Numerik WS 2021/22

Ida Hönigmann

March 29, 2024

Contents

1 Grundbegriffe

1.1 Gegenstand der Numerischen Mathematik

Von der Realität bis zur Interpretation einer Simulation ist es ein langer Weg.

- **Mathematisches Modell** versucht mit Hilfe von mathematischen Formeln (idR. Differentialgl.) die Realität zu beschreiben.
- Die wenigsten Lösungen dieser mathematischen Modelle kann man exakt berechnen, d.h. man approximiert die exakte Lösung mittels **numerischer Simulation** am Rechner.
- Diese **numerische Lösung** wird dann interpretiert und man hofft, dass diese Interpretation die Realität beschreibt.

Jede numerische Simulation zerfällt in kleinere **numerische Probleme**, die geeignet zu lösen sind. Die elementarsten numerischen Probleme sind Gegenstand dieser Vorlesung.

Beispiel 1.
- *Wie approximiert man komplizierte Funktionen mittels einfacher Funktionen (z.B. stückweise Polynome)?*
- *Wie berechnet man Grenzwerte (z.B. Integral, Differential)?*
- *Wie löst man lineare / nichtlineare Gleichungen?*

Jede numerische Simulation ist fehlerbehaftet.

- **Modellfehler:** Das mathematische Modell vereinfacht die Realität.
- **Datenfehler:** Die Eingangsdaten einer Simulation stammen meistens aus physikalischen Messungen und haben daher eine gewisse Mess(un-)genauigkeit.
- **Rundungsfehler:** Auf Rechnern ersetzt die endliche Menge an Gleitkommazahlen das kontinuierliche $\mathbb{R}$, d.h. sowohl die Daten als auch die Rechnungen sind rundungsfehlerbehaftet.
- **Verfahrensfehler:** Viele Probleme werden mathematisch in unendlich dimensionalen Räumen oder mit Limiten formuliert. Beides steht im Rechner nicht zur Verfügung und muss diskretisiert werden.

In der Vorlesung liegt unser Hauptaugenmerk auf dem Verfahrensfehler und dem Aufwand zugehöriger Algorithmen.

1.2 Kondition und Stabilität

Betrachte ein abstraktes Problem. Werte $\Phi : X \to Y$ bei $x \in X$ aus, wobei X, Y geeignete normierte Räume sind. Die **Kondition eines Problems** besagt, wie stark Änderungen in x (z.B. Rundungsfehler) sich auf $\Phi(x)$ auswirken.

Definition 1. *Das Problem ist* ***schlecht konditioniert bzgl. absolutem Fehler****, wenn es eine kleine Störung $\tilde{x}$ von x gibt mit $||\Phi(x) - \Phi(\tilde{x})|| \gg ||x - \tilde{x}||$.*

Das Problem ist ***schlecht konditioniert bzgl. relativem Fehler****, falls $x \neq 0 \neq \Phi(x)$ und es ex. eine kleine Störung $\tilde{x}$ von x gibt mit $\frac{||\Phi(x)-\Phi(\tilde{x})||}{||\Phi(x)||} \gg \frac{||x-\tilde{x}||}{||x||}$.*

Andernfalls bezeichnet man das Problem als ***gut konditioniert (bzgl. abs./rel. Fehler)****.*

Bemerkung 1. *Ist Φ stetig differenzierbar, d.h. $\Phi(x) - \Phi(\tilde{x}) = D\Phi(x)(x - \tilde{x}) + o(||x - \tilde{x}||)$ für $\tilde{x} \to x$ so beschreibt die Ableitung $D\Phi(x) \in L(X, Y)$ wie stark sich Änderungen in x auf den Fehler auswirken.*

Deshalb bezeichnet man $\kappa_{abs}(x) = ||D\Phi(x)||$, $\kappa_{rel}(x) = \frac{||D\Phi(x)||\cdot||x||}{||\Phi(x)||}$ als ***Konditionszahlen (bzgl. abs./rel. Fehler)****, d.h. man ist gut konditioniert für $\kappa_{abs}, \kappa_{rel}$ vergleichsweise klein.*

Definition 2. *Es sei $\tilde{\Phi}$ eine algorithmische Umsetzung von Φ. Der Algorithmus $\tilde{\Phi}$ ist* ***instabil****, wenn es eine kleine Störung $\tilde{x}$ von x gibt, sodass $\underbrace{||\Phi(x) - \tilde{\Phi}(\tilde{x})||}_{\text{tatsächlicher Fehler im Rechner}} \gg \underbrace{||\Phi(x) - \Phi(\tilde{x})||}_{\text{unvermeidlicher Fehler}}$.*

Andernfalls ist der Algorithmus stabil.

Bemerkung 2. *Mir ist bewusst, dass die Symbolik $\gg$ ("wesentlich größer") ungenauer ist, als Sie es aus anderen Vorlesungen kennen. Aber "schlecht konditioniert" und "instabil" hängt halt an der Genauigkeit der Daten und den Erfordernissen des Nutzers!*

In der Vorlesung geht es primär um die Asymptotik, d.h. was könnte im Worst-Case passieren.

Erinnerung/Warnung: Die Arithmetik im Rechner erfüllt weder Assoziativität noch Distributivgesetz, d.h. die Reihenfolge (und Formulierung) der Rechenoperatoren spielt eine Rolle für Stabilität.

Beispiel 2 (schlechte Kondition bei Auslöschung)**.** *Als* ***Auslöschung*** *bezeichnet man das Phänomen, dass bei Subtraktion zweier annähernd gleicher Zahlen im Rechner die hinteren Ziffern (welche rundungsbehaftet sind) signifikant werden. Der relative Fehler kann sogar beliebig groß werden, d.h.* $\Phi : \mathbb{R}^2 \to \mathbb{R}, (x,y) \mapsto x - y$ *hat* $\kappa_{rel}(x,y) = \frac{\sqrt{2}||(x,y)||_2}{|x-y|} \gg 0$ *für* $x \approx y$*.*

Achtung: Oft ist ein Problem gut konditioniert, wird aber in Teilprobleme zerlegt (im Algorithmus) sodass der resultierende Algorithmus instabil wird.

