3 Direkte Verfahren zur Lösung linearer Gleichungssysteme

3.1 Vorbemerkungen

Einige Stimmen zum Ansinnen, lineare Gleichungssysteme mit Gaußelimination auf Computern zu lösen:

„A 78-rowed Matrix would need to be carried to no less than 46 places to ensure even an approximate accuracy in the first decimal place.“

Hotelling, 1943

„Very little is known about the methods so far described, [but] what little information there is tends to indicate that these methods are unstable and that rounding errors accumulate so seriously that the methods are impractical for large values of n.“

Bargmann, Montgomery and von Neumann, 1946

Bezeichnungen:

$$\begin{array}{ccccccccccc}
a_{1,1} & x_1 & + & a_{1,2} & x_2 & + & \cdots & + & a_{1,n} & x_n & = & b_1 \\
a_{2,1} & x_1 & + & a_{2,2} & x_2 & + & \cdots & + & a_{2,n} & x_n & = & b_2 \\
\vdots & & & & & & & & & & \vdots & \vdots \\
a_{m,1} & x_1 & + & a_{m,2} & x_2 & + & \cdots & + & a_{m,n} & x_n & = & b_m
\end{array}$$

ist ein
System von m linearen Gleichungen in den n Unbekannten $x_1, x_2, \ldots, x_n$.

Kurzschreibweise: $\boldsymbol{A\boldsymbol{x}} = \boldsymbol{b}$

mit

$$A = [a_{i,j}]_{1 \le i \le m, 1 \le j \le n} \in \mathbb{C}^{m \times n},$$
$$\boldsymbol{x} = [x_1, \ldots, x_n]^\top \in \mathbb{C}^n,$$
$$\boldsymbol{b} = [b_1, \ldots, b_m]^\top \in \mathbb{C}^m.$$

A heißt Koeffizientenmatrix und $\boldsymbol{b}$ rechte Seite des linearen Gleichungssystems.

Geometrische Deutung: Jede der m Gleichungen repräsentiert eine Hyperebene im $\mathbb{C}^n$:

$$A(i,:)\boldsymbol{x} = b_i \text{ mit } A(i,:) = [a_{i,1}, a_{i,2}, \ldots, a_{i,n}] \quad (i = 1, 2, \ldots, m).$$

Gesucht ist der Durchschnitt dieser Hyperebenen.

Analytische Deutung: Mit $A(:,j) = [a_{1,j}, a_{2,j}, \ldots, a_{m,j}]^\top$ lässt sich $A\boldsymbol{x} = \boldsymbol{b}$ auch als

$$x_1 A(:,1) + x_2 A(:,2) + \cdots + x_n A(:,n) = \boldsymbol{b}$$

schreiben. Gesucht sind also Koeffizienten x_j, mit deren Hilfe man die rechte Seite $\boldsymbol{b}$ als Linearkombination der Spalten von A darstellen kann.

Existenz und Eindeutigkeit der Lösung:

Definiert man das Bild von A (engl. range) durch

$$\mathscr{R}(A) := \left\{ \boldsymbol{y} \in \mathbb{C}^m \; : \; \exists\, \boldsymbol{x} = [x_1, \ldots, x_n]^\top \in \mathbb{C}^n \text{ mit } \boldsymbol{y} = A\boldsymbol{x} = \sum_{j=1}^{n} x_j A(:,j) \right\},$$

so besitzt $A\boldsymbol{x} = \boldsymbol{b}$ genau dann (mindestens) eine Lösung, wenn $\boldsymbol{b} \in \mathscr{R}(A)$ gilt.

Der Kern oder Nullraum von A ist durch

$$\mathscr{N}(A) := \{\boldsymbol{z} \in \mathbb{C}^n \; : \; A\boldsymbol{z} = \boldsymbol{0}\}$$

definiert. Offensichtlich besitzt $A\boldsymbol{x} = \boldsymbol{b}$ höchstens eine Lösung, wenn $\mathscr{N}(A) = \{\boldsymbol{0}\}$ gilt.

Für quadratische Matrizen $A \in \mathbb{C}^{n \times n}$ (in diesem Kapitel werden wir uns ausschließlich mit solchen Matrizen befassen) heißt das:

$A\boldsymbol{x} = \boldsymbol{b}$ ist genau dann eindeutig lösbar, wenn A invertierbar ist.

Satz 3.1. *Für eine quadratische Matrix $A \in \mathbb{C}^{n \times n}$ sind die folgenden fünf Aussagen äquivalent:*

(a) A *ist invertierbar.*

(b) $\mathscr{N}(A) = \{\mathbf{0}\}$.

(c) $\mathscr{R}(A) = \mathbb{C}^n$.

(d) Alle Eigenwerte von A *sind ungleich 0.*

(e) $\det(A) \neq 0$.

3.2 Störungstheorie

Wir betrachten zunächst Störungen der Einheitsmatrix:

Lemma 3.2. *Für $A \in \mathbb{C}^{n \times n}$ gelte $\|A\| < 1$ für eine Matrixnorm $\|\cdot\|$ auf $\mathbb{C}^{n \times n}$. Dann gilt*

(a) $I - A$ ist invertierbar.

(b) Die Inverse besitzt die Darstellung

$$(I - A)^{-1} = \sum_{j=0}^{\infty} A^j \qquad \text{(Neumannsche Reihe).}$$

(c) Für die Inverse gilt die Abschätzung $\|(I - A)^{-1}\| \leq \dfrac{1}{1 - \|A\|}$.

Der folgende Satz gibt ein Maß für den Abstand einer invertierbaren Matrix zur nächstgelegenen singulären Matrix an.

Lemma 3.3. *Zu gegebenen Vektoren* $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{C}^n$ *mit* $\|\boldsymbol{x}\|_p = \|\boldsymbol{y}\|_p = 1$, $1 \le p \le \infty$, *existiert stets eine Matrix* $B \in \mathbb{C}^{n \times n}$ *mit*

$$\|B\|_p = 1 \quad \textit{und} \quad B\boldsymbol{x} = \boldsymbol{y}.$$

Satz 3.4. *Es sei* $A \in \mathbb{C}^{n \times n}$ *invertierbar. Dann gilt*

$$\min \left\{ \frac{\|\Delta A\|_p}{\|A\|_p} \ : \ \Delta A \in \mathbb{C}^{n \times n} \textit{ so, dass } A + \Delta A \textit{ singulär} \right\}$$
$$= \frac{1}{\|A\|_p \|A^{-1}\|_p} \qquad (1 \le p \le \infty).$$

Es sei $\|\cdot\|$ eine Matrixnorm in $\mathbb{C}^{n\times n}$. Ist $A \in \mathbb{C}^{n\times n}$ invertierbar, so heißt

$$\operatorname{cond}(A) = \operatorname{cond}_{\|\cdot\|}(A) := \|A\|\|A^{-1}\| \quad (\geq 1)$$

die Konditionszahl von A. Für singuläre Matrizen A setzen wir $\operatorname{cond}(A) = \infty$. (Kurzschreibweise: $\operatorname{cond}_p(A)$, falls $\|\cdot\| = \|\cdot\|_p$.)

