Essential "Mathematische Anwendungen"

Numerische Mathematik in aller Kürze

Inhaltsverzeichnis

1. Zahlendarstellung

Maschinenzahlen: $x = \pm y \cdot b^2$

$y = \sum_{i=m}^{n} y_i \cdot b^{-1}$ = „Mantisse“

b = „Basis“

$z = \sum_{i=0}^{t} z_i \cdot b^i$ = „Exponent“

2. Nichtlineare Gleichungen

2.1. Nullstellensuche

f(x) = 0
1. Bisektionsverfahren
2. Sekantenverfahren (q = 1,618) = "Mehrschrittverfahren"
3. Fixpunktverfahren = "Einzelschrittverfahren"

a) Fixpunktgleichung F(x)

nach Newton: $F(x) = x - \frac{f(x)}{f'(x)}$ (q = 2)

nach Steffensen: $F(x) = x - \frac{f(x)^2}{f\left(x + f(x)\right) - f(x)}$ (q = 1)

höhere Verfahren:

q = 3: $F(x) = x - \frac{f(x)}{f'(x)} - \frac{f(x)^2 \cdot f''(x)}{2 \cdot f'(x)^3}$

q = 4: $F(x) = F(q = 3) + f(x)^3 \cdot \left(\frac{f'''(x)}{6 \cdot f'(x)^4} - \frac{f''(x)^2}{2 \cdot f'(x)^5} \right)$

=> Untersuche F(x) ob F: A -> A für [a, b]

b) Untersuche $F'(x)$: wenn $\max\left|F'(x)\right| := k < 1$
=> F ist Kontraktion auf [a, b]

c) Wähle Punkt x_0 aus [a, b]
setze in F(x) rekursiv ein: $x_{i+1} = F(x_i)$

$x_0 = a$

$x_1 = F(x_0) = \ldots$

$x_2 = F(x_1)$

$x_n = F(x_{n-1})$

d) Untersuche Anzahl der Iterationen (= n)
Geg: $\varepsilon = |x_n - s|$
mit Hilfe folgender Abschätzungsformel:

$$\underbrace{|x_n - s|}_{\varepsilon} \le \frac{1}{1-\underbrace{k}_{geg}} \cdot |x_{n+1} - x_n| \le \frac{k^n}{1-k}(x_1 - x_0)$$

2.2. Polynome

$$p(x) = a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + \cdots + a_1 \cdot x + a_0$$
$$Grad(p(x)) = n$$

2.2.1. Darstellung nach Horner

$$p(x) = (((a_n \cdot x + a_{n-1}) \cdot x + a_{n-2}) \cdot x + \cdots + a1) \cdot x + a_0$$

$\Rightarrow$ besserer Algorithmus

2.2.2. Horner-Schema Beispiel:

(numerisches Verfahren für Polynomdivision)
p(3), p'(3),... von $4x^4 - 2x^2 + 9x - 4$

1. Schritt:

	4	0	-2	9	-4	
3		12	36	102	333	
	4	12	34	111	**329**	= p(3)

2. Schritt:

	4 (*1)	12 (*1)	34 (*1)	111 (*1)	
3		12	72	318	
	4	24	106	**429**	= p'(3)

3. Schritt:

	4 (*2)	24 (*2)	106 (*2)	
3		24	216	
	8	72	**428**	= p''(3)

4. Schritt:

	8 (*3)	72 (*3)	
3		72	
	24	**288**	= p'''(3)

nach 1.Schritt: $p(x) = (x-3) \cdot q(x) + 329$
mit: $q(x) = 4x^3 + 12x^2 + 34x + 111$

2.2.3. Abschätzung der Lage von Nullstellen:

Gebiet in ***C*** (Radius), in welchem Nullstellen liegen

Für $\xi \in$ ***C*** gilt:: (egal ob Polynom mit oder ohne komplexen Koeffizienten)