Beispiel 3. *Werte* $\Phi(x) = \frac{1}{x+1} - \frac{1}{x}$ *für* $x \gg 0$ *aus.*

$$\Phi'(x) = -\frac{1}{(x+1)^2} + \frac{1}{x^2} = \frac{(x+1)^2 - x^2}{x^2(x+1)^2} = \frac{2x+1}{x^2(x+1)^2}$$

$$\Phi(x) = \frac{x-(x+1)}{x(x+1)} = \frac{-1}{x(x+1)}$$

$$\kappa_{rel}(x) = \frac{|\Phi'(x)| \cdot |x|}{|\Phi(x)|} = \frac{(2x+1)}{x^2(x+1)^2} x^2(x+1) = 1 + \frac{x}{x+1} \leq 2$$

$\Longrightarrow$ *gut konditioniert!*

Beispiel 4. *Es sei* $||.||$ *eine Norm auf* $\mathbb{K}^n$ *und wir verwenden dieselbe Notation für die induzierte Operatornorm* $||A|| := \sum_{x \in \mathbb{K}^n, x \neq 0} \frac{||Ax||}{||x||}$ *für* $A \in \mathbb{K}^{n \times n}$*.*

Ist A *invertierbar, so bezeichnet* $cond(A) := ||A|| \cdot ||A^{-1}||$ *die* ***Konditionszahl von*** A ***(bzgl.*** $||.||$***).*** *Betrachtet man das Lösungsproblem* $\Phi : \mathbb{K}^n \to \mathbb{K}^n, \Phi(b) = A^{-1}b$*, so gilt für die relative Konditionszahl (mit* $x = A^{-1}b$*)*

$$\kappa_{rel}(b) = \frac{||D\Phi(b)|| \cdot ||b||}{||\Phi(b)||} = \frac{||A^{-1}|| \cdot ||Ax||}{||x||} \leq cond(A),$$

wobei die letzte Abschätzung ***scharf ist****, d.h. es gilt Gleichheit für mindestens ein* b *(und ein* x*).*

1.3 Verfahrensfehler

Im Wesentlichen gibt es zwei Arten von Verfahrensfehlern

- **Abbruchfehler**, wenn ein konvergenter (aber unendlicher) Algorithmus nach endlich vielen Schritten abgebrochen wird.
- **Diskretisierungsfehler**, wenn eine kontinuierliche Größe durch eine diskrete vereinfacht wird, z.B. Differenzenquotienten statt Differenzialquotient.

Beispiel 5 (Abbruchfehler Heron-Verfahren)**.** *Für* $x > 0$ *def.* $y_1 := \frac{1}{2}(1+x), y_{n+1} := \frac{1}{2}(y_n + \frac{x}{y_n})$*.*

$$\Longrightarrow y_{n+1}^2 - x = \frac{1}{4}(y_n^2 + 2x + \frac{x^2}{y_n}) - x = \frac{1}{4}(y_n - \frac{x}{y_n})^2 \geq 0$$

$$\Longrightarrow y_{n+1}^2 \geq x > 0 \text{ und } y_{n+1} - y_n = \frac{1}{2}(y_n + \frac{x}{y_n}) - y_n = \frac{x}{2y_n} - \frac{y_n}{2} = \frac{x - y_n^2}{2y_n} \leq 0$$

$$\Longrightarrow 0 < \sqrt{x} \leq y_{n+1} \leq y_n \Longrightarrow y_n \to y$$

$$\Longrightarrow y = \frac{1}{2}(y + \frac{x}{y}) \Longrightarrow \frac{1}{2}y^2 = \frac{1}{2}y^2 + \frac{1}{2}x \Longrightarrow y^2 = x \Longrightarrow y = \sqrt{x}$$

$\Longrightarrow$ (y_n) *konvergiert monoton fallend gegen* $\sqrt{x}$*. Sobald* $y_n = \sqrt{x}$*, würde auch* $y_{n+1} = \sqrt{x}$ *gelten, d.h. endkonstante Folge.*

Später: Heron-Verfahren ist tatsächlich **quadratisch konvergent**, d.h. ex. $C > 0$ mit $|\sqrt{x} - y_{n+1}| \leq C|\sqrt{x} - y_n|^2$. $\Longrightarrow$ schnelle konvergenz, weil sich korrekte Ziffern pro Schritt verdoppeln!

Beispiel 6 (Diskretesierungsfehler einseitiger Differenzenquotient). *Sei $f : \mathbb{R} \to \mathbb{R}$ differenzierbar, $x \in \mathbb{R}$*

Ziel: Approximiere $\Phi := f'(x)$ durch den einseitigen Diffquot. $\Phi_h = \frac{f(x+h)-f(x)}{h}$

klar: $\Phi_h \to \Phi$ für $h \to 0$, aber die Konvergenz kann beliebig langsam sein. Man interessiert sich in der Numerik auch für Konvergenzraten bzgl. des Diskretisierungsparameters.

Für $f \in \mathcal{C}^2$ (lokal um x) gilt nach Mittelwertsatz

$$f'(x) - \frac{f(x+h)-f(x)}{h} = f'(x) - f'(\zeta) = f''(\xi)(x-\zeta)$$

mit Zwischenstellen $x \leq \xi \leq \zeta \leq x+h$

$$\implies |\Phi - \Phi_n| \leq ||f''||_{L^\infty(x,x+h)} h = \mathcal{O}(h)$$

d.h. hier Konvergenzrate 1 in h.

Definition 3. *Es sei Φ eine kontinuierliche Große mit Diskretisierung Φ_h für $h > 0$. Dann bezeichnet man eine Abschätzung der Form $|\Phi - \Phi_h| = \mathcal{O}(h^\alpha)$ als **a-priori Fehlerabschätzung** mit **Konvergenzrate** $\alpha > 0$ (auch **Konvergenzordung**).*

Natürlich interessiert sich die Numerik für Verfahren, bei denen $\alpha > 0$ möglich groß ist.

Beispiel 7 (zentraler Differenzenquotient). *$f : \mathbb{R} \to \mathbb{R}$ diffbar, $x \in \mathbb{R}$, $\Phi := f'(x)$ und $\Phi_h := \frac{1}{2}\left(\frac{f(x+h)-f(x)}{h} + \frac{f(x)-f(x-h)}{h}\right)$*

klar: $\Phi_h \to \Phi$ für $h \to 0$, $|\Phi - \Phi_h| = \mathcal{O}(h)$ sofern $f \in \mathcal{C}^2$ (lokal um x).

Für $f \in \mathcal{C}^3$ (lokal um x) gilt mit Taylor $f(x+h) = f(x)) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(\zeta)$ für geeignete $x - h \leq \zeta_- \leq x \leq \zeta_+ \leq x+h$

$$\begin{aligned}
\implies \frac{f(x+h)-f(x)}{h} &= f'(x) + \frac{h}{2}f'(x) + \frac{h^2}{6}f'''(\zeta_-) \\
\frac{f(x)-f(x-h)}{h} &= f'(x) - \frac{h}{2}f'(x) + \frac{h^2}{6}f'''(\zeta_+) \\
\implies |\Phi - \Phi_h| &= \frac{h^2}{6}\frac{|f'''(\zeta_+) + f'''(\zeta_-)|}{2} = \mathcal{O}(h^2)
\end{aligned}$$

d.h. Konvergenzrate $\alpha = 2$.

$\implies$ höhere Genauigkeit für gleiches h bzw. gleiche Genauigkeit für größeres h.

