}^{N-1} A_k\,W_2(x_k) = \sum_{k=0}^{N-1} A_k\,W(x_k) - \underbrace{\sum_{k=0}^{N-1} A_k\,W_1(x_k)\,P_N(x_k)}_{0}\,. \tag{12.21}$$

Ostatnia suma jest równa zeru, gdyż $P_N(x_k) = 0$. Ostatecznie udowodniliśmy,

$$\int_a^b W(x)\,w(x)\,dx = \sum_{k=0}^{N-1} A_k\,W(x_k)\,, \tag{12.22}$$

tzn. kwadratura Gaussa jest dokładna dla dowolnego wielomianu $W(x)$ stopnia $< 2N$. Jej rząd wynosi więc $2N$.

12.4 Współczynniki kwadratury Gaussa

Współczynniki A_k kwadratury Gaussa są zadane przez [2]:

$$\boxed{A_k = \frac{\langle P_{N-1}\,|P_{N-1}\rangle}{P_{N-1}(x_k)\,P'_N(x_k)}} \tag{12.23}$$

gdzie $k = 0, 1, \ldots (N-1)$, natomiast P'_N to pochodna wielomianu P_N.

Twierdzenie
Wszystkie współczynniki kwadratury Gaussa są dodatnie.

Dowód
Rozważmy kwadrat wielomianu $L_i(x)$ z interpolacji Lagrange'a (12.5), stopnia $2N$. Kwadratura Gaussa jest dokładna dla tego wielomianu i stąd

$$\int_a^b w(x)\,[L_i(x)]^2\,dx = \sum_{k=0}^{N-1} A_k\,[L_i(x_k)]^2 = \sum_{k=0}^{N-1} A_k\,\delta_{ik} = A_i\,. \tag{12.24}$$

Całka po lewej stronie jest dodatnia i stąd $A_i > 0$.

12.5 Przykład kwadratury Gaussa

Skonstruujemy kwadraturę Gaussa-Legendre'a rzędu $r = 4$. Mamy wtedy do czynienia z dwoma węzłami $\{x_0, x_1\}$ i wtedy

$$\int_{-1}^{1} f(x)\,dx \approx A_0 f(x_0) + A_1 f(x_1)\,. \tag{12.25}$$

Węzły są zerami wielomianu Legendre'a P_2:

$$P_2(x) = \tfrac{1}{2}(3x^2 - 1) = 0 \tag{12.26}$$

i rozwiązaniami są:

$$x_0 = -\frac{1}{\sqrt{3}}\,, \qquad\qquad x_1 = \frac{1}{\sqrt{3}}\,. \tag{12.27}$$

Współczynniki A_i można obliczyć ze wzoru (12.23). Prościej jest wykorzystać fakt, że kwadratura (12.25) jest dokładna dla wielomianów stopnia < 4, w szczególności dla $W_0(x) = 1$ oraz $W_1(x) = x$. Wtedy

$$A_0 + A_1 \;=\; \int_{-1}^{1} dx = 2$$

$$-\frac{A_0}{\sqrt{3}} + \frac{A_1}{\sqrt{3}} \;=\; \int_{-1}^{1} x\,dx = 0\,.$$

Stąd współczynniki kwadratury $A_1 = A_2 = 1$ i ostatecznie

$$\int_{-1}^{1} f(x)\,dx \approx f(-\tfrac{1}{\sqrt{3}}) + f(\tfrac{1}{\sqrt{3}})\,. \tag{12.28}$$

Wykład 13

Równania różniczkowe

13.1 Równania zwyczajne pierwszego rzędu

Równanie różniczkowe zwyczajne rzędu pierwszego ma postać

$$\frac{dy}{dt} = F(t, y)\,. \tag{13.1}$$

Funkcja $y = y(t)$ jest rozwiązaniem jeśli po podstawieniu do (13.1) otrzymujemy tożsamość.

