# Numerische Mathematik
# MB, MPE, WI/MB, BSc. Angew. Mech.

## Sommersemester 2010

**Jens Lang**

**FB Mathematik**
**Technische Universität Darmstadt**

# Literatur

**Finck von Finkenstein, Lehn, Schellhaas, Wegmann: Arbeitsbuch für Ingenieure II, Teubner 2006**

**Dahmen, Reusken: Numerik für Ingenieure und Naturwissenschaftler, Springer-Lehrbuch 2008**

**Vorlesungsfolien (pdf-file):**

**https://www3.mathematik.tu-darmstadt.de/index.php?id=84&evsid=23&evsver=859**

**Übungen, Lösungen der Übungsaufgaben, Beispielprogramme:**

**https://www3.mathematik.tu-darmstadt.de/index.php?id=84&evsid=23&evsver=859**

**Numawww:**

**http://numawww.mathematik.tu-darmstadt.de:8081/**

# Inhaltsverzeichnis

## 4. Lösung nichtlinearer Gleichungssysteme

- Fixpunktiteration
- Newton–Verfahren

## 5. Lineare Ausgleichsprobleme

## 6. Numerische Berechnung von Eigenwerten

## 7. Numerische Berechnung von Differentialgleichungen

- Gewöhnliche Differentialgleichungen
- Partielle Differentialgleichungen

**Rutishauser: Interpolation ist die Kunst, zwischen den Zeilen einer Funktionstabelle zu lesen.**

## Interpolationsaufgabe

**Geg.:** **Stützstellen** $x_0, x_1, \ldots, x_n$

**zugehörige Stützwerte** $f_0, f_1, \ldots, f_n$

**Ges.:** **Interpolationsfunktion** $\Phi(x; a_0, a_1, \ldots, a_k)$

$x \in I \subset \mathbb{R}$, $a_i$ **reell**

$$\Phi(x_i; a_0, a_1, \ldots, a_k) = f_i\,, \quad i = 0, \ldots, n$$

## Beispiele:

1. **Polynominterpolation**

$$\Phi(x; a_0, \ldots, a_n) = a_0 + a_1 x + \ldots + a_n x^n = \sum_{i=0}^{n} a_i x^i$$

2. **Trigonometrische Interpolation**

$$\Phi(x; a_0, \ldots, a_{2n}) = a_0 + \sum_{i=1}^{n} (a_{2i-1} \sin(ix) + a_{2i} \cos(ix))$$

## Anwendungen:

**Funktionsdarstellung, numerische Integration, Extrapolation, numerische Behandlung von Differentialgleichungen**

# Polynominterpolation

**Algorithmen zur Auswertung eines Polynoms und seiner Ableitungen**

**DEF.: $\Pi_n$ – Menge der Polynome vom Grad $\leq n$**

**Geg.: $\Pi_n \ni p_n(x) = \sum_{i=0}^{n} a_i x^i$**

**Ges.: $p_n(x_0)$**

**Hornerschema: Ausklammern des Terms $x$, z.B.**

$$a_3x^3 + a_2x^2 + a_1x + a_0 = ((a_3x + a_2)x + a_1)x + a_0$$

# Polynominterpolation: Hornerschema

**Allgemein:**

| Hornerschema |
| --- |
| $a_n^{(1)} := a_n$<br>$a_i^{(1)} := a_{i+1}^{(1)} x_0 + a_i\,, \quad i = n-1, \ldots, 0$ |
| $p_n(x_0) = a_0^{(1)}$ |

**Aufwand: n (M=Multiplikationen) + n (A=Additionen)**

# Polynominterpolation: Hornerschema

**Beispiel:** $p_4(x) = x^4 - 3x^3 + 2x^2 + 1\,, \quad x_0 = 2$

| $i$ | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|
| $a_i$ | 1 | -3 | 2 | 0 | 1 |
| $x_0 = 2$ | | 2 | -2 | 0 | 0 |
| $a_i^{(1)}$ | 1 | -1 | 0 | 0 | $1 = p_4(2)$ |