Bemerkung 3. *Auslöschung tritt immer auf (insb. bei Diffquot.), aber sie wird abgemildert durch Verfahren höherer Ordnung. Eine andere Möglichkeit für Verfahren höherer Ordnung zur Approximation von $f'(x)$ ist die Verwendung von Polynomapproximation, d.h. $f \approx p$ Polynom und berechne $p'(x) \approx f'(x)$.*

2 Interpolation

Bei einem **Interpolationsproblem** sind im einfachsten Fall Paare (x_j, y_j) gegeben und eine "einfache" Funktion p mit $p(x_j) = y_j \forall j$ gesucht, z.B. Polynome, Splines (= stückweise Polynome), rationale Funktionen (= Quotienten von Polynomen). Verwandt, aber mathematisch schwieriger sind **Approximationsprobleme**. Dabei ist eine Funktion f und eine Norm $||.||$ gegeben, und es wird eine einfache Funktion p gesucht, die $||f-p||$ in dieser Klasse einfacher Fkt. minimiert. Oft ist dabei die Funktion f nur implizit gegeben, d.h. unbekannt.

2.1 Lagrange-Polynominterpolation

Problemstellung: Gegeben sind $n+1$ reelle **Stützstellen** $a \leq x_0 < ... < x_n \leq b$ und **Funktionswerte** $y_0, .., y_n \in \mathbb{K}$. Die **Lagrange-Interpolationsaufgabe** sucht ein Polynom $p \in \mathbb{P}_n = \{p(x) = \sum_{j=0}^n a_j x^j | a_0, ..., a_n \in \mathbb{K}\}$ vom Grad n mit $p(x_j) = y_j \forall j = 0, ..., n$

Lemma 1.

1. *$\mathbb{P}_n$ ist $\mathbb{K}$-Vektorraum mit $\dim \mathbb{P}_n = n+1$.*
2. *Die **Monome** $p_j(x) = x^j, j = 0, ..., n$ sind eine Basis von $\mathbb{P}_n$.*
3. *Die **Lagrange-Polynome** $L_j(x) = \prod_{k=0,k\neq j}^n \frac{x-x_k}{x_j-x_k} \in \mathbb{P}_n$ erfüllen $L_j(x_k) = \delta_{jk}$ für alle $j, k = 0, ..., n$ und bilden eine Basis von $\mathbb{P}_n$.*
4. *Die **Newton-Polynome** $q_j(x) = \prod_{k=0}^{j-1}(x-x_k) \in \mathbb{P}_j$ für $j = 0, ..., n$ bilden eine Basis von $\mathbb{P}_n$.*

Proof. klar: $\mathbb{P}_n$ ist $\mathbb{K}$-Vektorraum, $\dim \mathbb{P}_n \leq n+1$,
zz: $\{L_0, .., L_n\} \subseteq \mathbb{P}_n$ lin. unab.
Sei $\mu_0, ..., \mu_n \in \mathbb{K}$ mit $\sum_{j=0}^n \mu_j L_j(x) = 0 \forall x$
Für $x = x_k$ folgt

$$0 = \sum_{j=0}^{n} \mu_j \underbrace{L_j(x_k)}_{=\delta_{jk}} = \mu_k$$

$\implies$ lin. unab. laut Def. $\implies \dim \mathbb{P}_n \geq n+1 \implies$ Monome + Lagrange Pol. bilden Basis von $\mathbb{P}_n$.
zz. $\{q_0, ..., q_n\} \subseteq \mathbb{P}_n$ lin. unab.
Seien $\mu_0, ..., \mu_n \in \mathbb{K}$ mit $\sum_{j=0}^n \mu_j \underbrace{q_j(x)}_{=\prod_{k=0}^{j-1}(x-x_k)} = 0$.
Für $x = x_0$ folgt $\mu_0 q_0(x) = 0 \implies \mu_0 = 0$.
Für $x = x_1$ folgt $\mu_1 \underbrace{q_1(x)}_{\neq 0} = 0$, also $\mu_1 = 0$. Induktives Vorgehen zeigt $\mu_j = 0 \forall j$. □

Satz 1 (Eindeutigkeit + Existenz)**.** *Betrachte Lagrange-Interpolation zu Stützstellen $a \leq x_0 < ... < x_n \leq b$ und Funktionswerten $y_0, ..., y_n \in \mathbb{K}$. Dann existiert ein eindeutiges $p \in \mathbb{P}_n$ mit $p(x_j) = y_j \forall j$. Dieses wird gegeben durch $p = \sum_{j=0}^n y_j L_j$. Ist $\{q_0, ..., q_n\} \subseteq \mathbb{P}_n$ eine Basis von $\mathbb{P}_n$ und $p = \sum_{j=0}^n \lambda_j q_j$, so löst $\lambda = (\lambda_0, ..., \lambda_n)$ das lineare Gleichungssystem*

$$\underbrace{\begin{pmatrix} q_0(x_0) & ... & q_n(x_0) \\ \vdots & & \vdots \\ q_0(x_n) & ... & q_n(x_n) \end{pmatrix}}_{=:A} \lambda = \begin{pmatrix} y_0 \\ \vdots \\ y_n \end{pmatrix}$$

Die Matrix A ist regulär, d.h. λ ist die eindeutige Lösung.

Proof. Da $L_j(x_k) = \delta_{jk} \forall j,k$ ist offensichtlich, dass $p = \sum_{j=0}^n \mu_j L_j$ genau dann das Interpolationsproblem löst, wenn $\mu_j = y_j \forall j$. $\implies$ Eindeutigkeit + Existenz
Def. Lösungsoperator $\mathcal{P} : \mathbb{K}^{n-1} \to \mathbb{P}_n$ durch $(\mathcal{P}y)(x_j) = y_j \forall j = 0, ..., n \forall y \in \mathbb{K}^{n+1}$
$\implies$ wohldef, bijektiv
Def. Auswertungsoperator $\mathcal{A} : \mathbb{P}_n \to \mathbb{K}^{n+1}, p \mapsto (p(x_0), ..., p(x_n))$
$\implies$ wohldef, linear
$\mathcal{P} \circ \mathcal{A} =$ Identität, $\mathcal{A} \circ \mathcal{P} =$ Identität,
$\implies \mathcal{A} = \mathcal{P}^{-1}, \mathcal{P} = \mathcal{A}^{-1}$
$\implies A$ ist die darstellende Matrix $\mathcal{A}$. $\implies A$ ist regulär, da $\mathcal{A}$ bijektiv, linear. □

Bemerkung 4. *Die Konditionszahl cond(A) der sogenannten* ***Vandermonde-Matrix*** *A hängt stark von der Wahl der Basis ab. Für die Lagrange-Polynome wäre A die Identität. Für die Monome ist cond(A) in der Regel indiskutabel schlecht (hängt an der Wahl der x_j). Die Basiswahl beeinflusst auch die Besetzungsstruktur der Matrix.*

Beispiel 8. *Die Newton-Basis führt auf eine untere Dreiecksmatrix*

$$\begin{pmatrix} 1 & 0 & ... & 0 \\ 1 & q_1(x_1) & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & q_1(x_n) & ... & q_n(x_n) \end{pmatrix}$$

d.h. das lineare GLS kann in $\mathcal{O}(n^2)$ statt $\mathcal{O}(n^3)$ gelöst werden.