$$\frac{dy(t)}{dt} = F(t, y(t))\,. \tag{13.2}$$

Rozwiązanie jest wyznaczone jednoznacznie poprzez zadanie *warunku początkowego*:

$$y(t_0) = y_0\,. \tag{13.3}$$

Przyklad
Rozwiążmy równanie

$$\frac{dy}{dt} = y \tag{13.4}$$

z warunkiem początkowym $y(t_0) = y_0$. Rozwiązaniem ogólnym jest funkcja

$$y(t) = C\exp(t)\,, \tag{13.5}$$

co łatwo sprawdzić poprzez podstawienie do równania. Dla chwili początkowej t_0 otrzymujemy

$$y(t_0) = C\exp(t_0) = y_0\,. \tag{13.6}$$

Stąd wynika, że

$$C = y_0\exp(-t_0) \tag{13.7}$$

i rozwiązaniem spełniającym warunek początkowy jest

$$y(t) = y_0\exp(t - t_0)\,. \tag{13.8}$$

13.2 Układ równań różniczkowych

Układ równań różniczkowych 1. rzędu ma następującą postać

$$\begin{aligned} \frac{dy_1}{dt} &= F_1(t, y_1, y_2, \dots y_n) \\ \frac{dy_2}{dt} &= F_2(t, y_1, y_2, \dots y_n) \\ &\dots \\ \frac{dy_n}{dt} &= F_n(t, y_1, y_2, \dots y_n)\,. \end{aligned} \tag{13.9}$$

Wprowadzając notację wektorową

$$\mathbf{y} = (y_1, y_2, \dots y_n)\,, \qquad \mathbf{F} = (F_1, F_2, \dots F_n) \tag{13.10}$$

układ ten możemy zapisać w formie analogocznej do równania (13.1)

$$\frac{d\mathbf{y}}{dt} = \mathbf{F}(t, \mathbf{y})\,. \tag{13.11}$$

Rozwiązanie układu równanń $\mathbf{y} = \mathbf{y}(t)$ jest wyznaczone jednoznacznie poprzez warunek początkowy

$$\mathbf{y}(t_0) = (y_1(t_0), y_2(t_0), \dots, y_n(t_0)) \equiv \mathbf{y}_0\,. \tag{13.12}$$

13.3 Równania wyższych rzędów

Równanie różniczkowe zwyczajne rzędu n ma postać

$$\frac{d^n y}{dt^n} = F(t, y, y^{(1)}, y^{(2)}, \dots y^{(n-1)})\,, \tag{13.13}$$

gdzie $y^{(k)}$ jest k-tą pochodną. Rozwiązanie $y = y(t)$ jest wyznaczone jednoznacznie przez warunki początkowe

$$y(t_0) = y_{10}\,, \qquad y^{(1)}(t_0) = y_{20}\,, \qquad \dots \qquad y^{(n-1)}(t_0) = y_{n0}\,. \tag{13.14}$$

Każde równanie różniczkowe zwyczajne rzędu n można zapisać jako układ n równań 1. rzędu. Wprowadźmy bowiem oznaczenia

$$y_1 = y\,, \qquad y_2 = y^{(1)}\,, \qquad \dots \qquad y_n = y^{(n-1)}\,. \tag{13.15}$$

Wtedy równanie (13.13) można zapisać w równoważnej formie układu równań 1. rzędu:

$$\begin{aligned} \frac{dy_1}{dt} &= y_2 \\ \frac{dy_2}{dt} &= y_3 \\ &\dots \\ \frac{dy_n}{dt} &= F(t, y_1, y_2, \dots y_n)\,. \end{aligned} \tag{13.16}$$

Z obserwacji tej wynika wniosek, że równanie (13.1) jest podstawowym obiektem zainteresowań. Rozwinięcie metod przybliżonego rozwiązywania tego równania pozwala na znalezienie rozwiązań dla układu równań (13.11), a tym samym dla równań wyższych rzędów (13.13).

Przyklad
Rozważmy układ trzech równań Newtona (masa $m = 1$) drugiego rzędu

$$\frac{d^2\mathbf{r}}{dt^2} = \mathbf{F}\left(t, \mathbf{r}, \frac{d\mathbf{r}}{dt}\right). \tag{13.17}$$

Definiując $\mathbf{v} = d\mathbf{r}/dt$ dostajemu układ sześciu równań 1. rzędu na poszukiwane funkcje $(\mathbf{r}(t), \mathbf{v}(t))$:

$$\begin{aligned} \frac{d\mathbf{r}}{dt} &= \mathbf{v} \\ \frac{d\mathbf{v}}{dt} &= \mathbf{F}(t, \mathbf{r}, \mathbf{v}). \end{aligned} \tag{13.18}$$

13.4 Metody Eulera

Prezentowane tutaj metody znajdowania przybliżonych rozwiązań równania (13.1) bazują na dyskretyzacji przedziału czasu $[t_0, T]$, w którym chcemy znaleźć rozwiązanie

$$t_0 < t_1 \ldots < t_{n-1} < t_n \ldots < T. \tag{13.19}$$

Załóżmy, że punkty czasowe są równoodległe:

$$t_n - t_{n-1} = \tau\,., \tag{13.20}$$

Omawiane metody dostarczają rekurencji wiążącej wartość rozwiązania y_{n+1} w chwili t_{n+1} z wartościami rozwiązania y_n w chwili wcześniejszej t_n:

$$y_{n+1} = f(t_n,\, y_n). \tag{13.21}$$

Wartość $y_0 = y(t_0)$ jest zadana przez warunek początkowy (13.3). Bardzo ważnym zagadnieniem tak określonych metod jest numeryczna stabilność rozwiązania dla dużych czasów T.