# Polynominterpolation: Hornerschema

**Ges.: Ableitungen** $p_n^{(j)}(x_0)\,, j = 1, \ldots, m \leq n$

| **Vollständiges Hornerschema** |
|---|
| $a_i^{(0)} := a_i\,,\ i = 0, \ldots, n$ |
| $\left.\begin{array}{rcl} a_n^{(j+1)} & := & a_n^{(j)} \\ a_i^{(j+1)} & := & a_{i+1}^{(j+1)} x_0 + a_i^{(j)}\,, i = n-1, \ldots, j \end{array}\right\} j = 0, \ldots, m$ |
| $p_n^{(j)}(x_0) = a_j^{(j+1)} \cdot j!\,,\ j = 0, \ldots, m$ |

**Aufwand:** $\sum_{i=0}^{m} (n-i) = n(m+1) - \frac{m(m+1)}{2}$ **(M,A)**

# Polynominterpolation: Hornerschema

**Beispiel:** $p_4(x) = x^4 - 3x^3 + 2x^2 + 1\,, \quad x_0 = 2$

| $i$ | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|
| $a_i^{(0)}$ | 1 | -3 | 2 | 0 | 1 | |
| $x_0 = 2$ | | 2 | -2 | 0 | 0 | |
| $a_i^{(1)}$ | 1 | -1 | 0 | 0 | 1 | $= p_4(2) \hookrightarrow p_4(2) = 1$ |
| $x_0 = 2$ | | 2 | 2 | 4 | | |
| $a_i^{(2)}$ | 1 | 1 | 2 | 4 | | $= p_4^{(1)}(2)/1! \hookrightarrow p_4^{(1)}(2) = 4$ |
| $x_0 = 2$ | | 2 | 6 | | | |
| $a_i^{(3)}$ | 1 | 3 | 8 | | | $= p_4^{(2)}(2)/2! \hookrightarrow p_4^{(2)}(2) = 16$ |
| $x_0 = 2$ | | 2 | | | | |
| $a_i^{(4)}$ | 1 | 5 | | | | $= p_4^{(3)}(2)/3! \hookrightarrow p_4^{(3)}(2) = 30$ |
| $x_0 = 2$ | | | | | | |
| $a_i^{(5)}$ | 1 | | | | | $= p_4^{(4)}(2)/4! \hookrightarrow p_4^{(4)}(2) = 24$ |

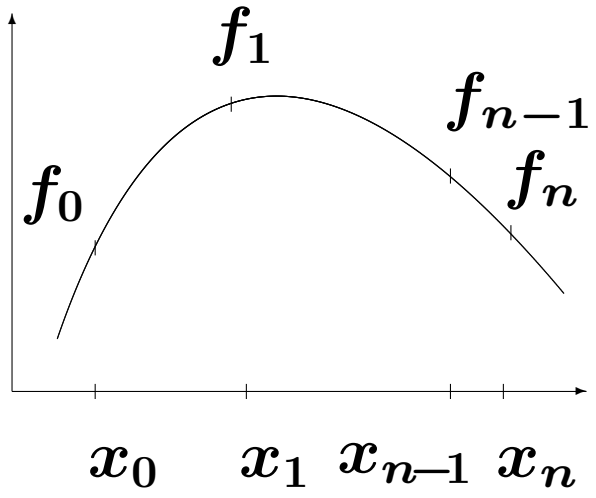
# Lagrangesche Interpolation

## Lagrangesche Interpolationsaufgabe

**Geg.:** $(x_i, f_i),\ i = 0, \ldots, n$, **paarweise verschiedene** $x_i$

**Ges.:** $\Pi_n \ni p_n(x) = a_0 + a_1 x + \ldots + a_n x^n \ :\ p_n(x_i) = f_i \ \ \forall i$

## Geometrische Interpretation



# Lagrangesche Interpolation

**SATZ: Die Lagrangesche Interpolationsaufgabe ist eindeutig lösbar. Es gilt**

$$p_n(x) = \sum_{i=0}^{n} f_i L_i(x) \quad \text{mit} \quad L_i(x) = \prod_{j=0,\, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}$$

