9. VL (Einf. in die Num. Mathematik) G. Bärwolff, 13.5.09

Algorithmische Aspekte der Polynominterpolation

a) Für die Berechnung eines Pol. in der Form

$$P(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$$

1 2 ... n Multiplikation

und n Additionen

$\rightsquigarrow \frac{n(n+1)}{2}$ Multipl. $\sim \mathcal{O}(n^2)$ flops

b) Horner-Schema

$$P(x) = a_0 + x(a_1 + x(a_2 + \dots))$$

$$= (\dots((a_n x + a_{n-1})x + a_{n-2})x + \dots + a_1)x + a_0$$

$\rightsquigarrow$ n Multiplikationen und
n Additionen $2n \sim \mathcal{O}(n)$ flops

Horner-Schema für die Newton-Basis

$$P(x) = \sum_{k=0}^{n} c_k N_k(x) \quad , c_k \text{ gegeben}$$

N_k rekursiv aufgebaut

$$N_k(x) = (x - x_0) \cdot \dots \cdot (x - x_{k-1})$$

$$\rightsquigarrow N_k(x) = N_{k-1}(x)(x - x_{k-1})$$

P kann in der Form

$$p(x) = c_0 + (x - x_0)(c_1 + (x - x_1)(c_2 + \dots + c_n(x - x_{n-1}) \dots))$$

Daraus resultiert der Algorithmus

- $u_{n+1} = 0$
  for $k = n(-1)0$ ← 1 Multipl. und 2 Additionen →
  $u_k = (x - x_k)\, u_{k+1} + c_k$
  end for
  $P(x) = u_0$

3n flops

c) Lagrange-Interpolation

Im Unterschied zur Newton-Interpolation ist der Aufwand bei der Lagrange-Interpol. bei Hinzunahme einer Stützstelle recht groß, denn sämtliche Basispolynome ändern sich (Grad wird um 1 erhöht)

Was kann man tun, um hier den Mehraufwand zur Berechnung von $P(x)$ an einer Stelle $x \neq x_j$ klein zu halten?

Man findet

$$P(x) = \sum_{k=0}^{n} y_k L_k(x) = \sum_{k=0}^{n} y_k \prod_{\substack{j=0 \\ j\neq k}}^{n} \frac{(x - x_k)}{(x_j - x_k)} =$$

$$= \sum_{k=0}^{n} y_k \frac{1}{x - x_k} \left[\prod_{\substack{i=0 \\ i\neq k}}^{n} \frac{1}{x_i - x_k} \right] \prod_{j=0}^{n} (x - x_j)$$

Die Koeffizienten in den eckigen Klammern

$$\lambda_k = \prod_{\substack{i=0 \\ i\neq k}}^{n} \frac{1}{x_i - x_k} = \frac{1}{\prod (x_i - x_k)} \qquad k = 0, 1, \ldots, n$$

nennt man Stützkoeffizienten. Damit führt man mit

$$\mu_k = \frac{\lambda_k}{x - x_k} \qquad k = 0, 1, \dots, n$$

Größen ein, die von der Stelle x, an der interpoliert werden soll, abhängen. Es ergibt sich

$$p(x) = \left[\sum_{k=0}^{n} \mu_k y_k \right] \prod_{j=0}^{n} (x - x_j) \qquad (3.9)$$

Betrachtet man (3.9) für die speziellen Werte $y_k = 1$, $k = 0, \dots, n$, dann ist $p(x) \equiv 1$ das eindeutig bestimmte Interpolationspolynom für die $n+1$ Stützpunkte $(x_k, 1)$, so dass

$$1 = p(x) = \left[\sum_{k=0}^{n} \mu_k \right] \prod_{j=0}^{n} (x - x_k) \Longleftrightarrow$$

$$\prod_{j=0}^{n} (x - x_k) = \frac{1}{\sum_{k=0}^{n} \mu_k} \qquad (3.10)$$

Aus (3.9) und (3.10) folgt mit

$$p(x) = \frac{\sum_{k=0}^{n} \mu_k y_k}{\sum_{k=0}^{n} \mu_k} \qquad (3.11)$$

die sogenannte baryzentrische Formel der Lagrange-Interpolation.