Punktem wyjścia metod Eulera jest równanie otrzymane po scałkowaniu po czasie obu stron równania (13.1) w przedziale (t_n, t_{n+1}):

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} F(t, y(t))\, dt. \tag{13.22}$$

Ponieważ nie znamy $y(t)$ w funkcji podcałkowej musimy zastosować metodę przybliżoną obliczenia tej całki. Wykorzystując wzór (11.6) na całkowanie metodą prostokątów,

$$\int_{t_n}^{t_{n+1}} dt\, F(t, y(t)) \simeq \tau\, F(t_n, y_n)\,., \tag{13.23}$$

otrzymujemy relację rekurencyjną w metodzie Eulera:

$$\boxed{y_{n+1} = y_n + \tau\, F(t_n, y_n)} \tag{13.24}$$

Zauważmy, że moglibyśmy przybliżyć całkę (13.23) przez wartość funkcji dla górnej granicy:

$$y_{n+1} = y_n + \tau\, F(t_{n+1}, y_{n+1})\,. \tag{13.25}$$

Pojawia się jednak w tym momencie problem, gdyż nieznane y_{n+1} występuje po obu stronach równania. Należałoby więc użyć dodatkowo metody poszukiwania zer z rozdziału 10. Jest to dość niepraktyczne, chociaż równanie (13.25) może być lepsze ze względu na stabilność numeryczną rozwiązania. Zagadnienie to zostanie omówione w następnym rozdziale.

13.5 Metoda przewidź i popraw

Przybliżmy całkę w równaniu (13.22) stosując metodą trapezów (11.10):

$$\int_{t_n}^{t_{n+1}} F(t, y(t))\, dt \simeq \frac{\tau}{2}\left(F(t_n, y_n) + F(t_{n+1}, y_{n+1})\right). \tag{13.26}$$

Stąd

$$y_{n+1} = y_n + \frac{\tau}{2}\left\{F(t_n, y_n) + F(t_{n+1}, y_{n+1})\right\}. \tag{13.27}$$

Podobnie jak we wzorze (13.25), nieznana wartość y_{n+1} znajduje się po obu stronach równania rekurencyjnego. Należy więc wykorzystać jedną z metod znajdowania zer funkcji z rozdziału 10.

Alternatywną metodą jest dwustopniowa procedura. Najpierw określamy *przewidywą* wartość y_{n+1} korzystając z metody Eulera (13.24):

$$\boxed{\bar{y}_{n+1} = y_n + \tau\, F(t_n, y_n)} \tag{13.28}$$

a następnie obliczamy *poprawioną* wartość:

$$\boxed{y_{n+1} = y_n + \frac{\tau}{2}\left\{F(t_n, y_n) + F(t_{n+1}, \bar{y}_{n+1})\right\}} \tag{13.29}$$

Stąd nazwa metody *przewidź i popraw*.

13.6 Stabilność numeryczna rozwiązań

Rozważmy równanie różniczkowe

$$\frac{dy}{dt} = -\lambda y\,, \qquad\qquad \lambda > 0\,. \tag{13.30}$$

Rozwiązanie dokładne z warunkiem początkowym $y(0) = y_0$ to

$$y(t) = y_0\,\mathrm{e}^{-\lambda t}\,. \tag{13.31}$$

Dąży ono do zera dla $t \to \infty$.

Szukając rozwiązania metodą Eulera (13.24), otrzymujemy

$$y_{n+1} = y_n - \lambda\tau\, y_n = (1-\lambda\tau)\, y_n\,, \tag{13.32}$$

co prowadzi do następującej relacji dla kolejnych chwil $t_n = n\,\tau$:

$$y_n = (1-\lambda\tau)^n\, y_0\,. \tag{13.33}$$

Powyższe rozwiąznie dobrze przybliża rozwiązanie dokładne (13.31) gdy zachodzi

$$|1-\lambda\tau| < 1\,. \tag{13.34}$$

Warunek ten oznacza, że krok czasowu musi być dostatecznie mały

$$\tau < \frac{2}{\lambda}\,. \tag{13.35}$$

W przeciwnym przypadku $|y_n| \to \infty$ dla rosnących n.