**Bezeichnung:**

**lineare (n=1)**
**quadratische (n=2)**
**kubische (n=3) Interpolation usw.**

# Lagrangesche Interpolation

**Beispiel: Quadratische Interpolation mit $(x_0, f_0), (x_1, f_1), (x_2, f_2)$**

$$p_2(x) = L_0(x)f_0 + L_1(x)f_1 + L_2(x)f_2 =$$
$$\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}f_0 + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}f_1 + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}f_2$$

**Vorteile der Lagrange–Darstellung:**

- **$L_j(x)$ sind explizit berechenbar**

**Nachteile:**

- **hoher Aufwand, Berechnung von $n+1$ Polynomen**
- **gewisse Anfälligkeit gegenüber Rundungsfehler**
- **bei Hinzunahme weiterer $(x_i, f_i)$ ändert sich alles**

## Ein konkretes Beispiel:

$$(x_0, x_1, x_2, x_3) = (-1, 0, 2, 3)$$

$$(f_0, f_1, f_2, f_3) = (-1, 3, 11, 27)$$

**Ansatz:** $p = (-1)L_0 + 3L_1 + 11L_2 + 27L_3$

**Lagrange-Polynome**

$$L_0(x) = \frac{x-0}{-1-0} \cdot \frac{x-2}{-1-2} \cdot \frac{x-3}{-1-3} = \frac{1}{12}(-x^3 + 5x^2 - 6x)$$

$$L_1(x) = \frac{x-(-1)}{0-(-1)} \cdot \frac{x-2}{0-2} \cdot \frac{x-3}{0-3} = \frac{1}{12}(+2x^3 - 8x^2 + 2x + 12)$$

$$L_2(x) = \frac{x-(-1)}{2-(-1)} \cdot \frac{x-0}{2-0} \cdot \frac{x-3}{2-3} = \frac{1}{12}(-2x^3 + 4x^2 - 6x)$$

$$L_3(x) = \frac{x-(-1)}{3-(-1)} \cdot \frac{x-0}{3-0} \cdot \frac{x-2}{3-2} = \frac{1}{12}(+x^3 - x^2 - 2x)$$

# Newtonsche Interpolation

**Frage: Wie können Nachteile vermieden werden?**

**Antwort: Newtonsche Interpolation**

**Rekursiver Ansatz:**

$$p_n(x) = c_0 + c_1(x - x_0) + \ldots + c_n(x - x_0) \cdot \ldots \cdot (x - x_{n-1})$$

**Ziel: vorteilhafte Berechnung der $c_i$**

# Newtonsche Interpolation

**DEF.: Seien $(x_i, f_i)$ mit paarweise verschiedenen $x_i$ gegeben. Die k–te *dividierte Differenz* (oder Steigung) $f[x_i, \ldots, x_{i+k}]$ ist rekursiv definiert durch**

$$f[x_i] = f_i\,, \quad i = 0, \ldots, n$$

$$f[x_i, \ldots, x_{i+k}] = \frac{f[x_{i+1}, \ldots, x_{i+k}] - f[x_i, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}$$

$$k = 1, 2, \ldots$$

**Praktische Berechnung mittels Differenzenschema**

| k | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| $x_0$ | $f[x_0] = f_0$ | | | |
| | | $f[x_0, x_1]$ | | |
| $x_1$ | $f[x_1] = f_1$ | | $f[x_0, x_1, x_2]$ | |
| | | $f[x_1, x_2]$ | | $f[x_0, x_1, x_2, x_3]$ |
| $x_2$ | $f[x_2] = f_2$ | | $f[x_1, x_2, x_3]$ | |
| | | $f[x_2, x_3]$ | | |
| $x_3$ | $f[x_3] = f_3$ | | | |

mit $f[x_0, x_1] = \frac{f[x_1] - f[x_0]}{x_1 - x_0}$, $f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0}$, usw.

