Řešení písemné práce – vzor

Příklad 1. Úloha řešení soustavy lineárních algebraických rovnic $\boldsymbol{Ax} = \boldsymbol{b}$.

a. Formulace dané úlohy.

Pro zadanou matici soustavy $\boldsymbol{A}$ a vektor pravé strany $\boldsymbol{b}$ určete vektor $\boldsymbol{x}$ takový, že platí $\boldsymbol{Ax} = \boldsymbol{b}$.

b. Obecný princip iteračních metod.

Konstruujeme posloupnost vektorů $\boldsymbol{x}_i$, $i = 0, 1, 2, \dots$ takovou, že její limitou bude přesné řešení $\boldsymbol{x}^*$ zadané úlohy. Počáteční aproximaci $\boldsymbol{x}_0$ volíme. Obecně lze iterační proces zapsat ve tvaru:

$$\boldsymbol{x}_{i+1} = \boldsymbol{F}_i(\boldsymbol{x}_i, \boldsymbol{x}_{i-1}, \dots, \boldsymbol{x}_0), \qquad i = 0, 1, \dots.$$

Jelikož by bylo náročné počítat $(i+1)$-tou iteraci vždy pomocí všech předchozích iterací, budeme ji počítat pouze pomocí poslední, tj. $\boldsymbol{x}_i$. A dále můžeme předpokládat, že zobrazení $\boldsymbol{F}_i$ je lineární, tj. dostáváme předpis:

$$\boldsymbol{x}_{i+1} = \boldsymbol{M}_i \boldsymbol{x}_i + \boldsymbol{c}_i, \qquad i = 0, 1, \dots.$$

Je vhodné vyjádřit závislost iteračního procesu na vektoru pravé strany $\boldsymbol{b}$:

$$\boldsymbol{x}_{i+1} = \boldsymbol{M}_i \boldsymbol{x}_i + \boldsymbol{N}_i \boldsymbol{b}, \qquad i = 0, 1, \dots.$$

Poslední vzorec představuje jednokrokovou maticovou iterační metodu.

c. Algoritmus metody SOR.

Metoda SOR je příkladem iterační metody. Nejprve je nutné zvolit počáteční aproximaci $\boldsymbol{x}_0$ a hodnotu parametru ω. Novou iteraci $\boldsymbol{x}_{i+1}$ hledáme jako lineární kombinaci předchozí hodnoty $\boldsymbol{x}_i$ a hodnoty získané Gaussovou-Seidelovou metodou $\boldsymbol{x}_{i+1}^{GS}$. Je zřejmé, že pro parametr $\omega = 1$ dostaneme Gaussovu-Seidelovu metodu.

d. Realizace.

$$\boldsymbol{A} = \begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} 3 \\ 9 \end{bmatrix}, \qquad \boldsymbol{x}_0 = \begin{bmatrix} 1,5 \\ 1,5 \end{bmatrix}.$$

Předpis pro Gaussovu-Seidelovu metodu:

$$x_{i+1} = \frac{3 - y_i}{2}, \qquad y_{i+1} = \frac{9 - x_{i+1}}{4}.$$

Předpis pro metodu SOR:

$$x_{i+1} = \omega\,\frac{3 - y_i}{2} + (1-\omega)x_i, \qquad y_{i+1} = \omega\,\frac{9 - x_{i+1}}{4} + (1-\omega)y_i.$$

1) $\omega = \omega_1 = 1,2$.

i	0	1	2	3	4
x_i	1,50000	0,60000	0.34800	0,43944	0,42940
y_i	1,50000	2,22000	2,15160	2.13784	2,14360

2) $\omega = \omega_2 = 1,6$.

i	0	1	2	3	4
x_i	1,50000	0,30000	0,15600	0,71472	0,27488
y_i	1,50000	2,58000	1,98960	2,12035	2,21784

e. Úloha parametru ω.