Satz 3.4 besagt, dass der (relative) Abstand (gemessen in $\|\cdot\|_p$) einer regulären Matrix A zur Menge der singulären Matrizen gerade $1/\operatorname{cond}_p(A)$ beträgt.

Man kann $\operatorname{cond}(A)$ als (normalisierte) Fréchet-Ableitung der Abbildung $A \mapsto A^{-1}$ interpretieren:

$$\text{cond}(A) = \lim_{h\to 0} \sup_{\|\Delta A\|\leq h\|A\|} \frac{\|(A+\Delta A)^{-1} - A^{-1}\|}{h} \frac{1}{\|A^{-1}\|}.$$

Satz 3.5. *Mit* $\|\cdot\|$ *bezeichnen wir eine Vektornorm in* $\mathbb{C}^n$ *sowie eine passende Matrixnorm in* $\mathbb{C}^{n\times n}$*. Es seien* $A, \Delta A \in \mathbb{C}^{n\times n}$ *sowie* $\boldsymbol{b}, \Delta\boldsymbol{b} \in \mathbb{C}^n$*. A sei invertierbar und für die Störungsmatrix ΔA gelte* $\|\Delta A\| < 1/\|A^{-1}\|$*. Schließlich seien* $\boldsymbol{x} = A^{-1}\boldsymbol{b}$ *die Lösung des ungestörten LGS* $A\boldsymbol{y} = \boldsymbol{b}$ *und* $\boldsymbol{x} + \Delta\boldsymbol{x} = (A+\Delta A)^{-1}(\boldsymbol{b} + \Delta\boldsymbol{b})$ *die Lösung des gestörten LGS* $(A + \Delta A)\boldsymbol{y} = \boldsymbol{b} + \Delta\boldsymbol{b}$ *(beachten Sie, dass $A + \Delta A$ invertierbar ist).*
Dann folgt

$$\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} \le \frac{\operatorname{cond}(A)}{1 - \operatorname{cond}(A)\frac{\|\Delta A\|}{\|A\|}} \left(\frac{\|\Delta A\|}{\|A\|} + \frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|}\right). \tag{3.1}$$

Interpretation: Relative Fehler in den Daten A und $\boldsymbol{b}$ verstärken sich mit dem Faktor $\operatorname{cond}(A)$ ins Ergebnis $\boldsymbol{x} = A^{-1}\boldsymbol{b}$. Auf einem Rechner mit Rundungseinheit u muss man also mindestens mit einem Fehler der Größenordnung $\operatorname{cond}(A)\cdot u$ in $\boldsymbol{x}$ rechnen – ein LGS $A\boldsymbol{x} = \boldsymbol{b}$ ist durch *kein* Verfahren zuverlässig lösbar, wenn $\operatorname{cond}(A) \ge 1/u$ gilt.

Weitere Eigenschaften der Konditionszahl:

1. Sie ist normabhängig, z.B. $\operatorname{cond}_p(I_n) = \|I_n\|_p \|I_n\|_p = 1$ für $p \geq 1$, aber $\operatorname{cond}_F(I_n) = \|I_n\|_F \|I_n\|_F = n$. Aber sind $\|\cdot\|_\alpha$ und $\|\cdot\|_\beta$ Matrixnormen in $\mathbb{C}^{n\times n}$, so gibt es stets Konstanten $c, C > 0$ mit $c \operatorname{cond}_\alpha(A) \leq \operatorname{cond}_\beta(A) \leq C \operatorname{cond}_\alpha(A)$ für alle A.

   Z.B. $$\frac{1}{n} \operatorname{cond}_2(A) \leq \operatorname{cond}_1(A) \leq n \operatorname{cond}_2(A),$$

   oder $$\frac{1}{n} \operatorname{cond}_\infty(A) \leq \operatorname{cond}_2(A) \leq n \operatorname{cond}_\infty(A).$$

2. Immer gilt cond$(A) \geq 1$.

3. Ist A symmetrisch und positiv definit, so ist

   $$\text{cond}_2(A) = \lambda_{\max}(A)/\lambda_{\min}(A)$$

4. Ist $U \in \mathbb{C}^{n\times n}$ unitär, so ist cond$_2(U) = 1$ und

   $$\text{cond}_2(AU) = \text{cond}_2(UA) = \text{cond}_2(A) \quad \forall A \in \mathbb{C}^{n\times n}.$$

Ist $\tilde{\boldsymbol{x}}$ eine Näherungslösung des linearen Gleichungssystems $A\boldsymbol{x} = \boldsymbol{b}$, so ist der relative Rückwärtsfehler (gemessen in einer Vektor- bzw. Matrixnorm $\|\cdot\|$ auf $\mathbb{C}^n$ bzw. $\mathbb{C}^{n\times n}$, „normwise backward error“) definiert durch

$$\eta(\tilde{\boldsymbol{x}}) = \min\left\{\varepsilon : (A + \Delta A)\tilde{\boldsymbol{x}} = \boldsymbol{b} + \Delta\boldsymbol{b}, \frac{\|\Delta A\|}{\|A\|} \le \varepsilon, \frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|} \le \varepsilon\right\}.$$

Satz 3.6 (Rigal und Gaches, 1967). *Es bezeichne $\|\cdot\|$ eine Vektornorm auf $\mathbb{C}^n$ sowie die dadurch induzierte Matrixnorm auf $\mathbb{C}^{n\times n}$. Der Norm-Rückwärtsfehler einer Näherungslösung $\tilde{\boldsymbol{x}} \in \mathbb{C}^n$ zum linearen Gleichungssystem $A\boldsymbol{x} = \boldsymbol{b}$ mit $A \in \mathbb{C}^{n\times n}$ invertierbar, $\boldsymbol{b} \in \mathbb{C}^n$, ist gegeben durch*

$$\eta(\tilde{\boldsymbol{x}}) = \frac{\|\boldsymbol{r}\|}{\|A\|\,\|\tilde{\boldsymbol{x}}\| + \|\boldsymbol{b}\|}, \tag{3.2}$$

wobei der Vektor $\boldsymbol{r} := \boldsymbol{b} - A\tilde{\boldsymbol{x}}$ das Residuum von $\tilde{\boldsymbol{x}}$ bezüglich des linearen Gleichungssystems bezeichnet.