Lemma 2 (Horner-Schema)**.** *Sei $p(x) = \sum_{j=0}^n \lambda_j \underbrace{q_j(x)}_{\prod_{k=0}^{j-1}(x-x_k)}$*

Für einen Auswertungspunkt $x \in \mathbb{R}$ betrachte

- $y = \lambda_n$
- *for* $k = n-1 : -1_0$
- $y = (x - x_k)y + \lambda_k$

- *end*

$\implies$ *Der Algorithmus berechnet in $3n$ Operationen den Funktionswert $y = p(x)$.*

Satz 2 (Interpolationsfehlerdarstellung)**.** *Sei $f \in \mathcal{C}^{n+1}[a,b], 0 \leq m \leq n, p \in \mathbb{P}_n$ mit $p(x_j) = f(x_j) \forall j = 0, ..., n$, wobei $a \leq x_0 < ... < x_n \leq b$*

$$\implies f^{(m)}(x) - p^{(m)}(x) = \frac{f^{(n+1)(\xi)}}{(n+1-m)!} \prod_{l=0}^{n-m} (x - \zeta_l),$$

wobei $\xi = \xi(m,x)$ und $\zeta_l = \zeta_l(m, x, x_0, .., x_n)$ in $[a,b]$
Für $m = 0$ gilt $\zeta_l = x_l \forall l$.

Proof. $e := f - p \in \mathcal{C}^{n+1}[a,b]$
$\implies$ e hat mindestens $n+1$ Nullstenen (bei x_l) $\implies$ e' hat mindestens n Nullstellen $\implies$ $e^{(m)}$ hat mindestens $n+1-m$ Nullstellen $a < \zeta_0 < ... < \zeta_{n-m} < b$
o.B.d.A. $x \notin \{\zeta_0, ..., \zeta_{n-m}\}$
Def. $F(y) := e^{(m)}(x)w(y) - e^{(m)}(y)w(x)$ mit $w(x) := \prod_{l=0}^{m-n}(y - \zeta_l)$
$\implies$ F hat $n+2-m$ Nullstellen $\implies$ $F^{(n+1-m)}$ hat mind. 1 Nullstelle ξ

$$0 = F^{(n+1-m)(\xi)} = \underbrace{e^{(m)}(x)}_{=f^{(m)}(x)-p^{(m)}(x)} \underbrace{w^{(n+1-m)}(\xi)}_{=(n+1-m)!} - \underbrace{e^{(n+1)}(\xi)}_{=f^{n+1}(\xi)} \underbrace{w(x)}_{=\prod_{l=0}^{n-m}(x-\zeta_l)}$$

□

Korollar 1 (Interpolationsfehler-Abschätzung)**.** *Seien $f \in \mathcal{C}^{n+1}[a,b]$ reell- oder komplexwertig, $a \leq x_0 < ... < x_n \leq b, p \in \mathbb{P}_n$ mit $p(x_j) = f(x_j) \forall j = 0, ..., n, 0 \leq m \leq n$*

$$\implies ||f^{(m)} - p^{(m)}||_{L^\infty(a,b)} \leq C_\mathbb{K} \frac{||f^{(n+1)}||_{L^\infty(a,b)}}{(n+1-m)!}(b-a)^{n+1-m}$$

mit $C_\mathbb{K} = 1$ für reelwertiges f, $C_\mathbb{K} = 2$ für komplexwertiges f.

Proof. klar für $\mathbb{K} = \mathbb{R}$.
Für $\mathbb{K} = \mathbb{C}$, betrachte $Re(f), Im(f) \in \mathcal{C}^{n+1}[a,b]$. □

Bemerkung 5. *Aus der Fehlerabschätzung und der Konvergenz der Exponentialreihe $exp(y) = \sum_{k=0}^{\infty} \frac{y^k}{k!}$ folgt, dass der Interpolationsfehler "schnell" konvergiert, sofern sich die Ableitungen $||f^{(k)}||_{L^\infty(a,b)}$ gut verhalten (z.B. $||f^{(k)}||_{L^\infty(a,b)} \leq M < \infty$).*

Bemerkung 6. *Für $m = 1, a = x$ und $b = x + h$ folgt*

$$|f'(x) - p'(x)| \leq ||f' - p'||_{L^\infty(x,x+h)} \leq C_\mathbb{K} \frac{||f^{(n+1)}||_{L^\infty(x,x+h)}}{n!} h^n$$

d.h. besser als die Differenzquotienten aus Kapitel 1.

2.2 Cebysev-Knoten

Definition 4. *Sei $f : \mathbb{R} \to \mathbb{R}$ eine Funktion und $x \in \mathbb{R}$. Man nennt x eine **n-fache Nullstelle von** f, gdw. $f(x) = 0$ und f is lokal um x $(n-1)$-mal diffbar mit $f^{(k)}(x) = 0 \forall k = 1, ..., n-1$. Wir schreiben $n(f,x) \in \mathbb{N}_0$ für die Vielfachheit.*

Lemma 3. *Sei $p \in \mathbb{P}_n$ mit Nullstellen $x_1 < ... < x_k$ und $N := \sum_{j=1}^n n(f, x_j) > n$.*
$\implies$ *$p = 0$, d.h. ein nicht-triviales Polynom vom Grad n hat $\leq n$ Nullstellen, wobei diese mit Vielfachheit gezählt werden.*

Proof. Induktion nach n.
Ind.anf.: $n = 0$, d.h. p ist konstant mit mind. einer Nullstelle $\implies p = 0$✓
Ind.hyp: Die Aussage gelte für alle Polynome $q \in \mathbb{P}_{n-1}$.
$p \in \mathbb{P}_n$ hat Nullstellen $x_1 < ... < x_k$ und $N = \sum_{j=1}^k n(p, x_j) > n$
$\implies$ $p \in \mathbb{P}_{n-1}$ hat Nullstellen $\zeta_1 < ... < \zeta_{k-1}$ mit $x_j < \zeta_j < x_{j+1}$ (nach MWS) und bei allen x_j mit $n(p, x_j) > 1$.