Smysl parametru ω je v možnosti urychlit výpočet. Samozřejmě záleží na volbě. Parametr ω musí být z intervalu $(0, 2)$. V bodě d. je vidět, že metoda SOR pro $\omega = \omega_1$ konverguje rychleji než pro $\omega = \omega_2$. Existuje postup na určení optimální hodnoty parametru ω.

Příklad 2. Numerické integrování.

a. Formulace úlohy numerického integrování.

Pro danou integrovatelnou funkci $f = f(x)$ na intervalu $\langle a, b\rangle$ chceme určit hodnotu určitého integrálu

$$I(f) = \int_a^b f(x)\,dx.$$

b. Vysvětlete princip Newtonových–Cotesových vzorců.

Základní tvar Newtonových–Cotesových vzorců získáme tak, že na intervalu integrace nahradíme integrovanou funkci interpolačním polynomem s ekvidistantními uzly. Složené Newtonovy–Cotesovy vzorce získáme tak, že provedeme nejprve ekvidistantní dělení celého intervalu $\langle a, b\rangle$ a na vzniklých podintervalech použijeme základní vzorce.

Nejjednodušší vzorec dostaneme aproximujeme-li funkci f konstantní funkcí. Obdržíme tzv. *obdélníkové pravidlo*. V případě, že funkci f aproximujeme lineární funkcí, obdržíme *lichoběžníkové pravidlo*. A v případě, že funkci f aproximujeme kvadratickou funkcí, obdržíme *Simpsonovo pravidlo*. (Načrtnout obrázky !)

c. Popište základní a složené lichoběžníkové pravidlo.

Základní vzorec odvodíme tak, že na intervalu $\langle x_k, x_{k+1}\rangle$ nahradíme funkci f lineárním interpolačním polynomem, který interpoluje f v uzlech x_k a x_{k+1}. Dostaneme

$$\int_{x_k}^{x_{k+1}} f(x)\,dx \approx \frac{h}{2}[f(x_k) + f(x_{k+1})], \text{ kde } h = x_{k+1} - x_k \text{ je krok dělení.}$$

Složený vzorec dostaneme sečtením těchto základních vzorců, tj.

$$\int_a^b f(x)\,dx \approx h[\frac{1}{2}f(x_0) + \sum_{k=1}^{N-1} f(x_k) + \frac{1}{2}f(x_N)], \text{ kde } N = \frac{b-a}{h} \text{ je počet uzlů dělení.}$$

Integrovanou funkci f jsme nahradili lomenou čarou. (Obrázek !)

d. Výpočet přibližné hodnoty určitého integrálu (krok $h = 0,2$)

$$\int_1^2 x^2\ dx \approx 0,2(\frac{1}{2}.1^2 + 1,2^2 + 1,4^2 + 1,6^2 + 1,8^2 + \frac{1}{2}.2^2) =$$

$$= 0,2(0,5 + 1,44 + 1,96 + 2,56 + 3,24 + 2) = 0,2.11,7 = 2,34.$$

e. Definice.

Algebraickým řádem kvadraturního vzorce rozumíme takové celé číslo n, pro něž platí, že kvadraturní vzorec integruje přesně polynomy až do stupně n včetně a polynom stupně $n+1$ již přesně neintegruje.

Konvergentní kvadraturou nazveme takové systémy kvadraturních vzorců s koeficienty w_i^n, $i = 0,1,\ldots,n$ a uzly x_i^n, $i = 0,1,\ldots,n$ definovanými pro všechna $n = 0,1,\ldots$, pro které platí

$$\lim_{n\to\infty} \sum_{i=0}^{n} w_i^n f(x_i^n) = \int_a^b f(x)\,dx$$

pro každou spojitou funkci f.

(S rostoucím n musíme dostat přesnější hodnotu daného integrálu.)

Příklad 3. Počáteční úloha pro obyčejnou diferenciální rovnici.

a. Formulace dané úlohy.