1. $|\xi| \le \max\left\{\left|\frac{a_0}{a_n}\right|,\quad 1+\left|\frac{a_1}{a_n}\right|,\quad \cdots,\quad 1+\left|\frac{a_{n-1}}{a_n}\right|\right\}$
2. $|\xi| \le \max\left\{\left|\frac{a_0}{a_1}\right|,\quad 2\cdot\left|\frac{a_1}{a_2}\right|,\quad \cdots,\quad 2\cdot\left|\frac{a_{n-1}}{a_n}\right|\right\}$
3. $|\xi| = 2\cdot\max\left\{\sqrt[n]{\left|\frac{a_0}{a_n}\right|},\quad \sqrt[n-1]{\left|\frac{a_1}{a_n}\right|},\quad \cdots,\quad \sqrt{\left|\frac{a_{n-2}}{a_n}\right|},\quad \left|\frac{a_{n-1}}{a_n}\right|\right\}$

(Es gilt der kleinste Wert aus 1., 2. und 3.)

$\Rightarrow$ liefert Startwert x_0 für Iteration

2.2.4. Suche von Nullstellen:

α_1 größte reelle Nullstelle und $\alpha_1 \le \max\{\operatorname{Re}(\xi_1), \ldots \operatorname{Re}(\xi_n)\}$ mit ξ_1 - ξ_n komplexe Nullstellen. (Größte Nullstelle muss eine reelle sein!)

$$F(x) = x - \frac{p(x)}{p'(x)}$$
$$\Rightarrow x_{i+1} = x_i - \frac{p(x_i)}{p'(x_i)}$$

Konvergiert für $x_i = x_0$ gegen α_1
(analog für kleinste reelle Nullstelle)

Wenn α_1 gefunden, dann suche nächste Nullstellen mit folgendem Verfahren:
$\alpha_1 > \alpha_2 > \ldots > \alpha_j$

$$F(x) = x - \frac{p(x)}{p'(x) - p(x)\cdot\sum_{k=1}^{j}\frac{1}{x-a_k}}$$

Summe aller schon gefundenen Nullstellen

3. Graphentheorie

Graph G:
V = Ecken (Knoten)
E = Kanten (Edge)
g = (V, E), $E \subset [V]^2$

Digraph:
$E \in V \times V$ ($V \times V$ ist geordnet)

gewichteter Graph:
Tripel G=(V, E, W)
W = Gewicht der Kante

Computerdarstellung:
|V|=n, |E|=m
- Nachbarschaftsmatrix ($n \times m$ Matrix)
- Linked List

4. Gaußscher Algorithmus

4.1. Zeilen-Stufen-Form:

$$B = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ 0 & b_{22} & \cdots & b_{2n} \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & b_{nn} \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$$

4.2. Approximation von Funktionen:

Theorem 1:
f[a,b]→***R*** stetig
⇒ es existiert Polynom p(x) mit |f(x) - p(x)| < ε

Theorem 2: Taylor
f[a, b] → ***R***, n+1 mal stetig diffbar
x0 ∈ [a, b]

$\Rightarrow p(x) = f(x_0) + \frac{f'(x_0)}{1!} \cdot (x - x_0) + \frac{f''(x_0)}{2!} \cdot (x - x_0)^2 + \cdots + \frac{f^n(x_0)}{n!} \cdot (x - x_0)^n = j_{x_0}^n(f)$ = n-Jet (Tylorpolynom)

Fehlerabschätzung: $|f(x) - p(x)| = \left| \frac{x^{2n}}{(2n)!} \cdot f(\xi) \right| \quad mit\ 0 < \xi < x$

Taylorpolynom hat mit f(x) Ableitungen an einem Punkt gemeinsam.

Interpolationspolynom: hat mit f(x) n+1 Punkte gemeinsam.