**Es gilt:** $c_i = f[x_0, \ldots, x_i], \quad i = 0, \ldots, n$

## Unser Beispiel:

$$(x_0, x_1, x_2, x_3) = (-1, 0, 2, 3)$$

$$(f_0, f_1, f_2, f_3) = (-1, 3, 11, 27)$$

## Ansatz:

$$p(x) = c_0 + c_1(x+1) + c_2(x+1)x + c_3(x+1)x(x-2)$$

| $x_i$ | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| -1 | -1 | | | |
| 0 | 3 | 4 | | |
| 2 | 11 | 4 | 0 | |
| 3 | 27 | 16 | 4 | 1 |

$$\Rightarrow \; p(x) = -1 + 4(x+1) + (x+1)x(x-2)$$

**Vorteil**: Hinzunahme von $(x_{n+1}, f_{n+1})$ erfordert (nur) Berechnung einer neuen Schrägzeile im Differenzenschema und

$$p_{n+1}(x) = p_n(x) + c_{n+1}(x - x_0) \cdot \ldots \cdot (x - x_n)$$

## Algorithmus

| Newtonscher Interpolationsalgorithmus |
|---|
| $i = 0, \ldots, n: \quad c_i := f_i$ |
| $k = 1, \ldots, n:$ <br> $i = n, \ldots, k: \quad c_i := \dfrac{c_i - c_{i-1}}{x_i - x_{i-k}}$ |

**Aufwand**: $\frac{n(n+1)}{2}$ (D=Divisionen), $n(n+1)$ (A)

# Der Interpolationsfehler

**Geg.:** $f(x), p_n(x)$ **mit** $p_n(x_i) = f(x_i) = f_i,\ i = 0, \ldots, n$

**Frage: Approximationsfehler?** $\Rightarrow$ **Restglied**

$$\boxed{R_{n+1}(x) = f(x) - p_n(x)}$$

**Abschätzung:**

$$|f(x) - p_n(x)| \leq \max_{\xi \in [\min\ x_i,\, \max\ x_i]} \left| \frac{f^{(n+1)}(\xi)}{(n+1)!} \right| \prod_{j=0}^{n} |x - x_j|$$

# Der Interpolationsfehler

**Beispiele:**

1. **lineare Interpolation,** $n = 1,\ x_0,\ x_1 = x_0 + h,\ x \in [x_0, x_1]$

   $|(x-x_0)(x-x_1)| \leq |(x_0+h/2-x_0)(x_0+h/2-x_1)| = \frac{h^2}{4}$

   **(Maximum wird im Mittelpunkt angenommen!)**

   $\Rightarrow\ |f(x) - p_1(x)| \leq \frac{h^2}{8} \max_{\xi \in [x_0,x_1]} |f^{(2)}(\xi)|$

2. $f(x) = \sin(2\pi x),\ [a,b] = [0,1],\ n = 6$, **äquidistant**

   $\max_{\xi \in [0,1]} |f^{(7)}(\xi)/7!| = 76.706,$

   $x \in [0,1] \Rightarrow \left| \prod_{i=0}^{6} (x - x_i) \right| \leq 3.43 \cdot 10^{-4}$