Für die Nullstellen von $p' \in \mathbb{P}_{n-1}$ gilt also

$$\sum_{j=1}^{k-1} \underbrace{n(p', \zeta_j)}_{\geq 1} + \sum_{j=1}^{k} \max\{n(p, x_j) - 1, 0\} \geq -1 + \underbrace{\sum_{j=1}^{k} \underbrace{(\max\{n(p, x_j) - 1, 0\} + 1)}_{\geq n(p,x_j)}}_{=N>n} > n - 1$$

$\implies p' = 0 \implies p$ konstant $\implies p = 0.$ □

Bemerkung 7. *Aus der linearen Algebra wissen wir, dass sich jedes Polynom $p \in \mathbb{P}_n$ mit Nullstelle x_0 in der Form $p(x) = q(x)(x - x_0)$ schreiben lässt mit $q \in \mathbb{P}_{n-1}$, sog.* ***Polynomdivision****.*

Ferner gilt der **Fundamentalsatz der Algebra**: Für jedes $p \in \mathbb{P}_n$ existieren $x_1, ..., x_n \in \mathbb{C}$ und $\lambda \in \mathbb{C}$ mit $p(x) = \lambda \prod_{j=1}^{n} (x - x_j)$. Offensichtlich ist diese Aussage viel stärker als "mein Lemma".

Ziel: Für $m = 0$ gilt für alle $x \in [a, b]$

$$|f(x) - p(x)| \leq C_{\mathbb{K}} \frac{||f^{(n+1)}||_{L^\infty(a,b)}}{(n+1)!} \prod_{l=0}^{n} |x - x_l|,$$

wenn f glatt und $p \in \mathbb{P}_n$ Lagrange-Interpolationspolynom zu x_j.

Nun wollen wir die x_j so wählen, dass $\max_{x \in [a,b]} \prod_{l=0}^{n} |x - x_l|$ minimal wird.

Definition 5. *Für $n \in \mathbb{N}_0$ definiere die* ***Cebysev-Polynome (der ersten Art)*** *durch $T_n(t) := \cos(n \arccos t)$ auf $[-1, 1]$.*

Lemma 4.

1. *$T_n(\cos(\Phi)) = \cos(n\Phi) \forall 0 \leq \Phi \leq \pi \forall n \in \mathbb{N}_0$*
2. *Auf $[-1, 1]$ gilt $T_0(t) = 1$, $T_1(t) = t$, $T_{n+1}(t) = 2tT_n(t) - T_{n-1}(t) \forall n \in \mathbb{N}$*
3. *$T_n \in \mathbb{P}_n[-1, 1]$ mit Leitkoeffizient 2^{n-1} für $n \geq 1$*
4. *$||T_n||_{L^\infty(-1,1)} = 1$*
5. *T_n hat in $[-1, 1]$ genau $n + 1$ lokale Extrema $T_n(s_j^{(n)}) = (-1)^j$ mit $s_j^{(n)} = \cos\left(\frac{j\pi}{n}\right)$ für $j = 0, ..., n$*
6. *T_n hat in $[-1, 1]$ genau n einfache Nullstellen $T_n(t_j^{(n)}) = 0$, $t_j^{(n)} = \cos\left(\frac{(2j-1)}{n} \frac{\pi}{2}\right)$ für $j = 1, ..., n$*

nur die sog. Drei-Term-Rekursion in (2). Whl: Additionstheorem des Cosinus: $\cos(x) + \cos(y) = 2 \cos\left(\frac{x+y}{2}\right) \cos\left(\frac{x-y}{2}\right)$
$t = \cos(\Theta), x := (n+1)\Theta, y := (n-1)\Theta$

$$\implies \frac{x+y}{2} = n\Theta, \frac{x-y}{2} = \Theta$$

$$\implies T_{n+1}(t) + T_{n-1}(t) = \cos(\underbrace{(n+1)\Theta}_{x}) + \cos(\underbrace{(n-1)\Theta}_{y}) = 2 \underbrace{\cos\left(\underbrace{\frac{x+y}{2}}_{n\Theta}\right)}_{T_n(t)} \underbrace{\cos\left(\underbrace{\frac{x-y}{2}}_{\Theta}\right)}_{=t}$$

□

Satz 3 (Optimalität der Cebysev-Knoten)**.** *Betrachte die affine Transformation $\Psi : [-1, 1] \to [a, b], \Psi(t) = \frac{1}{2}\{(a + b) + t(b - a)\}$.*

Seien $t_1^{(n+1)}, ..., t_{n+1}^{(n+1)}$ die Nullstellen von T_{n+1}.

$$\implies \min_{x_0,...,x_n \in [a,b]} \max_{x \in [a,b]} \prod_{j=0}^{n} |x - x_j| = \max_{x \in [a,b]} \prod_{j=0}^{n} |x - \Psi(t_{j+1}^{(n+1)})| = \left(\frac{b-a}{2}\right)^{n+1} \frac{1}{2^n}.$$

Die $\Psi(t_{j+1}^{(n+1)})$ für $j = 0, ..., n$ heißen ***Cebysev-Knoten in [a,b]****.*

Proof. 1. zz: $\max_{t \in [-1,1]} \prod_{j=0}^{n} |t - t_{j+1}^{(n+1)}| = \frac{1}{2^n}$

Lemma (iii) + (vi) $\implies T_{n+1}(t) = 2^n \prod_{j=0}^{n} (t - t_{j+1}^{(n+1)})$

Lemma (iv) $\implies 1 = ||T_{n+1}||_{L^\infty(-1,1)} = \max_{t \in [-1,1]} 2^n \prod_{j=0}^{n} |t - t_{j+1}^{(n+1)}|$

2. zz. $\frac{1}{2^n} \leq \inf_{t_0,...,t_n \in [-1,1]} \max_{t\in[-1,1]} \prod_{j=0}^n |t-t_j|$ (dann folgt die Behauptung für $[a,b]=[-1,1]$)

   Annahme: Ex. $t_0, ..., t_n \in [-1,1]$ mit $w(t) := \prod_{j=0}^n (t-t_i)$ erfüllt

$$||w||_{L^\infty(-1,1)} = \max_{t\in[-1,1]} \prod_{j=0}^n |t-t_j| \lneq \frac{1}{2^n}.$$

   Definiere $p := \underbrace{\frac{1}{2^n} T_{n+1}}_{\in \mathbb{P}_{n+1}} - \underbrace{w}_{\in \mathbb{P}_{n+1}} \in \mathbb{P}_n$. Ferner $\frac{1}{2^n} T_{n+1}(s_j^{(n+1)}) = \frac{(-1)^j}{2^n}$ und $|w(s_j^{n+1})| < \frac{1}{2^n}$.