4.3. Lagrange-Interpolationspolynom

($O(n^2)$)

Geg.: m = 0, 1, 2, ... n+1 Punkte
$f_0, f_1, f_2, \ldots f_n$

a) Berechne für jeden Punkt m Produkt:

$$l_m(x) = \prod_{\substack{k=0 \\ k \neq m}}^{n} \frac{x - x_k}{x_m - x_k}$$

b) $\Rightarrow p(x) = \sum_{m=0}^{n} f_m \cdot l_m(x)$

4.4. Spline- Interpolation

(Kubische Splines k=3)

Geg.: (x_i, f_i), i = 0, ... n Punkte
Teilintervalle [x_i, x_{i+1}]

Breite der Intervalle:
hi = (x_{i+1} - x_i) i = 0, ... n-1

S_i bestimmen:

$S_i = 6 \cdot \frac{f_{i+1} - f_i}{h_i} - 6 \cdot \frac{f_i - f_{i-1}}{h_{i-1}}$ i = 1, ... n-1

Stelle Matrix unter Berücksichtigung der Randbedingung auf:
(n Stützpunkte ⇒ n×n Matrix)

Natürliche Randbedingung:

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & & & 0 \\ h_0 & 2\cdot(h_0+h_1) & h_1 & 0 & & & \\ 0 & h_1 & 2\cdot(h_1+h_2) & h_2 & 0 & & \\ \vdots & & \ddots & \ddots & \ddots & & \vdots \\ & & & & h_{n-2} & 2\cdot(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & & & \cdots & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f_0'' \\ f_1'' \\ \\ \vdots \\ \\ f_n'' \end{pmatrix} = \begin{pmatrix} 0 \\ S_1 \\ S_2 \\ \vdots \\ S_{n-1} \\ 0 \end{pmatrix}$$

1. und letzte Zeile für vollständige Randbedingung:

$$\begin{pmatrix} 2\cdot h_0 & h_0 & 0 & \cdots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & \cdots & 0 & h_{n-1} & 2\cdot h_{n-1} \end{pmatrix} \begin{pmatrix} f_0'' \\ \vdots \\ f_n'' \end{pmatrix} = \begin{pmatrix} 6\cdot\frac{f_1-f_0}{h_0} - 6\cdot f_0 \\ \vdots \\ 6\cdot f_n - 6\cdot\frac{f_n-f_{n-1}}{h_{n-1}} \end{pmatrix}$$

1. und letzte Zeile für periodische Randbedingung:

$$\begin{pmatrix} 2\cdot(h_{n-1}+h_0) & h_0 & 0 & \cdots & 0 & h_{n-1} & 0 \\ \vdots & & & \ddots & & & \vdots \\ 1 & 0 & & \cdots & & 0 & -1 \end{pmatrix} \begin{pmatrix} f_0'' \\ \vdots \\ f_n'' \end{pmatrix} = \begin{pmatrix} 6\cdot\frac{f_1-f_0}{h_0} - 6\cdot\frac{f_0-f_{n-1}}{h_{n-1}} \\ \vdots \\ 0 \end{pmatrix}$$

1. und letzte Zeile für not-a-knot Randbedingung:

$$\begin{pmatrix} h_1 & -(h_0+h_1) & h_0 & 0 & & \cdots & 0 \\ \vdots & & \ddots & & & & \vdots \\ 0 & \cdots & 0 & h_{n-1} & -(h_{n-2}+h_{n-1}) & & h_{n-2} \end{pmatrix} \begin{pmatrix} f_0'' \\ \vdots \\ f_n'' \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$$

jetzt $f''_0, \dots f''_n$ durch lösen des Gleichungs-Systems berechnen

Für jedes Intervall $[x_i, x_{i+1}]$ Polynom bestimmen:

$$S(x) = a_i + b_i \cdot (x - x_i) + c_i \cdot (x - x_i)^2 + d_i \cdot (x - x_i)^3$$

mit:

$$a_i = f_i$$

$$b_i = \frac{f_{i+1} - f_i}{h_i} - \frac{h_i}{6} \cdot \left(f''_{i+1} + 2 \cdot f''_i\right)$$

$$c_i = \frac{f''_i}{2}$$

$$d_i = \frac{f''_{i+1} - f''_i}{6 \cdot h_i}$$

4.5. Spline-Interpolation von Kurven:

Geg.: $f_0 = \begin{pmatrix} f_{01} \\ f_{02} \end{pmatrix}, f_1 = \begin{pmatrix} f_{11} \\ f_{12} \end{pmatrix}, \cdots f_n = \begin{pmatrix} f_{n1} \\ f_{n2} \end{pmatrix}$ Vektoren