Najprostszym rozwiązaniem dla tego problemu (oprócz dobrania $\tau \ll 2/\lambda$) jest skorzystanie z drugiego wzoru Eulera (13.25). Wtedy otrzymujemy dla naszego równania

$$y_{n+1} = y_n - \lambda\tau\, y_{n+1}\,, \tag{13.36}$$

co prowadzi do

$$y_{n+1} = \frac{y_n}{1+\lambda\tau} = \left(\frac{1}{1+\lambda\tau}\right)^{n+1} y_0\,. \tag{13.37}$$

Wartość $1/(1+\lambda\tau) < 1$ i metoda ta jest stabilna niezależnie od kroku τ oraz wartości λ.

13.7 Równania typu stiff

Rozważmy **układ** dwóch równań różniczkowych

$$\frac{d\mathbf{y}}{dt} = -\mathbf{A}\,\mathbf{y} \tag{13.38}$$

z macierzą $\mathbf{A}$ o dodatnich wartościach własnych λ_1 oraz λ_2. Rozwiązanie jest sumą dwóch rozwiązań:

$$\mathbf{y}(t) = \mathrm{e}^{-\lambda_1 t}\mathbf{y}_1 + \mathrm{e}^{-\lambda_2 t}\mathbf{y}_2\,. \tag{13.39}$$

Jeżeli $\lambda_2 \gg \lambda_1$ to drugie rozwiązanie szybko dąży do zera i przestaje być istotne.

W numerycznej realizacji drugie rozwiązanie może jednak odgrywać kluczową rolę, prowadząc do niestabilnego zachowania całego rozwiązania. Stosując metodę Eulera (13.24), otrzymujemy

$$\mathbf{y}_n = (1-\lambda_1\tau)^n\,\mathbf{y}_1 + (1-\lambda_2\tau)^n\,\mathbf{y}_2\,. \tag{13.40}$$

Jeśli krok czasowy $\tau > 2/\lambda_2$ to wyrażenie $|1-\lambda_2\tau| > 1$, co prowadzi do rosnących i oscylujących wartości $\mathbf{y}_n$ gdy $n \to \infty$. Nawet dla $\tau \leqslant 2/\lambda_2$ rozwiązanie zachowuje charakter oscylacyjny dla niezbyt długich czasów.

Podsumowując, drugie rozwiązanie określa krok czasowy przy którym rozwiązanie (13.40) jest stabilne numerycznie

$$\tau \ll 2/\lambda_2\,. \tag{13.41}$$

Ceną jest zwyklę mała wartość τ, czyli duża liczba kroków $n = t/\tau$ by osiągnac końcowe t.

Przyklad
Jeżeli

$$\mathbf{y}(t) = \mathrm{e}^{-t}\mathbf{y}_1 + \mathrm{e}^{-1000\,t}\mathbf{y}_2\,, \tag{13.42}$$

to krok $\tau \ll 2\cdot 10^{-3}$, a całkowita liczba kroków $n \gg 10^3$ dla $t \sim 1$.

Wykład 14

Metoda Rungego-Kutty

Zastosujmy rozwinięcie Taylora do rozwiązania równania (13.1):

$$y(t_{n+1}) = y(t_n+\tau) = y(t_n) + \tau\,\frac{dy(t_n)}{dt} + \frac{\tau^2}{2}\frac{d^2y(t_n)}{dt^2} + \dots . \tag{14.1}$$

Pierwsza pochodna to

$$\frac{dy(t_n)}{dt} = F(t_n, y_n)\,. \tag{14.2}$$

Natomiast druga pochodna to

$$\frac{d^2y}{dt^2} = \frac{dF(t,y)}{dt} = \frac{\partial F}{\partial t} + \frac{\partial F}{\partial y}\,\frac{dy}{dt} = \frac{\partial F}{\partial t} + \frac{\partial F}{\partial y}\,F\,. \tag{14.3}$$

Stąd wzór (14.1) przyjmuje postać

$$y_{n+1} = y_n + \tau\,F(t_n,y_n) + \frac{\tau^2}{2}\{\,\partial_t F + F\,\partial_y F\,\}(t_n,y_n) + \mathcal{O}(\tau^3)\,, \tag{14.4}$$

Jest to wzór ścisły i możmy go wykorzystać do znajdowania rozwiązania. Wadą tej metody jest konieczność znajomości pochodnych funkcji F.