   $\implies p$ hat $n+1$ Vorzeichenwechsel $\implies n+1$ Nullstellen $\implies p=0 \implies w = \frac{1}{2^n} T_{n+1}$ ↯

3. klar: Ψ ist Bijektion von $[-1,1]$ auf $[a,b]$

   $\Psi(t) - \Psi(t_{j+1}^{(n+1)}) = \frac{1}{2}\{(t - t_{j+1}^{(n+1)})(b-a)\}$

$$\implies \max_{x\in[a,b]} \prod_{j=0}^n |x - \Psi(t_{j+1}^{(n+1)})| = \max_{t\in[-1,1]} \prod_{j=0}^n |\Psi(t) - \Psi(t_{j+1}^{(n+1)})| =$$

$$\left(\frac{b-a}{2}\right)^{n+1} \underbrace{\max_{t\in[-1,1]} \prod_{j=0}^n |t - t_{j+1}(n+1)|}_{=\frac{1}{2^n}}$$

□

2.3 Lebesgue-Konstante

Satz 4. *Seien $a \leq x_0 < ... < x_n \leq b$ Stützstellen mit zugehörigen Lagrange-Polynomen $L_0, ..., L_n \in \mathbb{P}_n$.*
Def. $I_n : \mathcal{C}[a,b] \to \mathbb{P}_n, I_n f := \sum_{j=0}^n f(x_j) L_j$.
$\implies I_n$ ist eine lineare Projektion auf $\mathbb{P}_n$ mit Operatornorm

$$||I_n|| := \sup_{f\in\mathcal{C}[a,b], f\neq 0} \frac{||I_n f||_{L^\infty(a,b)}}{||f||_{L^\infty(a,b)}} = \max_{x\in[a,b]} \sum_{j=0}^n |L_j(x)| =: \Lambda(x_0, ..., x_n).$$

*Die Zahl $\Lambda(x_0, ..., x_n)$ heißt **Lebesgue-Konstante**.*

Proof. I_n wohldef., linear ✓
Für $p \in \mathbb{P}_n$ gilt $I_n p = p$, da Polynominterpolation eine eindeutige Lsg. hat.
Für $f \in \mathcal{C}[a,b]$ mit $f \neq 0$ gilt

$$\frac{||I_n f||_{L^\infty(a,b)}}{||f||_{L^\infty(a,b)}} = \max_{x\in[a,b]} \left| \sum_{j=0}^n \frac{f(x_j)}{||f||_{L^\infty(a,b)}} L_j(x) \right| \leq \Lambda(x_0, ..., x_n).$$

Um Gleichheit zu zeigen, wähle $x \in [a,b]$ mit $\sum_{j=0}^n |L_j(x)| = \Lambda(x_0, ..., x_n)$. Wähle $f \in \mathcal{C}[a,b]$ als Polygonzug mit $||f||_{L^\infty(a,b)} \leq 1$ und $f(x_j) = sign L_j(x)$. $\implies$ Gleichheit bei obiger Abschätzung. □

Bemerkung 8. *Derselbe Beweis zeigt für $\tilde{I}_n : \mathbb{K}^{n+1} \to \mathbb{P}_k, \tilde{I}_n(y_0, ..., y_n) := \sum_{j=0}^n y_j L_j$, dass $||\tilde{I}_n(y_0, ..., y_n)||_{L^\infty(a,b)} \leq \Lambda \max_{j=0,...,n} |y_j|$ mit Gleichheit für spezielle $y_j = sign(L_j(x))$, wenn $x \in [a,b]$ mit $\sum_{j=0}^n |L_j(x)| = \max_{\tilde{x}\in[a,b]} \sum_{j=0}^n |L_j(x)|$.*
$\implies \tilde{I}$ ist der (lineare) Lösungsoperator der Pol.int.

$$\implies ||\tilde{I}_n(y_0, ..., y_n) - I_n(\tilde{y}_0, ..., \tilde{y}_n)||_{L^\infty(a,b)} \leq \Lambda \max_{j=0,...,n} |y_j - \tilde{y}_i|$$

d.h. Λ ist die abstrakte Konditionszahl der Polynominterpolation.

Abschließend einige Bemerkungen zum Bestapproximationsproblem.

Lemma 5. *X normierter Raum, $Y \leq X$ endlich-dim. Teilraum, $x \in X$*
$\implies$ Ex. $y \in Y$ mit $||x-y||_X = \min_{\tilde{y}\in Y} ||x - \tilde{y}||_X$

Proof. Wähle Folge $(y_n)_{n\in\mathbb{N}} \subseteq Y$ mit

$$\lim_{n\to\infty} ||x-y_n||_X = \inf_{\tilde{y}\in Y} ||x-\tilde{y}||_X$$

$$\implies ||y_n||_X \le \underbrace{||x-y_n||_X}_{\text{glm. beschränkt wegen Konvergenz}} + ||x||_X \implies \sup_{n\in\mathbb{N}} ||y_n||_X \le M < \infty$$

Da Y endl.-dim., gilt der Satz von Bolzano-Weierstraß, d.h. $(y_n)_{n\in\mathbb{N}}$ hat eine konvergente Teilfolge. O.B.d.A. ex. $y \in Y$ mit $||y-y_n||_X \to 0$ für $n \to \infty$.

$||x-y||_X = \lim_{n\to\infty} ||x-y_n||_X = \inf_{\tilde{y}\in Y} ||x-\tilde{y}||_X$. ☐

Bemerkung 9. *Mit Satz über Lebesgue-Konstante und dem Lemma gilt für alle $q \in \mathbb{P}_n$*

$$\underbrace{||f-I_nf||_{L^\infty(a,b)}}_{\ge \min_{q\in\mathbb{P}_n} ||f-q||_{L^\infty(a,b)}} \le ||f-q||_{L^\infty(a,b)} + \underbrace{||I_n(f-q)||_{L^\infty(a,b)}}_{\le \Lambda ||f-q||_{L^\infty(a,b)}}$$

$$\implies \min_{q\in\mathbb{P}_n} ||f-q||_{L^\infty(a,b)} \le ||f-I_nf||_{L^\infty(a,b)} \le (1+\Lambda) \min_{q\in\mathbb{P}_n} ||f-q||_{L^\infty(a,b)}$$

Bemerkung 10. *Nach Satz von Weierstraß gilt $\lim_{n\to\infty} \min_{p\in\mathbb{P}_n} ||f-p||_{L^\infty(a,b)} = 0 \forall f \in \mathcal{C}[a,b]$.*

Nach Satz von Faber gilt allerdings, dass es für jede Folge von Stützstellen $(x_0^{(n)}, ..., x_n^{(n)})_{n\in\mathbb{N}}$ eine Funktion $f \in \mathcal{C}[a,b]$ mit der Eigenschaft, dass $||f - I_n^{(n)} f||_{L^\infty(a,b)}$ divergiert!

Insbesondere muss also $\Lambda_n^{(n)} \to \infty$ gelten!

Bemerkung 11. *Für äquidistante Stützstellen divergiert Λ_n exponentiell schnell. Für Cebysev-Knoten gilt allerdings $\Lambda_n = \mathcal{O}(\log n)$.*

Bemerkung 12. *Der* ***Remez-Algorithmus*** *berechnet (in unendlich vielen Schritten) ein Polynom $q \in \mathbb{P}_n$ mit $||f-q||_{L^\infty(a,b)} = \min_{p\in\mathbb{P}_n} ||f-p||_{L^\infty(a,b)} \forall f \in \mathcal{C}[a,b]$.*

Startwert ist dafür der Cebysev-Interpoland.