Satz 3.11

Für die $n+1$ Stützkoeff. $\lambda_k^{(n)}$ zu den paarweise verschiedenen Stützstellen $x_0, x_1, \dots, x_n$ gilt

$$\sum_{k=0}^{n} \lambda_k^{(n)} = 0 \qquad (3.12)$$

Beweis als Übung

Die Formel (3.11) hat den Vorteil, dass man bei der Hinzunahme einer $(n+2)$-ten Stützstelle x_{n+1} zu $x_0, x_1, \dots, x_n$ die neuen λ-Werte $\lambda_k^{(n+1)}$ aus den alten $\lambda_k^{(n)}$ durch die Beziehungen

$$\lambda_k^{(n+1)} = \lambda_k^{(n)} / (x_k - x_{n+1}) \qquad k = 0, \dots, n$$

ermitteln kann. Den fehlenden Wert $\lambda_{n+1}^{(n+1)}$ bestimmt man unter Nutzung von (3.12) durch

$$\lambda_{n+1}^{(n+1)} = -\sum_{k=0}^{n} \lambda_k^{(n+1)}$$

Insgesamt braucht man zur Berechnung der p_k $2n$ Multipl. und n Additionen und damit zur Polynomauswertung mit der baryzentrischen Formel $3n$ Multiplikationen und $3n$ Additionen, wobei der zusätzliche Aufwand bei Hinzunahme einer $(n+2)$-Stützstelle mit n Multiplikationen und n Additionen moderat ist.

Verfahren von Neville und Aitken

- Es ist vergleichbar mit der Herangehensweise bei der Newton-Interpolation

Aus dem Hilfssatz 3.9 folgt mit

$$y_0 =: P_0^*(x)$$
$$\vdots$$
$$y_n =: P_n^*(x)$$

die Rekursion

$$P^*_{0,1}(x) = \frac{(x-x_0)P_1^*(x) - (x-x_1)P_0^*(x)}{x_1 - x_0}$$
$$\vdots$$
$$P^*_{n-1,n}(x) = \frac{(x-x_{n-1})P_n^*(x) - (x-x_n)P^*_{n-1}(x)}{x_n - x_{n-1}}$$

usw.

$$P^*_{0,1,2}(x) = \frac{(x-x_0)P^*_{1,2}(x) - (x-x_2)P^*_{0,1}(x)}{x_2 - x_0} \quad \dots$$

Für den Algorithmus von Neville und Aitken folgt das Schema zur Berechnung von p an der Stelle x

x_0	$y_0 = P_0^*(x)$		
		$P^*_{0,1}(x)$	
x_1	$y_1 = P_1^*(x)$		$P^*_{0,1,2}(x) = \frac{(x-x_0)P^*_{1,2}(x) - (x-x_2)P^*_{0,1}(x)}{x_2 - x_0}$
		$P^*_{1,2}(x)$	
x_2	$y_2 = P_2^*(x)$		$P^*_{1,2,3}(x) = \frac{(x-x_1)P^*_{2,3}(x) - (x-x_3)P^*_{1,2}(x)}{x_3 - x_1}$ $\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_n	$y_n = P_n^*(x)$	$P^*_{n-1,n}(x)$	

Beispiel

x_k	y_k
0	1
1	3
3	2

Polynomwert an der Stelle $x = 2$ soll berechnet werden

0 1 $\quad P^*_{0,1}(2) = \frac{(2-0)3-(2-1)1}{1-0} = 5$

1 3

3 2 $\quad P^*_{1,2}(2) = \frac{(2-1)2-(2-3)3}{3-1} = \frac{5}{2}$

$$P_{0,1,2}(2) = \frac{(2-0)\frac{5}{2}-(2-3)5}{3-0} = \frac{10}{3}$$

Hermite-Interpolation

Hat man einen Stützpunkt (x_0, y_0) vorgegeben, so ist damit ein Polynom 0-ten Grades festgelegt (Gerade parallel zur x-Achse). Hat man an der Stelle noch eine Ableitungsinformation, d.h. (x_0, y_0'), dann ist damit eine Gerade durch den Punkt (x_0, y_0) mit dem Anstieg y_0' festgelegt, also ein Polynom mit einem Grad ≤ 1 ist.

Satz 3.12

Sei f eine $(n+1)$-mal stetig diff'bare Funktion in einem Intervall um den Punkt x. Dann gilt

$$\lim_{x_0 \to x \dots x_n \to x} f[x_0, x_1, \dots, x_n, x] = \frac{f^{(n+1)}(x)}{(n+1)!}$$

Beweis Vollst. Induktion, HW 5

Der Satz 3.12 bedarf keiner Beweis

Definition 3.13

$$f[\underbrace{x, x, \dots, x}_{n+1}] = \frac{f^{(n+1)}(x)}{(n+1)!} \qquad (3.13)$$

Auf der Basis von Def. 3.13 entstehen generierte Differenzen wieder rekursiv, z.B.