14.1 Metoda drugiego rzędu

Metoda Rungego-Kutty pozwala uniknąć problemu pochodnych po prawej stronie wzoru (14.4) poprzez wzięcie wartości funkcji F po prawej stronie w odpowiednio dobranych punktach pomiędzy t_n i t_{n+1}.

W metodzie drugiego rzędu wybiera się dwa takie punkty:

$$\begin{aligned} k_1 &= \tau\,F(t_n,y_n) \\ k_2 &= \tau\,F(t_n + a\,\tau,\, y_n + a\,k_1) \\ y_{n+1} &= y_n + \alpha_1\,k_1 + \alpha_2\,k_2\,, \end{aligned} \tag{14.5}$$

gdzie współczynniki α_1, α_2 i a są tak dobrane by spełnić równanie (14.4) po rozwinięciu (14.5) w szereg w zmiennej τ. Rozwijając k_2 z dokładnością $\mathcal{O}(\tau^3)$, otrzymujemy:

$$\begin{aligned} k_2 &= \tau\{\, F + a\tau\, \partial_t F + a\, k_1\, \partial_y F\,\} \\ &= \tau\{\, F + a\tau\, \partial_t F + a\tau\, F\, \partial_y F\,\} \\ &= \tau F + a\tau^2\,(\,\partial_t F + F\,\partial_y F\,)\,, \end{aligned} \tag{14.6}$$

gdzie wszystkie funkcje po prawej stronie są wzięte w punkcie (t_n, y_n). Podstawiając k_1 i k_2 do wzoru (14.5), otrzymujemy

$$\begin{aligned} y_{n+1} &= y_n + \alpha_1\tau\, F + \alpha_2\Big\{\tau F + a\tau^2\,(\,\partial_t F + F\,\partial_y F\,)\Big\} \\ &= y_n + (\alpha_1+\alpha_2)\,\tau F + \alpha_2\, a\,\tau^2\,(\,\partial_t F + F\,\partial_y F\,)\,. \end{aligned} \tag{14.7}$$

Zgodność z równaniem (14.4) wymaga spełnienia warunków

$$\alpha_1 + \alpha_2 = 1 \qquad\qquad \alpha_2\, a = \tfrac{1}{2}\,. \tag{14.8}$$

Jednym z możliwych wyborów jest

$$\alpha_1 = \alpha_2 = \tfrac{1}{2} \qquad\qquad a = 1\,. \tag{14.9}$$

Otrzymujemy w ten sposób wzory dla metody Rungego-Kutty *drugiego rzędu*:

$$k_1 = \tau\, F(t_n, y_n) \tag{14.10}$$

$$k_2 = \tau\, F(t_n + \tau, y_n + k_1) \tag{14.11}$$

$$y_{n+1} = y_n + \tfrac{1}{2}\,(k_1 + k_2) \tag{14.12}$$

Jak łatwo zauważyć wzory te są identyczne ze wzorami w metodzie przewidź i popraw, gdyż dla argumentów funkcji f we wzorze (14.11), zachodzi

$$\tau_n + \tau = \tau_{n+1}\,, \qquad\qquad y_n + k_1 = y_n + \tau\, F(t_n, y_n) = \bar{y}_{n+1}$$

i stąd otrzymujemy wzór (13.29)

$$y_{n+1} = y_n + \frac{\tau}{2}\,\{F(t_n, y_n) + F(t_{n+1}, \bar{y}_{n+1})\}$$

Dla układu równań różniczkowych (13.11) otrzymujemy wzory analogiczne do powyższych, zapisane w formie wektorowej

$$\begin{aligned} \mathbf{k}_1 &= \tau\, \mathbf{F}(t_n, \mathbf{y}_n) \\ \mathbf{k}_2 &= \tau\, \mathbf{F}(t_n + \tau, \mathbf{y}_n + \mathbf{k}_1) \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + \tfrac{1}{2}\,(\mathbf{k}_1 + \mathbf{k}_2) \end{aligned} \tag{14.13}$$