Der ***Alternantensatz von Cebysev*** *zeigt, dass das Bestapprox.polynom $q \in \mathbb{P}_n$ bzgl. $||.||_{L^\infty(a,b)}$ in der Tat eindeutig ist.*

2.4 Auswertung von Interpolationspol.

Satz 5 (Neville-Verfahren)**.** *Seien $a \le x_0 < ... < x_n \le b$ Stützstellen mit Funktionswerten $y_j \in \mathbb{K}$ und $p \in \mathbb{P}_n$ mit $p(x_j) = y_j \forall j = 0, ..., n, x \in [a,b]$ Auswertungspunkt.*

Für $j, m \in \mathbb{N}_0$ mit $j + m \le n$, definiere $p_{j,m} \in \mathbb{P}_m$ als eind. Int.polynom mit $p(x_k) = y_k \forall k = j, ..., j+m$

$$p(x) = p_{0,n}(x)$$
$$p_{j,0}(x) = y_j$$
$$p_{j,m}(x) = \underbrace{\frac{(x-x_j)p_{j+1,m-1}(x) - (x-x_{j+m})p_{j,m-1}}{x_{j+m}-x_j}}_{=:q(x), q\in\mathbb{P}_m}$$

Proof. $q(x) = y_j, q(x_{j+m}) = y_{j+m}, q(x_k) = y_k, k = j+1, ..., n-m+1 \implies q = p_{j,m}$ ☐

Dieser Satz führt auf das induktive **Neville-Schema**

$$\begin{array}{ccccccc}
y_0 & = & p_{0,0}(x) & \to & p_{0,1}(x) & \to \; ... & p_{0,n}(x) = p(x) \\
 & & & \nearrow & & & \\
y_1 & = & p_{1,0}(x) & \to & p_{1,1}(x) & \nearrow & \\
y_2 & = & p_{2,0}(x) & \nearrow & & & \\
 & \vdots & & & & & \\
y_{n-1} & = & p_{n-1,0}(x) & \to & p_{n-1,1}(x) & & \\
y_n & = & p_{n,0}(x) & \nearrow & & &
\end{array}$$

Bemerkung 13.
- *Das Neville-Verfahren ist ein sog.* ***Einschritt-Verfahren****, d.h. eine "neue Spalte" nur mit Hilfe der vorausgegangenen Spalte berechnet.*
- *Wenn man "von oben nach unten rechnet", ist kein zusätzlicher Speicher nötig. In diesem Fall sollte man die "Diagonale" speichern.*

- *Man kann im Neville-Verfahren dann leicht einen neuen Punkt (x_{n+1}, y_{n+1}) hinzunehmen und erhält $p_{0,n+1}(x)$, indem man nur die neue Diagonale rechnet.*

Algorithmus 1 (Neville)**.** *Input: Stützstellen $a \leq x_0 < ... < x_n \leq b$, Funktionswerte $y_0, ..., y_n \in \mathbb{K}$, Auswertungspunkt $x \in \mathbb{R}$*

- *for $m = 1 : n$*
- *for $j = 0 : n - m$*
- $y_j = \frac{(x-x_j)y_{j+1}-(x-x_{j+m})y_j}{x_{j+m}-x_j}$
- *end*
- *end*

Output: $y_0 = p(x)$, wobei $p \in \mathbb{P}_n$ mit $p(x_j) = y_j \forall j$
klar: Speicherbedarf $n + 1$ (überschreiben von y-Vektor), Arithmetischer Aufwand $\frac{7}{2}n(n + 1)$.

Definition 6. *Sei $p = \sum_{j=0}^n \lambda_j x^j \in \mathbb{P}_n$. Dann bezeichnet man λ_n als **führenden Koeffizienten von p bzgl. $\mathbb{P}_n$**.*
*Falls $j = 0$ oder ($\lambda_j \neq 0$ und $\lambda_k = 0 \forall k > j$), so bezeichnet man λ_j als **Leitkoeffizient von p**.*

Satz 6 (Newtons Dividierte Differenzen)**.** *Seien $a \leq x_0 < ... < x_n \leq b$ Stützstellen, $y_j \in \mathbb{K}, p \in \mathbb{P}_n$ mit $p(x_j) = y_j \forall j = 0, ..., n$. Für $j, m \in \mathbb{N}_0$ mit $j + m \leq n$ definiere*

$$y_{j,0} := y_j y_{j,m} := \frac{y_{j+1,m-1} - y_{j,m-1}}{x_{j+m} - x_j}$$

$\Longrightarrow$

1. *$y_{j,m}$ ist der führende Koeff. von $p_{j,m} \in \mathbb{P}_m$ aus dem Neville-Verfahren.*
2. *Mit $\lambda_j := y_{0,j}$ gilt $p(x) = \sum_{j=0}^n \lambda_j \underbrace{\prod_{k=0}^{j-1}(x - x_k)}_{=q_j \in \mathbb{P}_j}$ d.h. die dividierten Differenzen geben die Koeffizienten des Int.pol. bzgl. Newton-Basis.*

Proof. $q_k := p_{0,k} - p_{0,k-1} \in \mathbb{P}_n$ mit führendem Koeff. $y_{0,k}$ und Nullstellen $x_0, ..., x_{n-1}$
$\Longrightarrow$ $q_k = y_{0,k} \prod_{j=0}^{k-1}(x - x_j)$ nach Pol.div.

$$\Longrightarrow p = p_{0,k} = p_{0,0} + \sum_{k=1}^{n} \underbrace{(p_{j,k} - p_{0,k-1})}_{=q_k} = y_{0,0} + \sum_{k=1}^{n} y_{0,k} \prod_{j=0}^{k-1}(x - x_j) = \sum_{k=0}^{n} \underbrace{y_{0,k}}_{=\lambda_k} \prod_{j=0}^{k-1}(x - x_j)$$

□

Schema der dividierten Differenzen

$$\begin{array}{ccccccccc}
y_0 & = & y_{0,0} & \searrow & & & \\
y_1 & = & y_{1,0} & \rightarrow & y_{0,1} & & \\
 & \vdots & & \searrow & & & \\
 & \vdots & & \rightarrow & y_{1,1} & \rightarrow & y_{0,2} \\
 & \vdots & & & & & \ddots \\
y_{n-1} & = & y_{n-1,0} & \searrow & & & \\
y_n & = & y_{n,0} & \rightarrow & y_{n-1,1} & \rightarrow & ... & y_{0,n}
\end{array}$$

$\Longrightarrow$ arithmetischer Aufwand $3\frac{n(n+1)}{2}$, um alle $y_{0,j}$ zu berechnen.