Pro danou reálnou funkci $f = f(x, y(x))$ určete řešení $y = y(x)$ rovnice

$$y'(x) = f(x, y(x))$$

s počáteční podmínkou

$$y(x_0) = y_0.$$

b. Eulerova metoda.

Jedná se o nejjednodušší jednokrokovou metodu. Základem metody je diskretizace intervalu pro proměnnou a aproximace řešení pomocí Taylorova rozvoje. Přibližné řešení se nekonstruuje jako spojitá funkce, ale generujeme body $x_0, x_1, x_2, \ldots$ a určujeme hodnoty $y_0, y_1, y_2, \ldots$, která aproximují $y(x_0), y(x_1), y(x_2), \ldots$.

Princip:

y_0 je dáno (počáteční podmínka)

y_1 počítáme extrapolací z hodnoty y_0, přičemž se na intervalu $\langle x_0, x_1\rangle$ řešení aproximuje přímkou, která prochází bodem (x_0, y_0) a má směrnici $y' = f(x_0, y_0)$. Ta má rovnici

$$y = y_0 + (x - x_0)f(x_0, y_0),$$

tj. pro x_1 dostáváme

$$y_1 = y_0 + (x_1 - x_0)f(x_0, y_0).$$

Tento postup opakujeme pro určení hodnoty y_2. Obecně potom dostáváme rekurentní vztah:

$$y_{n+1} = y_n + (x_{n+1} - x_n)f(x_n, y_n), \quad n = 0,1,\ldots.$$

(Obrázek !)

c. Řešení úlohy (Pro ekvidistantní dělení intervalu $\langle 0,2\rangle$ s diskretizačním krokem $h = 0,5$)

$$\begin{aligned} y' &= x - y, \ x \in (0,2), \\ y(0) &= 1. \end{aligned}$$

n	0	1	2	3	4
x_n	0	0,5	1	1,5	2
y_n	1	0,5	0,5	0,75	1,125

d. Podmínky řešitelnosti úlohy.

Postačující podmínkou existence a jednoznačnosti řešení počáteční úlohy je splnění spojitosti a Lipschitzovy podmínky v proměnné y funkce f na množině $\langle a,b\rangle \times \mathbb{R}$. Řekneme, že f splňuje na množině $\langle a,b\rangle \times \mathbb{R}$ Lipschitzovu podmínku v proměnné y, existuje-li konstanta L (nezávislá na x, y) tak, že platí

$$\|f(x,y) - f(x,z)\| \leq L\,\|y - z\|, \quad \forall x \in \langle a,b\rangle, \quad \forall y,z \in \mathbb{R}.$$

e. Definice.

Během numerického řešení počátečních úloh vznikají chyby dvojího druhu, chyby aproximace a chyby zaokrouhlovací. Chyby aproximace (metody) dělíme na lokální a globální chybu metody.

Lokální chyba metody d_n na intervalu $\langle x_n, x_{n+1}\rangle$ je nepřesnost, s níž hodnoty teoretického řešení dané úlohy splňují rekurentní vztah, ze kterého se počítá hodnota y_{n+1} přibližného řešení. Pro d_n platí:

$$y(x_{n+1}) = y(x_n) + \underbrace{(x_{n+1} - x_n)}_{h_n} f(x_n, y(x_n)) + d_n,$$

kde $y(x)$ jsou hodnoty přesného řešení.

Jinak lze lokální diskretizační chybu chápat jako nepřesnost, které se dopouštíme v jednom kroku metody.

Globální chyba metody je definována takto:

$$e_n = y(x_n) - y_n,$$

a je samozřejmě závislá na lokálních diskretizačních chybách.

Řád metody je největší přirozené číslo p takové, že pro danou metodu aplikovanou na počáteční úlohu s dostatečně hladkým řešením platí při každém pevném n a $h_n \to 0$ odhad

$$d_n = O(h_n^{p+1}),$$

tj. existuje číslo C nezávislé na h_n takové, že pro malá h_n platí $|d_n| \leq C h_n^{p+1}$.