14.2 Metoda czwartego rzędu

W metodzie *czwartego rzędu* otrzymuje się następujące wzory dla układu równań różniczkowych:

$$\begin{aligned}
\mathbf{k}_1 &= \tau\, \mathbf{F}(t_n, \mathbf{y}_n) \\
\mathbf{k}_2 &= \tau\, \mathbf{F}(t_n + \tfrac{1}{2}\tau, \mathbf{y}_n + \tfrac{1}{2}\mathbf{k}_1) \\
\mathbf{k}_3 &= \tau\, \mathbf{F}(t_n + \tfrac{1}{2}\tau, \mathbf{y}_n + \tfrac{1}{2}\mathbf{k}_2) \\
\mathbf{k}_4 &= \tau\, \mathbf{F}(t_n + \tau, \mathbf{y}_n + \mathbf{k}_3) \\
\mathbf{y}_{n+1} &= \mathbf{y_n} + \tfrac{1}{6}\left(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right). \qquad (14.14)
\end{aligned}$$

Relacja (14.14) jest zgodna z rozwięciem Taylora (14.1) rozwiązania równania różniczkowego aż do wyrazów rzędu τ^4.

Wykład 15

Metody Monte Carlo

Podstawowym elementem metod Monte Carlo są **liczby losowe**, które są generowane z pewnym prawdopodobieństwem. Metody te stosuje się w takich zagadnieniach jak

- symulacja procesów o charakterze stochastycznym, w których wynik pojawia się z określonym prawdopodobieństwem,
- obliczanie całek. Zaletą metod Monte Carlo jest możliwość obliczania wielowymiarowych całek po skomplikowanych obszarach w wielowymiarowej przestrzeni. Na ogół inne metody całkowania zawodzą w takich przypadkach.
- rozwiązywanie równań różniczkowych.

W ogólności, bardziej ogólnym pojęciem jest zmienna losowa. Wartościami tej zmiennej są właśnie liczby losowe.

15.1 Zmienna losowa i rozkład prawdopodobieństwa

Zmienna losowa X przyjmuje **wartości liczbowe** z pewnym **prawdopodobieństwem**.

Innymi słowy, zmienna losowa X jest określona, gdy podamy zakres wartości liczbowych x, które może przyjmować oraz dodatnią funkcję $P(X = x) \leqslant 1$, przy pomocy której określamy **prawdopodbieństwo, że zmienna losowa X przyjmuje wartość liczbową x.**

Przyklad
Zmienną losową jest wynik rzutu kostką. Przyjmuje ona sześć wartości:

$$x \in \{1,2,3,4,5,6\}\,. \tag{15.1}$$

Zakladając, że kostka jest idealna określamy prawdopodobieństwo, iż zmienna losowa przyjmie dozwolone wartości jako

$$P(X = x) = \frac{1}{6}\,. \tag{15.2}$$

Przykład ten jest ilustracją zmiennej losowej o wartościach dyskretnych.

W przypadku gdy zmienna losowa przyjmuje wartości ciągłe $x \in \Re$, prawdopodbieństwo, że wartości zmiennej losowej X są z przedziału $[a, b]$ jest określone przez

$$P(a \leqslant X \leqslant b) = \int_a^b p(x)\,dx \tag{15.3}$$

Dodatnia funkcja $p(x)$ jest nazywana **gęstością prawdopodobieństwa** zmiennej losowej X. Jest ona unormowana do jedynki

$$\int_{-\infty}^{\infty} p(x)\,dx = 1\,. \tag{15.4}$$

Rozkłady zmiennej losowej są charakteryzowane przez takie wielkości jak,

- **wartość średnia** zmiennej losowej

$$\langle X \rangle = \int_{-\infty}^{\infty} x\,p(x)\,dx\,. \tag{15.5}$$

- **wariancja zmiennnej** losowej

$$\sigma_X^2 = \int_{-\infty}^{\infty} (x - \langle X \rangle)^2\,p(x)\,dx = \left\langle X^2 \right\rangle - \langle X \rangle^2\,. \tag{15.6}$$

  Wariancja jest miarą rozrzutu wartości zmiennej losowej wokół jej wartości średniej

- w ogólności definiujemy **momenty** zmiennej losowej

$$\langle X^n \rangle = \int_{-\infty}^{\infty} x^n\,p(x)\,dx\,, \qquad\qquad n = 1, 2, \ldots \tag{15.7}$$

  Gdy powyższe całki nie są zbieżne momenty nie istnieją.

15.2 Przykłady rozkładów prawdopodobieństwa

Przykładami rozkładów zmiennych losowych o ciągłych wartościach są.

