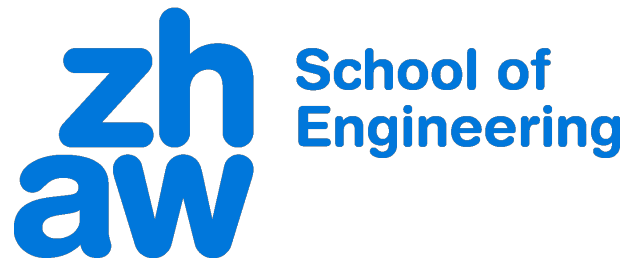


ZHAW Zurich University of Applied Sciences
Winterthur



**Zusammenfassung NUM -
CheatSheet
Studienwochen 1-14**

Written by: Severin Sprenger
June 17, 2026
Zf. NUM - CheatSheet SW 1-14

1 Fehlerrechnung & Kondition

1.1 Fehlerarten

- **Absoluter Fehler:** $E_{abs} = |x - \tilde{x}|$
- **Relativer Fehler:** $E_{rel} = \frac{|x - \tilde{x}|}{|x|}$ (für $x \neq 0$)

1.2 Vektor- und Matrixnormen

- **1-Norm (Spaltensummennorm):** $\|A\|_1 = \max_j \sum_i |a_{ij}|$
- **∞ -Norm (Zeilensummennorm):** $\|A\|_\infty = \max_i \sum_j |a_{ij}|$
- **Frobenius-Norm:** $\|A\|_F = \sqrt{\sum_{i,j} |a_{ij}|^2}$
- **Konditionszahl:** $\kappa(A) = \|A\| \cdot \|A^{-1}\|$. Schlecht konditioniert, wenn $\kappa(A) \gg 1$.

2 Lineare & Nichtlineare Systeme

2.1 Lösungsverfahren für $Ax = b$

- **LR-Zerlegung (LU):** $A = LR$. Vorwärtseinsetzen $Ly = b$, Rückwärtseinsetzen $Rx = y$. (Mit Pivotisierung: $PA = LR$).
- **Cholesky-Zerlegung:** Für symmetrisch positiv definite (spd) Matrizen. $A = LL^T$.
- **QR-Zerlegung:** $A = QR$ (Q orthogonal, R obere Dreiecksmatrix). $Rx = Q^T b$.

2.2 Nichtlineare Systeme ($f(x) = 0$)

- **Newton-Verfahren (1D):** $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$
- **Newton-Verfahren (Mehrdimensional):** $x^{(k+1)} = x^{(k)} - [Df(x^{(k)})]^{-1} f(x^{(k)})$.
 1. Berechne Jacobi-Matrix $J = Df(x^{(k)})$.
 2. Löse LGS: $J \cdot \Delta x = -f(x^{(k)})$.
 3. Update: $x^{(k+1)} = x^{(k)} + \Delta x$.

3 Ausgleichsrechnung (Least Squares)

Minimiere Fehlerquadrate $\|f(x) - y\|_2^2$.

- **Linear (Normalengleichung):** $A^T A x = A^T b$. Lösbar z.B. mit Cholesky oder QR.
- **Nichtlinear (Gauss-Newton):** System $F(x) \approx 0$.
 1. Berechne Jacobi-Matrix $J = DF(x^{(k)})$.
 2. Löse LGS (Normalengleichung): $J^T J \Delta x = -J^T F(x^{(k)})$.
 3. Update: $x^{(k+1)} = x^{(k)} + \Delta x$.

4 Gewöhnliche Differentialgleichungen (AWP)

Problem: $\dot{y}(t) = f(t, y(t))$, $y(t_0) = y_0$. Schrittweite h .

4.1 Einschrittverfahren

- **Expliziter Euler:** $y_{n+1} = y_n + hf(t_n, y_n)$. Konsistenzordnung $p = 1$.
- **Impliziter Euler:** $y_{n+1} = y_n + hf(t_{n+1}, y_{n+1})$. Erfordert oft Newton-Verfahren zur Lösung!
- **Heun-Verfahren ($p = 2$):** $k_1 = f(t_n, y_n)$, $k_2 = f(t_n + h, y_n + hk_1)$. $y_{n+1} = y_n + \frac{h}{2}(k_1 + k_2)$.
- **Klassisches Runge-Kutta (RK4, $p = 4$):** $k_1 = f(t_n, y_n)$
 $k_2 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1)$
 $k_3 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2)$
 $k_4 = f(t_n + h, y_n + hk_3)$
 $y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$.

4.2 Butcher-Tableaus & Stabilität

Ein allgemeines s-stufiges RK-Verfahren wird definiert durch:

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array}$$

- **Explizit:** Matrix A ist strikt untere Dreiecksmatrix.
- **Stabilitätsfunktion $R(z)$:** Wende Verfahren auf Testgleichung $\dot{y} = \lambda y$ an. Resultat hat Form $y_{n+1} = R(z)y_n$ mit $z = \lambda h$.
- **A-Stabilität:** $|R(z)| \leq 1$ für alle $z \in \mathbb{C}$ mit $\text{Re}(z) \leq 0$. Explizite Verfahren sind *nie* A-stabil. Impliziter Euler ist A-stabil.

5 Partielle Differentialgleichungen (FDM)

5.1 Differenzenquotienten

- **Vorwärts:** $u'(x) \approx \frac{u(x+h)-u(x)}{h}$ ($\mathcal{O}(h)$)
- **Rückwärts:** $u'(x) \approx \frac{u(x)-u(x-h)}{h}$ ($\mathcal{O}(h)$)
- **Zentral (1. Abl.):** $u'(x) \approx \frac{u(x+h)-u(x-h)}{2h}$ ($\mathcal{O}(h^2)$)
- **Zentral (2. Abl.):** $u''(x) \approx \frac{u(x+h)-2u(x)+u(x-h)}{h^2}$ ($\mathcal{O}(h^2)$)

5.2 2D Poisson-Gleichung (5-Punkte-Stern)

$\Delta u = u_{xx} + u_{yy} = f(x, y)$. Diskretisierung mit Gitterweite h :

$$\frac{1}{h^2} (u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}) = f_{i,j}$$

Dies führt zu einem linearen Gleichungssystem $Ah^2u = f$, wobei A eine Blocktridiagonalmatrix ist.

6 Spline-Interpolation

Gegeben: Punkte $(x_0, y_0), \dots, (x_n, y_n)$. Kubischer Spline $S(x)$ besteht aus Polynomen $S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$.

- **Interpolationsbedingung:** $S_i(x_i) = y_i$ und $S_i(x_{i+1}) = y_{i+1}$.
- **Stetigkeitsbedingungen:** $S'_i(x_{i+1}) = S'_{i+1}(x_{i+1})$ und $S''_i(x_{i+1}) = S''_{i+1}(x_{i+1})$.
- **Natürlicher Spline:** Randbedingungen $S''(x_0) = 0$ und $S''(x_n) = 0$.