Bemerkung 14.
- *Die dividierten Differenzen sind ein Einschrittverfahren.*
- *Wenn man den y-Vektor überschreibt, braucht man keinen zusätzlichen Speicher.*
- *Das Verfahren löst das Vandermonde-System für die Newton-Basis, aber die Matrix aufzustellen.*

- *Die Auswertung von $p(x)$ erfolgt mit Horner-Schema und Aufwand $3n$ pro $x \in \mathbb{K}$.*

Algorithmus 2. *Input: Stützstellen $x_0 < ... < x_n$, Funktionswerte $y_0, ..., y_n \in \mathbb{K}$*

- *for $m = 1 : n$*
- *for $j = n - m : -1 : 0$*
- $y_{j,m} := \frac{y_{j+m} - y_{j,m-1}}{x_{j+m} - x_j}$
- *end*
- *end*

Output: Koeffizienten des Interpol.pl. $p \in \mathbb{P}_n$ $y_0, ..., y_n$ bzgl. Newton-Basis.

Bemerkung 15. *Will man das Interpolationspolynom $p(x)$ an N Stellen auswerten, so gilt für den Gesamtaufwand: Aufwand(Neville) $= \frac{7}{2}Nn(n+1)$, Aufwand(Div. Diff. + Horner) $= \underbrace{\frac{3}{2}n(n+1)}_{div.\ Diff.} + \underbrace{3Nn}_{Horner}$.*

Es gilt immer: Aufwand(Div. Diff. + Horner) $\leq$ Aufwand(Neville). Wenn man sich den Fortpflanzungsfehler anschaut dann sieht man aber, dass Neville weniger anfällig ist für Auslöschung.

In der Praxis verwendet man deshalb Neville für kleine N und Div. Diff. + Horner für große N.

2.5 Hermite-Polynominterpolation

Satz 7 (Wohlgestelltheit). *Gegeben seien Stützstellen $a \leq x_0 < ... < x_n \leq b$, Funktionswerte $y_j^(k) \in \mathbb{K}$ für $j = 0, ..., n$ und $k = 0, ..., n_j \in \mathbb{N}_0$ (Lagrange $n_j = 0 \forall j$), Def $N := \left(\sum_{j=0}^n (n_j + 1)\right) - 1$*

$\implies$ *Ex. eind. $p \in \mathbb{P}_N$ mit $p^{(k)}(x_j) = y_j^{(k)} \forall j = 0, ..., n, \forall k = 0, ..., n_j$, wobei $p^{(0)} = p$.*

Proof. Betrachte den Auswertungsoperator $\mathcal{A} : \mathbb{P}_N \to \mathbb{K}^{N+1}, \mathcal{A}p := (p(x_0), ..., p^{(n_0)}(x_0), p(x_1), ..., p^{(n_1)}(x_1), ..., p^{(n_n)}(x_n))$

klar: $\mathcal{A}$ ist linear und $\dim \mathbb{P}_N = N + 1$

$\implies \mathcal{A}$ ist $\underbrace{\text{bijektiv}}_{\text{=Behauptung}}$, gdw. $\mathcal{A}$ $\underbrace{\text{injektiv}}_{\text{zu zeigen!}}$ $\underbrace{\text{(oder } \mathcal{A} \text{ ist surj)}}_{\text{"schwierig"}}$.

Sei $p \in \mathbb{P}_N$ mit $\mathcal{A}p = 0$, d.h. x_j eine (n_j+1)-fache Nullstelle von $p \forall j$. $\implies p \in \mathbb{P}_N$ hat $\sum_{j=0}^n (n_j+1) = N+1$ viele Nst. (bzgl. Vielfachheit) $\implies p = 0$. □

Bemerkung 16.

- *Der vorausgegangene Beweis ist das "normale Beweisprinzip" für lineare Interpolationsaufgaben. Klar: Man kann die Interpolationsaufgabe insb. lineares Gleichungssystem (äquivalent) formulieren.*
- *Neville-Verfahren und dividierte Differenzen lassen sich auch für das Hermite-Interpolationsproblem formulieren.*
- *Analog zu Lagrange (dieselbe Basis) kann man Fehlerdarstellung und Feherabschätzung beweisen, z.B.*

$$|f(x) - p(x)| \leq C_{\mathbb{K}} \frac{||f^{(N+1)}||_{L^\infty(a,b)}}{(N+1)!} \prod_{j=0}^n |x - x_j|^{n_j+1}$$

$C_{\mathbb{C}=\sqrt{2}}$ *(vorher 2)*

2.6 Spline-Interpolation

Die Polynominterpolation erfordert hohe Glätte an f, um Fehlerabschätzung zu kriegen. Alternativ kann man deshalb stückweise Polynome betrachten (sog. Splines), um Verfahren und Fehlerkontrolle zu haben, falls f nicht so glatt ist.

Beispiel 9 (affiner Interpolationsspline). *Zu Stützstellen $a = x_0 < x_1 < ... < x_n = b$ und $f \in \mathcal{C}[a, b]$ ist $s \in \mathcal{C}[a, b]$ mit*

- $s|_{[x_{j-1}, x_j]} \in \mathbb{P}_1 \forall j = 1, ..., n$
- $s(x_j) = f(x_j) \forall j = 0, ..., n$

$\implies$ *Offensichtlich eindeutig $s(x) = f(x_{j-1}) \frac{x - x_j}{x_{j-1} - x_j} + f(x_j) \frac{x - x_{j-1}}{x_j - x_{j-1}} \forall j \forall x \in [x_j - 1, x_j]$*

Lemma 6. *Zu* $f \in \mathcal{C}[a,b] \cap \mathcal{C}^2[x_{j-1}, x_j] \forall j = 1, ..., n$ *sei* $s \in \mathcal{C}[a,b]$ *der affine Interpolationsspline.*
Def $h : [a,b] \to \mathbb{R}_{>0}, h|_{[x_{j-1},x_j]} := x_j - x_{j-1}$ ***lokale Netzweite***
$\implies ||f-s||_{L^\infty(a,b)} \leq \frac{C_\mathbb{K}}{8}||h^2 f''||_{L^\infty(a,b)}$

Proof. Sei $x \in [x_{j-1}, x_j]$

$$\implies |f(x) - s(x)| \leq C_\mathbb{K} \frac{||f''||_{L^\infty(x_{j-1},x_j)}}{2} \underbrace{|(x-x_{j-1})(x-x_j)}_{\text{maximal für } x=\frac{x_{j-1}+x_j}{2}}$$

$$\implies ||f-s||_{L^\infty(x_{j-1},x_j)} \leq C_\mathbb{K} \frac{||f''||_{L^\infty(x_{j-1},x_j)}}{2} \cdot \frac{(x_j - x_{j-1})^2}{9} = \frac{C_\mathbb{K}}{8}||h^2 f''||_{L^\infty(x_{j-1},x_j)}$$

□