   $|R_7(x)| \leq 2.63 \cdot 10^{-2},$ **Err** $\approx 1.89 \cdot 10^{-2}$

**Ein weiteres Beispiel: Die Runge–Funktion**

**Ziel: Konstruktion eines Interpolationspolynoms 10. Grades**

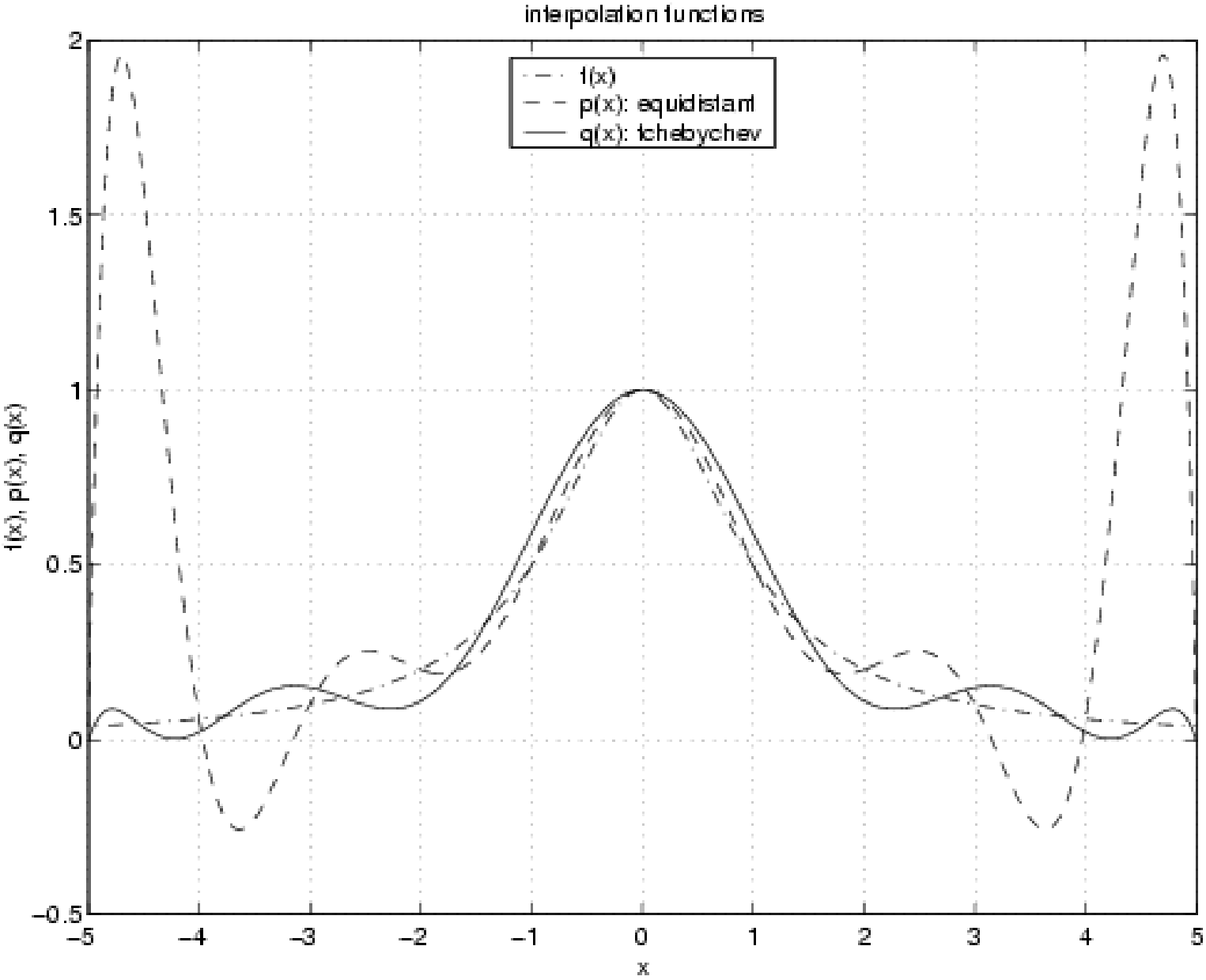
**3.** $f(x) = (1 + x^2)^{-1},\ [a, b] = [-5, 5],\ n = 10$

**3a. äquidistante Stellen**

$$x_i = -5 + i,\ i = 0, \ldots, 10$$

**3b. Tschebyscheff–Knoten**

$$x_i = 5 \cos\left(\frac{2(10 - i) + 1}{22}\pi\right),\ i = 0, \ldots, 10$$


interpolation functions
f(x)
p(x): equidistant
q(x): tchebychev
f(x), p(x), q(x)
x
2
1.5
1
0.5
0
−0.5
−5
−4
−3
−2
−1
0
1
2
3
4
5