Normabschätzungen sind nicht immer hinreichend aussagekräftig. In solchen Fällen ist oft eine komponentenweise Analyse notwendig. Der Komponenten-Rückwärtsfehler („componentwise backward error“) ist definiert durch

$$\omega_{E,\boldsymbol{f}}(\tilde{\boldsymbol{x}}) := \min\left\{\varepsilon : (A + \Delta A)\tilde{\boldsymbol{x}} = \boldsymbol{b} + \Delta\boldsymbol{b}, |\Delta A| \le \varepsilon E, |\Delta\boldsymbol{b}| \le \varepsilon\boldsymbol{f},\right\} .$$

(Die Ungleichungen sind komponentenweise zu lesen.) Hierbei besitzen die Matrix E und der Vektor $\boldsymbol{f}$ nichtnegative Einträge und stellen Fehlertoleranzen für jede einzelne Komponente dar. Spezielle Wahlen:

- $E = |A|$, $\boldsymbol{f} = |\boldsymbol{b}|$: liefert den relativen Komponenten-Rückwärtsfehler.
- $E = |A|\boldsymbol{e}\boldsymbol{e}^\top$, $\boldsymbol{e} = [1, \ldots, 1]^\top$, $\boldsymbol{f} = |\boldsymbol{b}|$: Zeilen-Rückwärtsfehler
- $E = \boldsymbol{e}\boldsymbol{e}^\top|A|$, $\boldsymbol{f} = \|\boldsymbol{b}\|_\infty\boldsymbol{e}$: Spalten-Rückwärtsfehler
- $E = \|A\|\boldsymbol{e}\boldsymbol{e}^\top, \boldsymbol{f} = \|\boldsymbol{b}\|\boldsymbol{e}$: liefert bis auf Konstante Norm-Rückwärtsfehler.

Satz 3.7 (Oettli und Prager, 1964). *Der Komponenten-Rückwärtsfehler besitzt die Darstellung*

$$\omega_{E,\boldsymbol{f}}(\tilde{\boldsymbol{x}}) = \max_{i=1}^{n} \frac{|r_i|}{(E|\tilde{\boldsymbol{x}}| + \boldsymbol{f})_i},$$

wobei $\boldsymbol{r} = \boldsymbol{b} - A\tilde{\boldsymbol{x}}$ *und* $\xi/0$ *gleich Null zu setzen ist, falls,* $\xi = 0$ *und andernfalls* $\xi/0 = \infty$.

3.3 Das Lösen von Dreieckssystemen

Eine Matrix der Form

$$L = \begin{bmatrix} \ell_{1,1} & & & \\ \ell_{2,1} & \ell_{2,2} & & \\ \vdots & & \ddots & \\ \ell_{n,1} & \ell_{n,2} & \cdots & \ell_{n,n} \end{bmatrix} \in \mathbb{C}^{n\times n} \qquad (\ell_{i,j} = 0 \text{ für } i < j)$$

heißt untere Dreiecksmatrix. L ist genau dann invertierbar, wenn $\det(L) = \prod_{j=1}^{n} \ell_{j,j} \neq 0$. Die invertierbaren unteren Dreiecksmatrizen bilden bez. der Matrizenmultiplikation eine Gruppe. L heißt normierte untere $\triangle$-Matrix, wenn $\ell_{i,i} = 1$ $(i = 1, 2, \ldots, n)$ ist. Die normierten unteren $\triangle$-Matrizen bilden bez. der Matrizenmultiplikation ebenfalls eine Gruppe. Analoge Aussagen gelten für obere $\triangle$-Matrizen $R = [r_{i,j}] \in \mathbb{C}^{n\times n}$ $(r_{i,j} = 0$ für $i > j)$.

Ist $\det(L) \neq 0$, so besitzt das untere $\triangle$-System $L\boldsymbol{x} = \boldsymbol{c}$ genau eine Lösung, die man durch Vorwärts-Substitution,

$$x_j = \frac{1}{\ell_{j,j}}(c_j - \ell_{j,1}x_1 - \cdots - \ell_{j,j-1}x_{j-1}) \qquad (j = 1, 2, \ldots, n),$$

mit n Divisionen, $n(n-1)/2$ Multiplikationen und $n(n-1)/2$ Additionen, also mit insgesamt n^2 Gleitpunktoperationen berechnen kann.

Obere $\triangle$-Systeme $R\boldsymbol{x} = \boldsymbol{d}$ werden – falls $\det(R) \neq 0$ – analog durch Rückwärts-Substitution,

$$x_j = \frac{1}{r_{j,j}}(d_j - r_{j,j+1}x_{j+1} - \cdots - r_{j,n}x_n) \qquad (j = n, n-1, \ldots, 1),$$

in $O(n^2)$ Gleitpunktoperationen gelöst.

Wir wollen als nächstes zeigen, dass Vorwärts- und Rückwärtssubstitution rückwärtsstabile Algorithmen zur Lösung von Dreieckssystemen sind. Hierzu sind zwei Lemmata hilfreich:

Lemma 3.8. *Sind $|\delta_j| \le u$ und $\rho_j = \pm 1$ für $j = 1, \ldots, n$ und gilt $nu < 1$, so ist*

$$\prod_{j=1}^{n} (1+\delta_j)^{\rho_j} = 1 + \vartheta_n, \qquad \textit{mit} \qquad |\vartheta_n| \le \frac{nu}{1-nu}.$$

Lemma 3.9. *Der Ausdruck $y := \left(c - \sum_{i=1}^{k-1} a_i b_i\right) / b_k$ werde gemäß folgendem Algorithmus in Gleitpunktarithmetik mit Rundungseinheit u ausgewertet:*

$$s := c; \qquad \textit{for } i = 1 : k-1,\ s := s - a_i b_i;\ \textit{end} \qquad s := s/b_k;$$

Dann besitzt der berechnete Wert $\tilde{y} := \mathrm{fl}(y)$ die Darstellung

$$b_k \tilde{y}(1+\vartheta_k) = c - \sum_{i=1}^{k-1} a_i b_i (1+\vartheta_i), \qquad |\vartheta_i| \le \frac{iu}{1-iu}.$$

Satz 3.10. *Die untere* $\triangle$*-Matrix* $L = [\ell_{i,j}] \in \mathbb{C}^{n\times n}$ *sei invertierbar. Die Lösung von* $L\boldsymbol{x} = \boldsymbol{c}$ *werde in Gleitpunktarithmetik mit Rundungseinheit* u *durch Vorwärtssubstitution bestimmt. Dann erfüllt die berechnete Lösung* $\tilde{\boldsymbol{x}}$ *das LGS*

$$(L + \Delta L)\tilde{\boldsymbol{x}} = \boldsymbol{c}$$

mit einer unteren Dreiecksmatrix $\Delta L = [\lambda_{i,j}] \in \mathbb{C}^{n\times n}$*, wobei*

$$|\lambda_{i,j}| \leq n\,u\,|\ell_{i,j}| + O\left(u^2\right) \qquad (1 \leq i, j \leq n)$$

gilt.