- Rozkład **jednostajny**:

$$p(x) = \begin{cases} 1 & x \in [0,1] \\ 0 & \text{poza}\,. \end{cases} \tag{15.8}$$

  Wartość średnia $\langle X \rangle = 1/2$, natomiast wariancja $\sigma_X^2 = 1/12$.

- Rozkład **normalny** Gaussa z parametrami μ oraz σ^2:

$$p(x) = \frac{1}{\sqrt{2\pi\,\sigma^2}}\,\mathrm{e}^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{15.9}$$

  Wartość średnia $\langle X \rangle = \mu$, natomiast wariancja $\sigma_X^2 = \sigma^2$.

Przykładami rozkładów zmiennych losowych o wartościach dyskretnych są.

- Rozkład **dwumianowy**.
  Rozważmy N niezależnych prób (np. rzut monetą). Pytamy jakie jest prawdopodobieństwo odniesienia n sukcesów, jeśli prawdopodbieństwo pojedynczego sukcesu wynosi q. Zmienna losowa X opisuje liczbę sukcesów przyjmując wartości $n = 0, 1, \ldots N$. Jej rozkład prawdopodobieństwa to

$$P(X = n) = \binom{N}{n} q^n\,(1-q)^{N-n}\,. \tag{15.10}$$

  Wartość średnia $\langle X \rangle = Nq$, natomiast dyspersja $\sigma_X^2 = Nq(1-q)$.

- Rozkład **Poissona**.
  Zmienna losowa X przyjmuje wartości $n = 0, 1, 2, \ldots$. Jej rozkład prawdopodobieństwa jest dany przez

$$P(X = n) = \frac{\mu^n}{n!}\,\mathrm{e}^{-\mu}\,, \qquad \mu > 0\,. \tag{15.11}$$

  Wartość średnia $\langle X \rangle = \mu$, a dyspersja $\sigma_X^2 = \mu$. Rozkład Poissona można otrzymać z rozkładu dwumianowego w granicy: $N \to \infty, q \to 0$ takiej, że $Nq = \mu = \text{const}$. Opisuje on więc liczbę sukcesów w bardzo dużej próbie, gdy prawdopodobieństwo pojedynczego sukcesu jest małe.

15.3 Generowanie liczb losowych

Generowanie liczb losowych o zadanym rozkładzie prawdopodbieństwa to centralny element metod Monte Carlo. Wykorzystuje się do tego celu *generatory liczb losowych o rozkładzie jednostajnym* na odcinku $[0,1]$, dla których gęstość prawdopodobieństwa jest zadana przez wzór

$$p(x) = \begin{cases} 1 & x \in [0,1] \\ 0 & \text{poza } [0,1]\,. \end{cases} \tag{15.12}$$

Generatory takie są oferowane w ramach bibliotek programów komputerowych, na przykład CERNLIB. W praktyce generowane liczby nie są w pełni losowe. Na przykład, powtarzają się w ramach dość długiego cyklu. Mogą także istnieć korelacje między nimi.

Tym niemniej dobry generator powinien spełniać trzy podstawowe kryteria.

- Mieć bardzo długi okres powtarzalności.
  Na przykład dla komputera 32-bitowego powinien mieć okres bliski

  $$2^{31} - 1 = 2\ 147\ 483\ 647\,,$$

  gdyż zakres liczb całkowitych dla tego komputera to $[-2^{31}, 2^{31} - 1]$
- Charakteryzować się dobrą *losowością*. Oznacza to, że korelacje pomiędzy wszystkimi, kolejno generowanymi liczbami powinny być możliwie małe.
- Być szybki.

Przykładem dobrego generatora jest generator, w którym kolejne liczby losowe otrzymuje się z relacji

$$x_{n+1} = (ax_n + b) \mod c\,. \tag{15.13}$$

Liczby a, b, c nazywa się magicznymi, od ich wyboru zależy jakość generatora. Jednym z dobrych zestawów tych liczb jest $a = 7^5 = 16\ 807,\ b = 0,\ c = 2^{31} - 1$.

Przyklad
Przyjmijmy $a = 3,\ b = 0,\ c = 2^3 - 1 = 7$. Zaczynając od 1 otrzymujemy kolejne liczby, generowane zgodnie z regułą $x_{n+1} = 3x_n \mod 7$:

$$1\ \ 3\ \ 2\ \ 6\ \ 4\ \ 5\ \ 1\ \ \ldots$$

Dzieląc je przez 7 znajdujemy sześć liczb z przedziału $[0,1]$.

Dysponując liczbami o rozkładzie jednostajnym możemy otrzymać liczby losowe o dowolnych innych rozkładach, np. gausowskim.