Suche Kurve, welche $f_0, \dots f_n$ Interpoliert?

1. $x_0, \dots x_n$ bestimmen
$x_0 = 0$
$x_i = x_{i-1} + ||f_i - f_{i-1}||$ $i = 1, \dots n$ ($||x-y||$ = Abstand)
$\Rightarrow x_0 < x_1 < \dots < x_n$

2. Löse 2 Interpolationsaufgaben:
a) 1. Komponente der Punkte $S_1(x)$ mit:
$(x_0, f_{01}), (x_1, f_{11}), \dots (x_n, f_{n1})$
b) 2. Komponente der Punkte S2(x) mit:
$(x_0, f_{02}), (x_1, f_{12}), \dots (x_n, f_{n2})$

3. dann $S : [x_0, x_n] \rightarrow \boldsymbol{R}^2$

$$x \rightarrow \begin{pmatrix} S_1(x) \\ S_2(x) \end{pmatrix}$$

analog $\boldsymbol{R}^3$

5. Numerische Integration

(Simpson Regel)

Geg.: $\int_a^b f(x)dx$

1. a) Unterteile [a, b] in n (n gerade!) Teilintervalle:

$a = x_0 < x_1 < \cdots < x_n = b$

$h = \frac{b-a}{n} \quad x_i = a + h \cdot i \quad i = 1, \cdots n$

b) Interpoliere jeweils 3 Punkte
(x_0, x_1, x_2) (x_2, x_3, x_4) ... (x_{n-2}, x_{n-1}, x_n) mit $p(x) \le 2$
mit

$\int_{x_{2j-2}}^{x_{2j}} P_j(x)dx = \frac{h}{3}\left(y_{2j-2} + 4 \cdot y_{2j-1} + y_{2j}\right)$ i = 1, ... n-1

2. Insgesamt:

$$\int_a^b f(x)dx = \frac{h}{3}\Big(f(x_0) + 4 \cdot f(x_1) + 2 \cdot f(x_2) + 4 \cdot f(x_3) + 2 \cdot f(x_4) + \cdots + 2 \cdot f(x_{n-2}) + 4 \cdot f(x_{n-1}) + f(x_n) \Big)$$

Fehler: $|R(h)| \le \frac{h^4}{180} \cdot \max_{x \in [a,b]} \left| f^4(x) \right| \cdot (b-a)$

5.1. Newton-Codes

(Newtonsche $^3/_8$ Regel)

1. k+1 Stützstellen
2. lege Polynom vom Grad K durch Stützpunkte
3. Integriere auf Teilintervallen

z.B.: k=3

$$\int_a^b f(x)dx = \frac{3}{8} \cdot h \cdot \Big(f(x_0) + 3 \cdot f(x_1) + 3 \cdot f(x_2) + 2 \cdot f(x_3) + 3 \cdot f(x_4) + 3 \cdot f(x_5) + 2 \cdot f(x_6) + \cdots + f(x_n) \Big)$$

Fehler: $|R| \le \frac{h^4}{80} \cdot \max_{x \in [a,b]} \left| f^4(x) \right| \cdot (b-a)$

6. Differentialgleichungen:

Anfangswertaufgabe:
DGL 1. Ordnung:
$y' = f(x, y)$
$y(x_0) = y_0$

Ges: $y(x)$ mit $y'(x) = f(x, y)$ $y(x_0) = y_0$