3.4 Gauß-Elimination

3.4.1 Geschichtliches

220 v. Chr. – 9 n. Chr. *Chiu Chang Suan Shu (Neun Bücher arithmetischer Technik)*. China. Buch 8 enthält eine Anleitung, LGS bis zur Dimension 5 mittels Elimination zu lösen.

1750. Gabriel Cramer: *Cramersche Regel.* Theoretisch einwandfrei aber praktisch unbrauchbar. (Lösung eines 10×10 Problems erfordert ca. 300 Mio Multiplikationen.)

1809,1823. Carl Friedrich Gauß: *Theoria motus corporum ...* und *Theoria combinationis observationum ...* . Beschreibt Eliminationsverfahren für symmetrische Matrizen aus der Ausgleichsrechnung.

1890. Wilhelm Jordan: *Handbuch der Vermessungskunde.* Erste schriftliche Erwähnung des Gauß-Jordan Algorithmus.

1948. Alan Turing: Darstellung von GE als Folge von Multiplikationen mit unteren Dreiecksmatrizen.

1961. James Wilkinson: Rundungsfehleranalyse von GE.

Stand der Kunst. Zuverlässige Lösung von sehr großen Systemen möglich ($N \approx 10^5$ für vollbesetzte, $N \approx 10^7$ für dünn besetzte Matrizen). Software von hoher Qualität verfügbar (e.g. LAPACK).

3.4.2 Der Algorithmus

Idee: Transformiere $A\boldsymbol{x} = \boldsymbol{b}$, $A \in \mathbb{C}^{n\times n}$ mit $\det(A) \neq 0$, auf ein oberes Dreieckssystem $R\boldsymbol{x} = \boldsymbol{d}$, ohne die Lösung zu verändern. (Löse dann $R\boldsymbol{x} = \boldsymbol{d}$ durch Rückwärts-Substitution.)

Die Lösung eines Gleichungssystems verändert sich nicht, wenn man von der j-ten Gleichung das λ-fache der i-ten Gleichung subtrahiert ($i < j$). Formal: Statt $A\boldsymbol{x} = \boldsymbol{b}$ betrachte $(LA)\boldsymbol{x} = L\boldsymbol{b}$ mit

$$L = I - \boldsymbol{m}_j \boldsymbol{u}_i^\top \in \mathbb{R}^{n\times n}.$$

Dabei ist $\boldsymbol{m}_j = \lambda \boldsymbol{u}_j \in \mathbb{C}^n$; $\boldsymbol{u}_i$ und $\boldsymbol{u}_j$ bezeichnen den i-ten bzw. j-ten Einheitsvektor im $\mathbb{R}^n$.

Beispiel.

$$\begin{bmatrix} 2 & -1 & -3 & 3 \\ 4 & 0 & -3 & 1 \\ 6 & 1 & -1 & 6 \\ -2 & -5 & 4 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ -8 \\ -16 \\ -12 \end{bmatrix}.$$

<u>1. Schritt:</u> Eliminiere x_1 aus der zweiten, dritten und vierten Gleichung. Um das zu erreichen, subtrahiert man das $l_{i,1} = \frac{a_{i,1}}{a_{1,1}}$-fache der ersten Gleichung von der i-ten $(i = 2, 3, 4)$. Hier: $l_{2,1} = 2$, $l_{3,1} = 3$, $l_{4,1} = -1$. Mit

$$L_1 = I - \begin{bmatrix} 0 \\ \ell_{2,1} \\ \ell_{3,1} \\ \ell_{4,1} \end{bmatrix} [1\ 0\ 0\ 0] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 \\ -3 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

ergibt sich

$$L_1[A|\boldsymbol{b}] = \left[\begin{array}{rrrr|r} 2 & -1 & -3 & 3 & 1 \\ 0 & 2 & 3 & -5 & -10 \\ 0 & 4 & 8 & -3 & -19 \\ 0 & -6 & 1 & 4 & -11 \end{array}\right].$$

<u>2. Schritt:</u> Eliminiere x_2 aus der dritten und vierten Gleichung, d.h. subtrahiere das $l_{i,2} = \frac{a_{i,2}}{a_{2,2}}$-fache der zweiten Gleichung von der i-ten $(i = 3, 4)$. Hier: $l_{3,2} = 2, l_{4,2} = -3$. Mit

$$L_2 = I - \begin{bmatrix} 0 \\ 0 \\ \ell_{3,2} \\ \ell_{4,2} \end{bmatrix} [0\ 1\ 0\ 0] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -2 & 1 & 0 \\ 0 & 3 & 0 & 1 \end{bmatrix}$$

ergibt sich

$$L_2L_1[A|\boldsymbol{b}] = \left[\begin{array}{rrrr|r} 2 & -1 & -3 & 3 & 1 \\ 0 & 2 & 3 & -5 & -10 \\ 0 & 0 & 2 & 7 & 1 \\ 0 & 0 & 10 & -11 & -41 \end{array}\right].$$

3. Schritt: Eliminiere x_3 aus der vierten Gleichung, d.h. subtrahiere das $l_{i,3} = \frac{a_{i,3}}{a_{3,3}}$-fache der dritten Gleichung von der i-ten $(i = 4)$. Hier $l_{4,3} = 5$. Mit

$$L_3 = I - \begin{bmatrix} 0 \\ 0 \\ 0 \\ \ell_{4,3} \end{bmatrix} [0\ 0\ 1\ 0] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -5 & 1 \end{bmatrix}$$

ergibt sich

$$\left[\begin{array}{cccc|c} 2 & -1 & -3 & 3 & 1 \\ 0 & 2 & 3 & -5 & -10 \\ 0 & 0 & 2 & 7 & 1 \\ 0 & 0 & 0 & -46 & -46 \end{array}\right].$$

Die Eliminationsphase ist jetzt abgeschlossen und wir können x_4, x_3, x_2, x_1 durch Rückwärts-Substitution bestimmen:

$$x_4 = 1,\ x_3 = \frac{1}{2}[1 - 7x_4] = -3,\ x_2 = \frac{1}{2}[-10 - 3x_3 + 5x_4] = 2,$$
$$x_1 = \frac{1}{2}[1 + x_2 + 3x_3 - 3x_4] = -\frac{9}{2}.$$

Das Gaußsche Eliminationsverfahren liefert eine LR-Zerlegung von A, d.h. eine Faktorisierung $A = L \cdot R$ mit einer normierten unteren $\triangle$-Matrix L und einer oberen $\triangle$-Matrix R.