15.4 Całkowanie metodą Monte Carlo

Podstawowy wzór na oblicznie całki w metodzie Monte Carlo to

$$\int_a^b f(x)\,dx \approx V\langle f\rangle \pm V\sqrt{\frac{\langle f^2\rangle - \langle f\rangle^2}{N}}, \tag{15.14}$$

gdzie $V = b-a$ oraz

$$\langle f\rangle = \frac{1}{N}\sum_{i=1}^{N} f(x_i), \qquad \left\langle f^2\right\rangle = \frac{1}{N}\sum_{i=1}^{N} f^2(x_i). \tag{15.15}$$

W powyższych wzorach x_i są liczbami losowymi o *rozkładzie jednostajnym* na odcinku $[a,b]$, natomiast N jest liczbą wygenerowanych liczb losowych. Stała V wynika z konieczności normalizacji rozkładu jednostajnego do jedynki, gdyż przymując $f \equiv 1$ powinniśmy otrzymać

$$\int_a^b dx = b-a. \tag{15.16}$$

Drugie wyrażenie po prawej stronie (15.14) jest estymacją błędu wyniku

Wzór (15.14) jest słuszny także dla całkowania w n-wymiarowej przestrzeni z punktami $\mathbf{x} = (x_1, x_2, \ldots, x_n)$,

$$\int_W f(\mathbf{x})\,d\mathbf{x} \approx V\langle f\rangle \pm V\sqrt{\frac{\langle f^2\rangle - \langle f\rangle^2}{N}}. \tag{15.17}$$

We wzorze tym V jest n-wymiarową objętością przestrzeni W, po której całkujemy

$$V = \int_W dx_1\,dx_2\ldots dx_n, \tag{15.18}$$

natomiast

$$\langle f\rangle = \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_i), \qquad \left\langle f^2\right\rangle = \frac{1}{N}\sum_{i=1}^{N} f^2(\mathbf{x}_i), \tag{15.19}$$

gdzie tym razem $\mathbf{x}_i = (x_{1i}, x_{2i}, \ldots, x_{ni})$ to wielowymiarowe liczby losowe o rozkładzie jednostajnym w obszarze całkowania.

W praktyce obszar W może być na tyle skomplikowany, że nie jest łatwo próbkować go przy pomocy rozkładu jednostajnego. Wybieramy wtedy obszar U zawierający W,

$$W \subset U \tag{15.20}$$

w którym łatwo wygenerować liczby losowe o rozkładzie jednostajnym. Dodatkowo definiujemy nową funkcję $\tilde{f}$ równą funkcji f w obszarze W i zero poza nim. Wtedy

$$\int_U \tilde{f}(\mathbf{x})\,d\mathbf{x} = \int_W f(\mathbf{x})\,d\mathbf{x}. \tag{15.21}$$

Wybór obszaru U wpływa na wielkość błędu w metodzie Monte Carlo. Błąd jest tym większy im większa jest różnica między obszarami U i W. Dobrze jest więc wybrać U tak, by liczba wylosowanych punktów poza obszarem W była jak najmniejsza.

Przyklad
Opisaną metodą możemy policzyć całkę $\int_a^b f(x)\,dx$ z dodatniej funkcji f, definiując funkcję dwóch zmiennych

$$F(x,y) = \begin{cases} 1 & x \leqslant y = f(x) \\ 0 & x > y \end{cases} \tag{15.22}$$

określoną w kwadracie $U = [a,b] \times [0, \max(f)]$. Wtedy

$$\int_a^b f(x)\,dx = \int_U F(x,y)\,dx\,dy \approx V\,\frac{1}{N} \sum_{x_i \leqslant y_i} 1, \tag{15.23}$$

gdzie V jest polem kwadratu U, po którym całkujemy. Wartość całki jest więc proporcjonalna do stosunku liczby punktów leżacych pod wykresem funkcji $y = f(x)$ do całkowitej liczby N wygenerowanych punktów o rozkładzie jednostajnym w obszarze całkowania.

Literatura

[1] Z. Fortuna, B. Macukow, J. Wąsowski; *Metody numeryczne*; WNT, Warszawa, 2005.

[2] W. H. Press, S. A. Teukolski, W. T. Vetterling, B. P. Flannery; *Numerical Recipes*; Cambridge University Press, 1992. http://www.nr.com/nronline_switcher.php

[3] Tao Pang; *Metody obliczeniowe w fizyce*, PWN, Warszawa, 2001.