# Numerische Grundlagen – SS 2019

**Formelsammlung**

Lukas Heiland
Yannis Blosch

last updated:

29. April 2019

# Inhaltsverzeichnis

# 1 Grundlagen

## 1.1 Absoluter und relativer Fehler

Um Fehler mathematisch exakt quantifizieren zu können, müssen wir messen. Für das Messen benötigen wir eine Norm:

### 1.1.1 Norm

Eine Norm auf einem $\mathbb{K}-$Vektorraum V (V ist entweder $\mathbb{R}$ oder $\mathbb{C}$) ist eine Abbildung $||\cdot||$ mit folgenden Eigenschaften:

- Definitheit: $\forall v \in V : ||v|| \geq 0$ und dazu $||v|| = 0 \leftrightarrow v = 0$
- Homogenität: $\forall a \in \mathbb{K}, v \in V : ||av|| = |a| \cdot ||v||$
- Dreiecksungleichung: $\forall v, w \in V : ||v + w|| \leq ||v|| + ||w||$

### 1.1.2 Ein paar Standard-Normen, die man immer mal wieder braucht

$$||v||_2 := \sqrt{\sum_{i=1}^{n} v_i^2}$$

$$||v||_1 := \sum_{i=1}^{n} |v_i|$$

$$||v||_\infty := \max_{1 \leq i \leq n} |v_i|$$

### 1.1.3 Absoluter Fehler

Absoluter Fehler zwischen Wert $x$ und der Näherung (dem gestörten $x$) $\tilde{x} = x + \Delta x$ ist die Norm der Differenz:

$$||\Delta x|| = ||x - \tilde{x}||$$

### 1.1.4 Relativer Fehler

Der relative Fehler zwischen Wert und Näherung (gestörtem Wert) ist

$$\delta_x := \frac{||\Delta x||}{||x||} = \frac{||x - \tilde{x}||}{||x||}$$

## 1.2 Gleitpunkt- und Maschinenzahlen, Rundung und Grundrechenarten

### 1.2.1 Gleitpunktzahlen

Für $B \in \mathbb{N} \setminus \{1\}$ und $t \in \mathbb{N}$ ist die Menge der t-stelligen Gleitpunktzahlen zur Basis B

$$\mathbb{F}_{B,t} := \left\{ M \cdot B^E : M = 0 \vee B^{t-1} \leq |M| < B^t; M, E \in \mathbb{Z} \right\}$$

Der relative Abstand von zwei Aufeinanderfolgenden Gleitpunktzahlen ist

$$\frac{(|M|+1)B^E - |M|B^E}{|M|B^E} = \frac{B^E}{|M|B^E} = \frac{1}{|M|}$$

**Auflösung** Die Auflösung ist der maximale relative Abstand in einer Menge von Gleitpunktzahlen:

$$\varrho := \frac{1}{B^{t-1}} = B^{1-t}$$

Daraus leitet sich die sog. Maschinengenauigkeit, d.h. die Schranke für den relativen Abstand einer Zahl $x \in \mathbb{R}$ zur nächsten Gleitpunktzahl, ab:

$$eps := \frac{\varrho}{2}$$

### 1.2.2 Maschinenzahlen

Da eine Menge $\mathbb{F}_{B,t}$ immer noch unendlich ist, kann man sie nicht im Rechner darstellen. Deswegen muss man dem Exponenten E einen Wertebereich geben, damit die Zahlen nicht beliebig weit abhauen:

$$\mathbb{F}_{B,t,\alpha,\beta} := \left\{ M \cdot B^E : M = 0 \vee B^{t-1} \leq |M| < B^t; M, E \in \mathbb{Z}, \alpha \leq E \leq \beta \right\}$$

Maschinenzahlen mit B,t,$\alpha, \beta$ sind also die Gleitpunktzahlen, bei denen der Exponent zwischen $\alpha$ und $\beta$ liegt.