Im allgemeinen: $L_{n-1} \cdots L_2 L_1 A = R$, d.h.
$A = (L_{n-1} \cdots L_2 L_1)^{-1} R = \left(L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1}\right) R =: L \cdot R$.
In unserem Beispiel:

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 \\ -3 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -2 & 1 & 0 \\ 0 & 3 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -5 & 1 \end{bmatrix}^{-1}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & -3 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 5 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ -1 & -3 & 5 & 1 \end{bmatrix}, \text{ also } A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ -1 & -3 & 5 & 1 \end{bmatrix} \begin{bmatrix} 2 & -1 & -3 & 3 \\ 0 & 2 & 3 & -5 \\ 0 & 0 & 2 & 7 \\ 0 & 0 & 0 & -46 \end{bmatrix}.$$

Lemma 3.11. *Mit $\boldsymbol{u}_i \in \mathbb{R}^n$ bezeichnen wir den i-ten Einheitsvektor. Sei $\boldsymbol{m}_i = [0, \ldots, 0, \ell_{i+1,i}, \ldots, \ell_{n,i}]^\top \in \mathbb{C}^n$ ($1 \leq i \leq n-1$). Für die Matrizen*

$$L_i = I - \boldsymbol{m}_i \boldsymbol{u}_i^\top \in \mathbb{C}^{n \times n}$$

gelten:

(a) $L_i^{-1} = I + \boldsymbol{m}_i \boldsymbol{u}_i^\top$.

(b) $L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1} = I + \boldsymbol{m}_1 \boldsymbol{u}_1^\top + \boldsymbol{m}_2 \boldsymbol{u}_2^\top + \cdots \boldsymbol{m}_{n-1} \boldsymbol{u}_{n-1}^\top$.

Bei der numerischen Rechnung wird die Matrix A durch die „Zwischenmatrizen“ überschrieben (analog für die rechte Seite). Außerdem müssen die Nullen unterhalb der Hauptdiagonalen nicht gespeichert werden, man verwendet diese Felder für die Zahlen $\ell_{i,j}$.

Pseudocode:

for $j = 1 : n-1$
  if $a_{j,j} = 0$ **then**
    stop
  else
    for $i = j+1 : n$
      $a_{i,j} := a_{i,j}/a_{j,j}$
      $b_i := b_i - a_{i,j} b_j$
      for $k = j+1 : n$
        $a_{i,k} := a_{i,k} - a_{i,j} a_{k,j}$

Aufwand: $\frac{2}{3}n^3 + O(n^2)$ flops.

In unserem Beispiel:

$$
\left[\begin{array}{rrrr|r} 2 & -1 & -3 & 3 & 1 \\ 4 & 0 & -3 & 1 & 8 \\ 6 & 1 & -1 & 6 & -16 \\ 2 & 5 & 4 & 1 & -12 \end{array}\right]
\xrightarrow{\text{1.}}
\left[\begin{array}{rrrr|r} 2 & -1 & -3 & 3 & 1 \\ 2 & 2 & 3 & -5 & -10 \\ 3 & 4 & 8 & -3 & -19 \\ -1 & -6 & 1 & 4 & -11 \end{array}\right]
$$

$$
\xrightarrow{\text{2.}}
\left[\begin{array}{rrrr|r} 2 & -1 & -3 & 3 & 1 \\ 2 & 2 & 3 & -5 & -10 \\ 3 & 2 & 2 & 7 & 1 \\ -1 & -3 & 10 & -11 & 41 \end{array}\right]
\xrightarrow{\text{3.}}
\left[\begin{array}{rrrr|r} 2 & -1 & -3 & 3 & 1 \\ 2 & 2 & 3 & -5 & -10 \\ 3 & 2 & 2 & 7 & 1 \\ -1 & -3 & 5 & -46 & -46 \end{array}\right]
$$

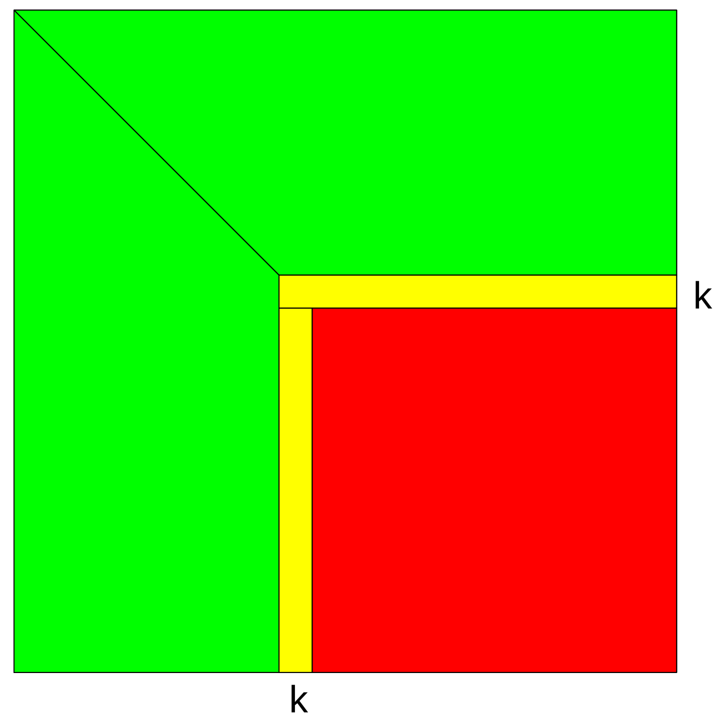
3.4.3 Algorithmische Varianten von Gauß-Elimination

kij-Variante: (Lehrbuchvariante)

for $k = 1 : n-1$
  for $i = k+1 : n$
    $\ell_{i,k} = a_{i,k}/a_{k,k}$
    for $j = k+1 : n$
      $a_{i,j} = a_{i,j} - \ell_{i,k}a_{k,j}$

kji-Variante: (spaltenorientiert)

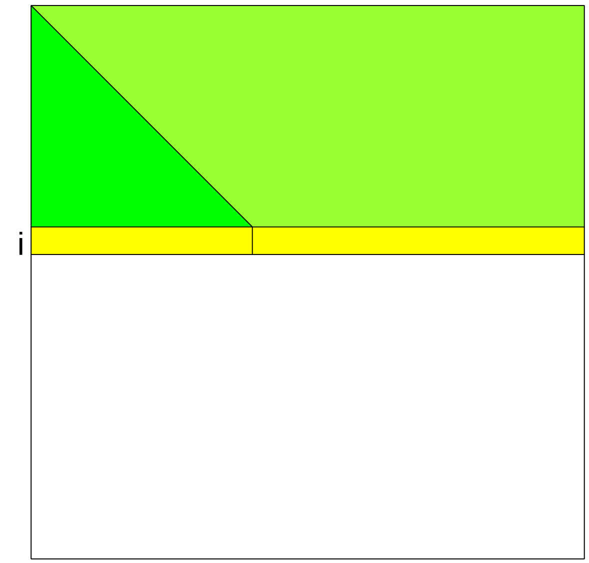
for $k = 1 : n-1$
  for $s = k+1 : n$
    $\ell_{s,k} = a_{s,k}/a_{k,k}$
  for $j = k+1 : n$
    for $i = k+1 : n$
      $a_{i,j} = a_{i,j} - \ell_{i,k}a_{k,j}$



Zugriffsmuster kij / kji Varianten.