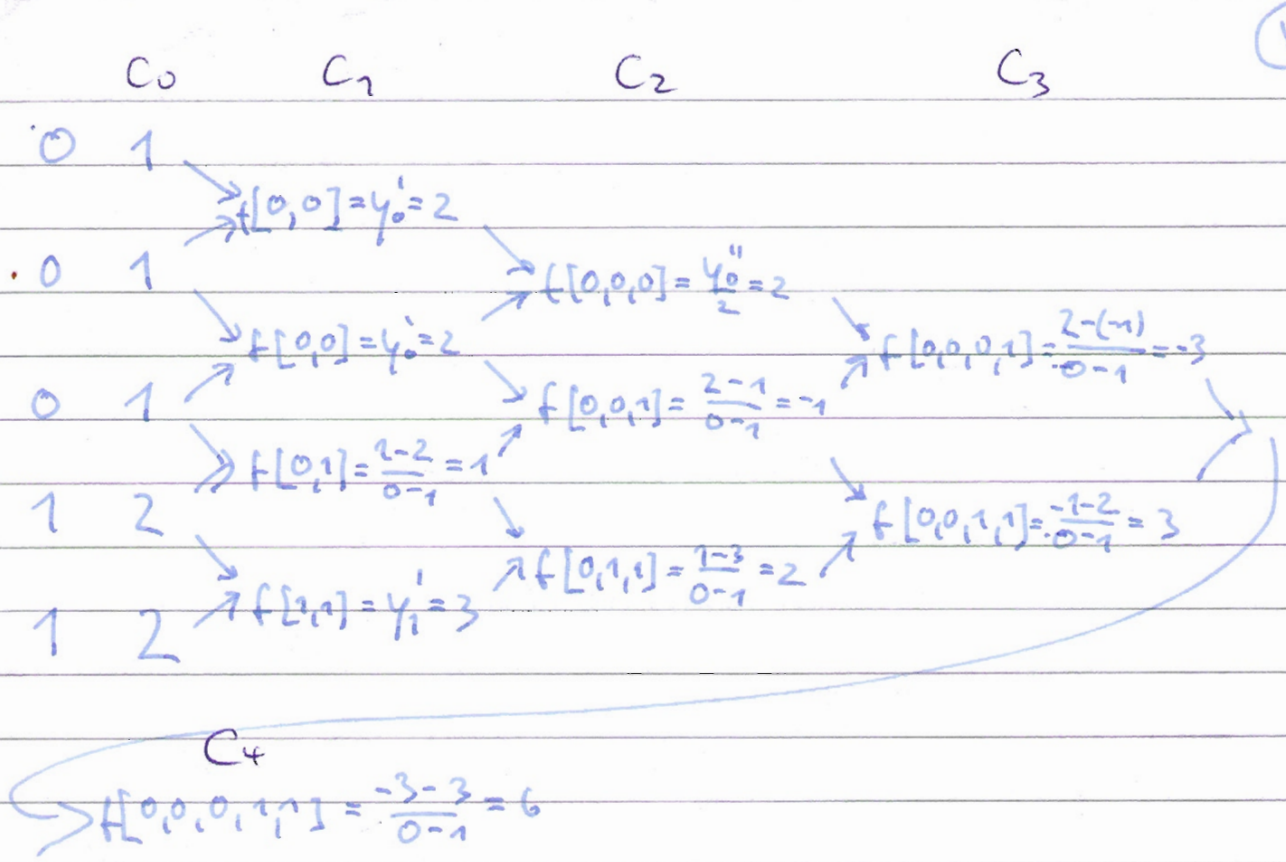
$$f[x_0, x_1, x_1] = \frac{f[x_0, x_1] - f[x_1, x_1]}{x_0 - x_1} \left(= \frac{f[x_0, x_1] - f'(x_1)}{x_0 - x_1} \right)$$

$$f[x_0, x_0, x_0, x_1] = \frac{f[x_0, x_0, x_0] - f[x_0, x_0, x_1]}{x_0 - x_1}$$