**Kenngrößen** Da es nur endlich viele Maschinenzahlen zu den gegebenen Koeffizienten gibt, kann man bestimmte 'Grenzzahlen' bestimmen:

- kleinste positive Zahl: $\sigma := B^{t-1}B^{\alpha} = B^{t-1+\alpha}$
- größte Zahl: $\lambda := (B^t - 1)B^{\beta}$

### 1.2.3 Runden, Nachbarn

Da mit den Gleitpunktzahlen nur eine Teilmenge der reellen Zahlen abgebildet werden kann, wird es vorkommen, dass das Ergebnis einer Operation nicht genau eine Gleitpunktzahl ist, sondern zwischen zweien liegt. Dann muss das Ergebnis gerundet werden. Zur einfacheren Notation seien nun $f_l(x)$ die nächstkleinere und $f_r(x)$ die nächstgrößere Gleitpunktzahl (linker/rechter Nachbar).

**Wichtigste Rundungsverfahren**

- Abrunden: $rd_-(x) := f_l(x)$
- Aufrunden: $rd_+(x) := f_r(x)$
- Abschneiden $rd_0(x) := \begin{cases} rd_-(x), & x \geq 0 \\ rd_+(x), & x < 0 \end{cases}$
- korrektes Runden: $rd_*(x) := \begin{cases} f_l(x), & x < \frac{f_l(x)+f_r(x)}{2} \\ \text{Nachbar mit gerader Mantisse}, & x = \frac{f_l(x)+f_r(x)}{2} \\ f_r(x), & x > \frac{f_l(x)+f_r(x)}{2} \end{cases}$
  (Sehr häufig in der Praxis)

**Rundungsfehler**

- absoluter Rundungsfehler: $rd(x) - x$
- relativer Rundungsfehler: $\varepsilon = \dfrac{rd(x) - x}{x}, x \neq 0$

Wichtig für später ist die Umformung $rd(x) = x \cdot (1 + \varepsilon)$

### 1.2.4 Fehler beim Rechnen mit Maschinenzahlen

**Notation** Für eine Operation $* \in \{+, -, \cdot, \div\}$ wird die approximative Operation mit einem Punkt darüber dargestellt, also $\dot{*}$

**Relativer Fehler** Der relative Fehler einer Operation $*$ ist

$$\varepsilon_*(a, b) := \frac{(a \dot{*} b) - (a * b)}{a * b}, a * b \neq 0$$

### 1.2.5 A-priori-Fehleranalyse

**Vorwärtsfehleranalyse** Hier wird ein Faktor gesucht, um den das Ergebnis vom exakten Ergebnis abweicht:

$$a \dot{*} b\dot{\Delta}c = (a * b\Delta c) \cdot (1 + \varepsilon), |\varepsilon| \leq ?$$

**Rückwärtsfehleranalyse** Hier ist die Frage: Wie stark sind die Eingabedaten maximal verändert, damit mit veränderten Daten und exakter Rechnung das gleiche rauskommt wie mit den 'korrekten' Daten und approximativer Operation?

$$a \dot{*} b\dot{\Delta}c = (a(1 + \varepsilon_a)) * (b(1 + \varepsilon_b)) * (c(1 + \varepsilon_c)), |\varepsilon_a|, |\varepsilon_b|, |\varepsilon_c| \leq ?$$

## 1.3 Kondition und Stabilität

Die zukünftige Auffassung eines 'Problems' wird es sein, eine Funktion

$$f : X \longrightarrow Y$$

an einer Stelle $x \in X$ auswerten zu müssen. Dabei ist z.B. $X = \mathbb{R}^n$, $Y = \mathbb{R}^m$.

### 1.3.1 Kondition

Die Kondition ist eine Eigenschaft des Problems und beschreibt wie sich Störungen in den Eingabedaten auf das Ergebnis auswirken, wenn man sonst alles 'richtig' macht. Wenn das Problem schlecht konditioniert ist, dann kann man unabhängig vom Lösungsverfahren nie eine hohe Genauigkeit erzielen.

Für ein Problem $f : X \longrightarrow Y$, gestörte Eingabedaten $\tilde{x} = x + \Delta x$ mit gestörtem Ergebnis
$\tilde{y} = f(\tilde{x}) = y + \Delta y$ ist

**die absolute Kondition**

$$\kappa_{abs}(x) = \frac{||\Delta y||_Y}{||\Delta x||_X}$$

**die relative Kondition**

$$\kappa_{rel}(x) = \frac{\delta_y}{\delta_x} = \frac{||\Delta y||_Y}{||\Delta x||_X} \cdot \frac{||x||_X}{||y||_Y} = \kappa_{abs} \cdot \frac{||x||_X}{||y||_Y}$$