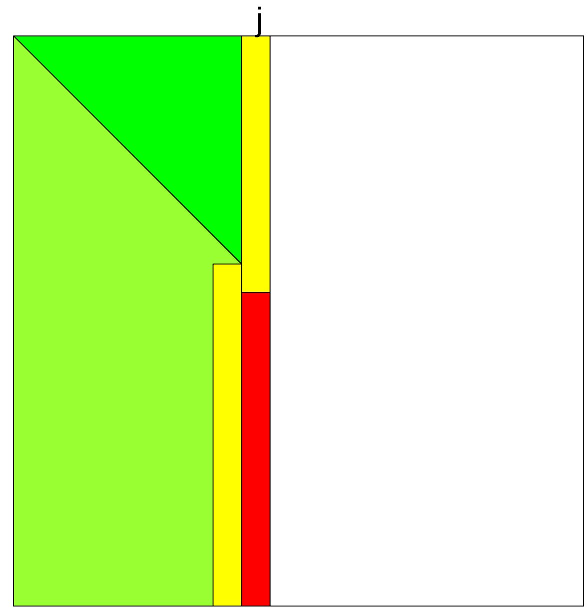
ikj-Variante: (zeilenorientiert)

for $i = 2 : n$
  for $k = 1 : i - 1$
    $\ell_{i,k} = a_{i,k}/a_{k,k}$
    for $j = k + 1 : n$
      $a_{i,j} = a_{i,j} - \ell_{i,k} a_{k,j}$



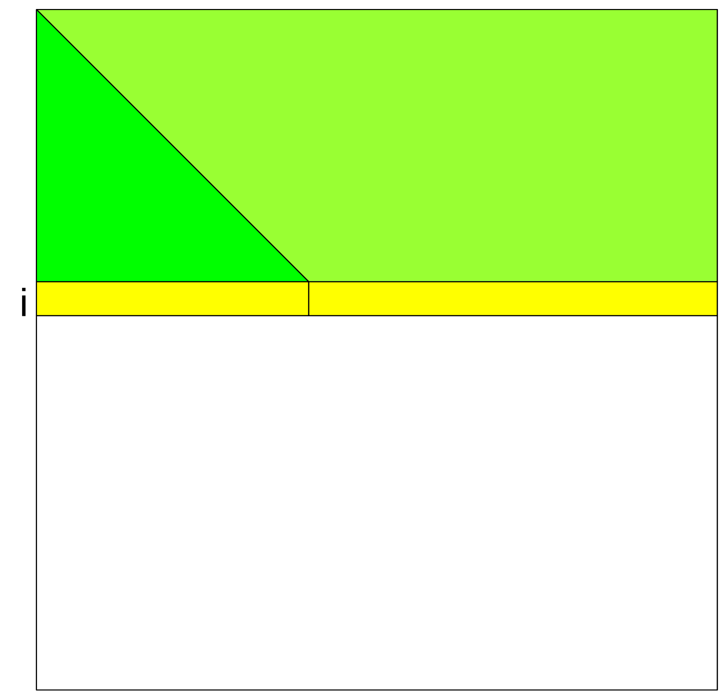
jki-Variante: (spaltenorientiert)

for $j = 2 : n$
  for $s = j : n$
    $\ell_{s,j-1} = a_{s,j-1}/a_{j-1,j-1}$
  for $k = 1 : j - 1$
    for $i = k + 1 : n$
      $a_{i,j} = a_{i,j} - \ell_{i,k} a_{k,j}$



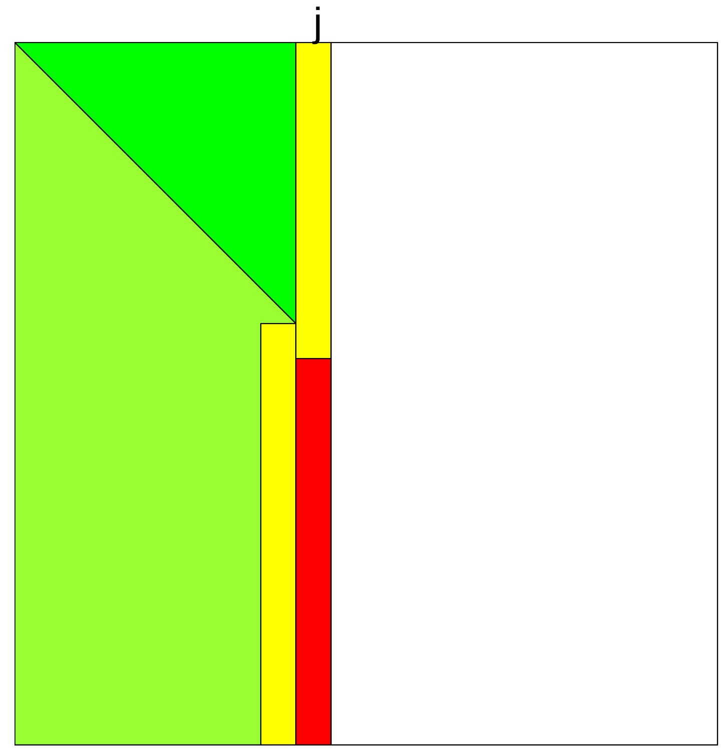
ijk-Variante: (zeilenorientiert)

for $i = 2 : n$
  for $j = 2 : i$
    $\ell_{i,j-1} = a_{i,j-1}/a_{j-1,j-1}$
    for $k = 1 : j-1$
      $a_{i,j} = a_{i,j} - \ell_{i,k}a_{k,j}$
  for $j = i+1 : n$
    for $k = 1 : i-1$
      $a_{i,j} = a_{i,j} - \ell_{i,k}a_{k,j}$