Man überlegt sich, dass zur Bestimmung der Polynomkoeffizienten eines Hermiteschen Interpolationspolynoms (zur Erfüllung von Interpol. bedingungen bei Berücksichtigung von Ableitungsinformationen) das folgende Schema für die Bedingungen

$$(x_0, y_0) = (0, 1)$$
$$(x_0, y_0') = (0, 2)$$
$$(x_0, y_0'') = (0, 4)$$
$$(x_1, y_1) = (1, 2)$$
$$(x_1, y_1') = (1, 3)$$

die Form



Daraus folgt das Hermite-Interpolationspolynom

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

also für $x_0 = 0, x_1 = 1$ und die Koeff. aus dem Schema

$$P(x) = 1 + 2(x-0) + 2(x-0)(x-0) - 3(x-0)(x-0)(x-0)$$
$$+ 6(x-0)(x-0)(x-0)(x-1)$$
$$= 1 + 2x + 2x^2 - 3x^3 + 6x^3(x-1)$$

Fehlerabschätzung der Polynominterpolation

- Handelt es sich bei den Stützpunkten (x_k, y_k) nicht um diskrete Messwerte, sondern um die Wertetabelle einer gegebenen Funktion $f(x)$, dann ist der Fehler $f(x) - p_n(x)$, den man bei der Interpolation macht, von Interesse.

Nimmt man zu den Stützwerten $x_0, \dots, x_n$ den Wert $x = x_{n+1}$ hinzu, erfüllt die Interpolationsbedingung $y = f(x) = p_{n+1}(x)$

$$\underbrace{p_{n+1}(x) = f(x)}_{p_{n+1}(x_{n+1}) = y_{n+1}} = p_n(x) + f[x_0, x_1, \dots, x_n, x] \prod_{k=0}^{n} (x - x_k)$$

bzw.

$$f(x) - p_n(x) = f[x_0, x_1, \dots, x_n, x] (x - x_0)(x - x_1) \cdot \dots \cdot (x - x_n) \qquad (3.14)$$

Der folgende Satz liefert die Grundlage für die Abschätzung des Interpolationsfehlers (3.14).

Satz 3.13

Sei $]a, b[=]\min_{0 \le j \le n} x_j, \max_{0 \le j \le n} x_j[$ und sei $p_n(x)$ das Interpolationspolynom zur Wertetabelle $(x_k, f(x_k))$, der $(n+1)$-mal stetig diff'baren Funktion f auf $[a, b]$, wobei die Stützstellen x_k paarweise verschieden sind.

Dann gibt es für jedes $\bar{x} \in]a, b[$ einen zwischen liegenden Wert $\xi = \xi(x_0, \dots, x_n) \in]a, b[$ mit

$$f(\tilde{x}) - P_n(\tilde{x}) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(\tilde{x}-x_0)(\tilde{x}-x_1)\cdots(\tilde{x}-x_n)$$

Beweis

siehe z.B. Bärwolff: Numerik für Ing., Math. und Physiker

Aus dem Satz 3.13 folgt direkt für eine $(n+1)$-mal stetig diff'bare Funktion die Intervallabschätzung

$$|f(x) - P_n(x)| \le \frac{\max\limits_{\xi\in[a,b]} |f^{(n+1)}(\xi)|}{(n+1)!} \left|\prod_{k=0}^{n}(x-x_k)\right| \qquad (3.15)$$

$\omega(x) :=$

Hat man bei den Stützstellen die freie Wahl und soll auf dem Intervall $[a,b]$ interpoliert werden, dann ist die Wahl der Stützstellen des Tschebyscheff-Polynoms $T_{n+1}(x)$ auf $[a,b]$ transformiert, d.h.

$$x_k^* = \frac{a+b}{2} + \frac{b-a}{2}\cos\left(\frac{2(n+1-k)-1}{2(n+1)}\pi\right), \quad k = 0, \dots, n \qquad (3.16)$$

von Vorteil, denn für $\omega^*(x) = \prod_{k=0}^{n}(x-x_k^*)$ gilt

Satz 3.14

Sei x_k äquidistante und x_k^* gemäß (3.16) verteilte Stützstellen des Intervalls $[a,b]$. Dann gilt

$$\max_{x\in[a,b]} |\omega^*(x)| \le \max_{x\in[a,b]} |\omega(x)|,$$ und falls

f bel. oft diff'bar ist, gilt $\lim P_n^*(x) = f(x)$ auf $[a$