### 1.3.2 Stabilität

Die Stabilität ist eine Eigenschaft des Lösungsverfahrens. Sie beschreibt, wie sich Fehler im Verfahren (Diskretisierung, Rundung,...) auf das Ergebnis auswirken (exakte Eingabedaten).
Ein numerischer Algorithmus heißt stabil, wenn für alle Eingabedaten $x$ unter dem Einfluss von Rundungs- und Approximationsfehlern das Ergebnis $\tilde{y}$ das exakte Ergebnis zu leicht modifizierten Eingabedaten ist,

$$\tilde{y} = f(\tilde{x}) \quad mit \quad \tilde{x} \in U_\varepsilon(x)$$

# 2 Lineare Gleichungssysteme

## 2.1 Algorithmen

### 2.1.1 Gauß-Elimination

Verfahren zur Lösung des LGS $Ax = b$, $det(A) \neq 0$. Dabei eliminiert man zuerst, um dann die Rücksubstitution anzuwenden.

**Elimination** Hier versucht man, in jeder Zeile eine Null mehr zu erzeugen. In Zeile 1 bleiben alle $a_{1,j}$ stehen, in Zeile $j$ sollen dann vor Eintrag $a_{j,j}$ nur Nullen stehen. Vor dem $j$-ten Durchlauf der Elimination sieht das LGS so aus:

$$\left(\begin{array}{cccccc|c} a_{1,1} & a_{1,2} & \dots & a_{1,j} & \dots & a_{1,n} & b_1^{(j-1)} \\ 0 & a_{2,2} & \dots & a_{2,j} & \dots & a_{2,n} & b_2^{(j-1)} \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & a_{j,j} & \dots & a_{j,n} & b_j^{(j-1)} \\ 0 & \dots & 0 & a_{j+1,j} & \dots & a_{j+1,n} & b_{j+1}^{(j-1)} \\ \vdots & & \vdots & \vdots & & \vdots & \vdots \\ 0 & \dots & 0 & a_{n,j} & \dots & a_{n,n} & b_n^{(j-1)} \end{array}\right)$$

Danach dann so (alles unter $a_{j,j}$ wurde eliminiert):

$$\left(\begin{array}{cccccccc|c} a_{1,1} & a_{1,2} & \dots & a_{1,j} & \dots & & a_{1,n} & b_1^{(j)} \\ 0 & a_{2,2} & \dots & a_{2,j} & \dots & & a_{2,n} & b_2^{(j)} \\ \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & a_{j,j} & \dots & & a_{j,n} & b_j^{(j)} \\ 0 & \dots & 0 & \mathbf{0} & a_{j+1,j+1} & \dots & a_{j+1,n} & b_{j+1}^{(j)} \\ \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & \mathbf{0} & a_{n,j+1} & \dots & a_{n,n} & b_n^{(j)} \end{array}\right)$$

Am Ende erhält man dann eine obere Dreiecksmatrix, wobei ganz unten nur noch $a_{n,n} = b_n^{(n)}$ steht, wobei $b_n^{(n)}$ der Wert nach n Durchläufen der Elimination ist (der verändert sich ja, wenn man Zeilenumformungen macht).
**Merken:** Das Eliminieren der Einträge in jeder Spalte impliziert, dass für alle i,j ein Koeffizient $\ell_{i,j}$ existiert mit $\ell_{i,j} = \dfrac{a_{i,j}^{(j)}}{a_{j,j}^{(j)}}$. Dann ist

**Rücksubstitution** Hier muss man nur noch von unten nach oben die Lösungen einsetzen: In Zeile $n-1$ steht ja im Prinzip

$$a_{n-1,n-1}x_{n-1} + a_{n-1,n}x_n = b_{n-1}^{(n)}$$

Wir haben die Lösung für $x_n$ aber schon aus der letzten Zeile gegeben, denn da steht

$$a_{n,n}x_n = b_n^{(n)}, \text{ also } x_n = \frac{b_n^{(n)}}{a_{n,n}}$$

Das kann man dann in die vorletzte Zeile einsetzen und erhält dann die Lösung für $x_{n-1}$ usw.

### 2.1.2 LR-Zerlegung

Aufgabenstellung – berechne zu $A \in \mathbb{K}^{n \times n}$ eine obere Dreiecksmatrix R und eine untere Dreiecksmatrix L mit Diagonalelementen 1, so dass $A = L \cdot R$