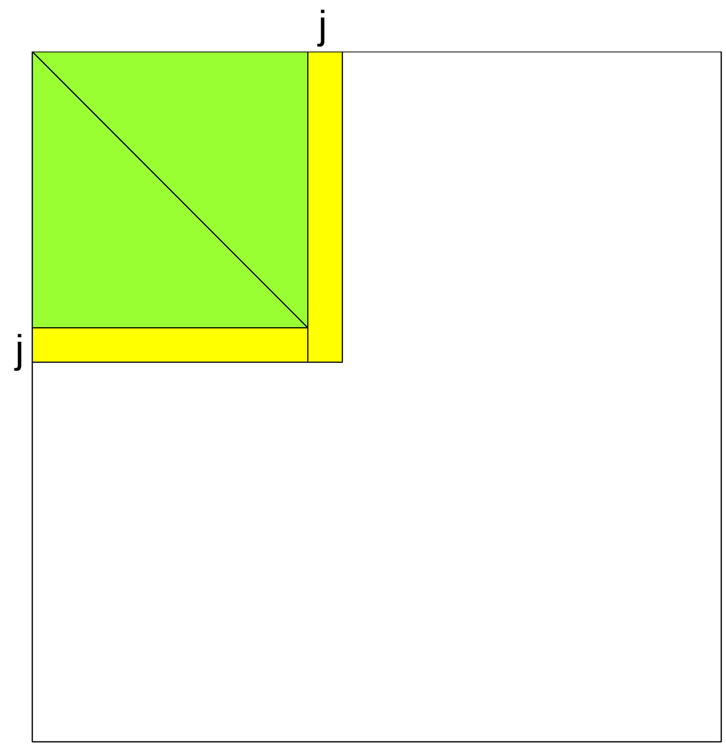
jik-Variante: (spaltenorientiert)

for $j = 2 : n$
  for $s = j : n$
    $\ell_{s,j-1} = a_{s,j-1}/a_{j-1,j-1}$
  for $i = 2 : j$
    for $k = 1 : i-1$
      $a_{i,j} = a_{i,j} - \ell_{i,k} a_{k,j}$
  for $i = j+1 : n$
    for $k = 1 : j-1$
      $a_{i,j} = a_{i,j} - \ell_{i,k} a_{k,j}$

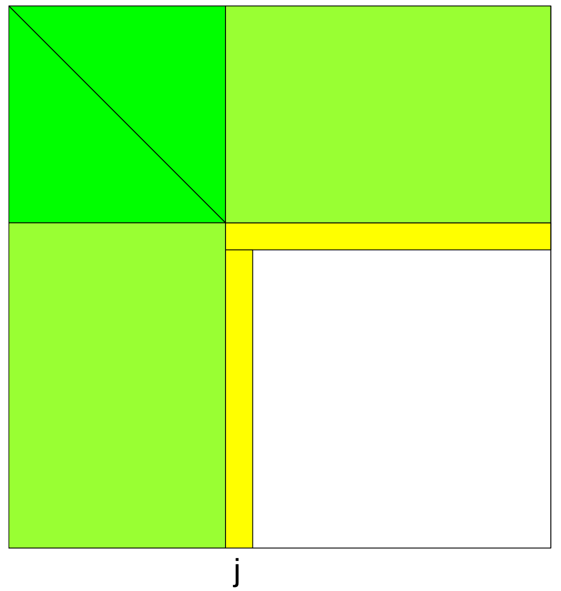


Bordering-Variante: (jki)

for $j = 2 : n$
  for $k = 1 : j - 2$
    for $i = k + 1 : j - 1$
      $a_{i,j} = a_{i,j} - \ell_{i,k} a_{k,j}$
  for $k = 1 : j - 1$
    $\ell_{j,k} = a_{j,k} / a_{k,k}$
    for $i = 1 : k + 1 : j$
      $a_{j,i} = a_{j,i} - \ell_{j,k} a_{k,i}$



Verfahren von Crout/Doolittle:



for $j = 1 : n$

$$a_{j:n,j} = a_{j:n,j} - \ell_{j:n,1:j-1} a_{1:j-1,j}$$
$$a_{j+1:n,j} = a_{j+1:n,j} / a_{j,j}$$
$$a_{j,j+1:n} = a_{j,j+1:n} - \ell_{j,1:j-1} a_{1:j-1,j+1:n}$$

Hier wird das Aufdatieren der Restmatrix (Schur-Komplement) so spät wie möglich ausgeführt.

3.4.4 Speicherhierarchien und die BLAS

Um auf Rechnern mit ausgeprägter Speicherhierarchie dennoch schnelle Programme zu erzielen, benutzen moderne Implementierungen als Bausteine die Basic Linear Algebra Subroutines (BLAS). Diese werden von Rechnerherstellern in für ihre Systeme optimierter Form zur Verfügung gestellt. Man unterscheidet drei Levels unterschiedlicher Effizienz; hier einige typische Vertreter

Operation (Level)	Definition	n_A	n_S	$q = n_A/n_S$
`saxpy` (1)	$\boldsymbol{y} := \alpha\boldsymbol{x} + \boldsymbol{y}$	$2n$	$3n+1$	$2/3$
Matrix-Vektor Produkt (2)	$\boldsymbol{y} := A\boldsymbol{x} + \boldsymbol{y}$	$2n^2$	n^2+3n	2
Matrix-Matrix Produkt (3)	$Y := Y + AX$	$2n^3$	$4n^2$	$n/2$

in der Tabelle: $\alpha \in \mathbb{R}$, $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, $A, X, Y \in \mathbb{R}^{n\times n}$

n_A : Anzahl arithmetischer Operationen

n_S : Anzahl Speicherzugriffe

Je größer der Quotient q, desto näher ist man am optimalen Durchsatz:

Bezeichnen t_A die Dauer einer (typischen) Gleitpunktoperation,
t_S die Dauer einer (typischen) Speicherzugriffs, sowie
t_G die Ausführungsdauer des Algorithmus,

so gilt

$$t_G = t_A \cdot n_A + t_S \cdot n_S = t_A n_A \left(1 + \frac{t_S n_S}{t_A n_A}\right) = t_A n_A \left(1 + \frac{1}{q} \frac{t_S}{t_A}\right)$$

Es ist somit effizient, Algorithmen der numerischen linearen Algebra möglichst aus BLAS2 bzw. BLAS3 Routinen aufzubauen.

Wir zeigen eine Möglichkeit, dies bei Gauß-Elimination durch Blockeinteilung zu erreichen. Wir benötigen hierzu eine Abwandlung der Gauß-Elimination für $m \times n$-Matrizen, $m \geq n$.

Algorithmus 3.1:

for $k = 1 : \min\{m-1, n\}$

$a_{k+1:m,k} = a_{k+1:m,k}/a_{k,k}$

if $k < n$ **then**

$a_{k+1:m,k+1:n} = a_{k+1:m.k+1:n} - \ell_{k+1:m,k}a_{k,k+1:n}$

Wir partitionieren die Matrizen nach $k-1$ Schritten von GE:

$$\begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \\ A_{3,1} & A_{3,2} & A_{3,3} \end{bmatrix} = \begin{bmatrix} L_{1,1} & O & O \\ L_{2,1} & I & O \\ L_{3,1} & O & I \end{bmatrix} \begin{bmatrix} R_{1,1} & R_{1,2} & R_{1,3} \\ O & \tilde{A}_{2,2} & \tilde{A}_{2,3} \\ O & \tilde{A}_{3,2} & \tilde{A}_{3,3} \end{bmatrix}$$

mit quadratischen Matrizen $A_{1,1}$, $A_{2,2}$, $A_{3,3}$ der Dimension $k-1$, n_B bzw. $n-(k-1)-n_B$ (n_B eine Blockgröße zwischen 1 und n).

Folgende Variante der GE ist reicher an BLAS3-Operationen (gezeigt ist ein Blockschritt):

1. Verwende Algorithmus 3.1 zur Faktorisierung von

$$\begin{bmatrix} \tilde{A}_{2,2} \\ \tilde{A}_{3,2} \end{bmatrix} = \begin{bmatrix} L_{2,2} \\ L_{3,2} \end{bmatrix} R_{2,2} \ .$$

2. Bestimme $R_{2,3} :=: L_{2,2}^{-1}\tilde{A}_{2,3}$ als Lösung eines unteren Dreieckssystems mit n_B rechten Seiten (BLAS3).

3. Setze $\tilde{\tilde{A}}_{3,3} := \tilde{A}_{3,3} - L_{3,2}R_{2,3}$ (BLAS3).

3.5 Pivotisierung

Das Eliminationsverfahren „bricht zusammen“ (bereits im ersten Schritt), wenn es auf die Matrix $A = \left[\begin{smallmatrix} 0 & 1 \\ 1 & 1 \end{smallmatrix}\right]$ angewandt wird (obwohl $\det(A) \neq 0$). A besitzt keine LR-Zerlegung.

Satz 3.12. *Es sei $A \in \mathbb{C}^{n\times n}$ regulär. Dann sind die folgenden Aussagen äquivalent:*

(a) A besitzt eine LR-Zerlegung mit einer normierten unteren Dreiecksmatrix L und einer regulären oberen Dreiecksmatrix R.

(b) Alle führenden Hauptuntermatrizen von A sind regulär.

Bemerkung. Besitzt die reguläre Matrix $A \in \mathbb{C}^{n\times n}$ eine LR-Zerlegung, so ist diese eindeutig bestimmt.

Satz 3.13 (Satz von Gerschgorin). *Für $A = [a_{i,j}] \in \mathbb{C}^{n\times n}$ sind die Gerschgorin-Kreisscheiben durch*

$$D_i := \left\{\zeta \in \mathbb{C} \,:\, |\zeta - a_{i,i}| \leq \sum_{\substack{j=1\\j\neq i}}^{n} |a_{i,j}|\right\} \qquad (i = 1, 2, \ldots, n)$$

definiert. Dann gilt

$$\Lambda(A) \subseteq \bigcup_{i=1}^{n} D_i.$$

Dabei ist $\Lambda(A) := \{\lambda \in \mathbb{C} \,:\, \det(A - \lambda I) = 0\}$ das Spektrum von A.

Korollar 3.14. *Ist $A = [a_{i,j}] \in \mathbb{C}^{n\times n}$ streng diagonaldominant, d.h. gilt*

$$|a_{i,i}| > \sum_{\substack{j=1\\j\neq i}}^{n} |a_{i,j}| \qquad \text{für alle } i = 1, 2, \ldots, n,$$

dann ist A regulär und besitzt eine LR-Zerlegung.

Ist π eine Permutation der Indexmenge $\{1, 2, \dots, n\}$, dann ist die zugehörige Permutationsmatrix $P = P_\pi = [p_{i,j}] \in \mathbb{R}^{n \times n}$ durch

$$p_{i,j} = \begin{cases} 1, & \text{falls } j = \pi(i), \\ 0, & \text{falls } j \neq \pi(i), \end{cases}$$

definiert.

- Es gilt $P_\pi^{-1} = P_{\pi^{-1}} = P_\pi^\top$.
- Die Menge aller Permutationsmatrizen ist eine Gruppe bezüglich der Matrizenmultiplikation.
- Die Determinante einer Permutationsmatrix hat entweder den Wert 1 oder den Wert -1.
- Ist $A = [a_{i,j}] \in \mathbb{C}^{n \times n}$ mit den Zeilen $z_1^\top, \dots, z_n^\top$ und Spalten $s_1, \dots, s_n$, so besitzt $P_\pi A$ die Zeilen $z_{\pi(1)}^\top, \dots, z_{\pi(n)}^\top$ und $AP_\pi^\top$ die Spalten $s_{\pi(1)}, \dots, s_{\pi(n)}$. Insbesondere ist $P_\pi A P_\pi^T = [a_{\pi(i),\pi(j)}]$.

Satz 3.15. *Es sei $A \in \mathbb{C}^{n\times n}$ invertierbar. Dann gibt es eine Permutationsmatrix $P \in \mathbb{R}^{n\times n}$, so dass PA eine LR-Zerlegung besitzt:*

$$PA = LR.$$

Dabei kann P so gewählt werden (Spaltenpivotsuche), dass die Einträge $\ell_{i,j}$ der Matrix L alle der Ungleichung $|\ell_{i,j}| \le 1$ genügen.

Die Lösung eines LGS $A\boldsymbol{x} = \boldsymbol{b}$ für beliebige reguläre Matrizen $A \in \mathbb{C}^{n\times n}$ mittels Gauß-Elimination besteht somit aus folgenden Schritten:

(a) Bestimme P, L, R, so dass $PA = LR$.
(b) Löse $L\boldsymbol{y} = P\boldsymbol{b}$ durch Vorwärtssubstitution.
(c) Löse $R\boldsymbol{x} = \boldsymbol{y}$ durch Rückwärtssubstitution.

(Insbesondere dann vorteilhaft, wenn $A\boldsymbol{x} = \boldsymbol{b}_i$ mit einer Matrix A und mehreren rechten Seiten $\boldsymbol{b}_i$ zu lösen ist.)

Pseudocode: (Gauß-Elimination mit Spaltenpivotsuche)

for $k = 1 : n-1$
  Bestimme p so, dass $|a_{p,k}| = \max_{k \le i \le n} |a_{i,k}|$
  Vertausche $a_{k,:}$ mit $a_{p,:}$
  Vertausche b_k mit b_p
  if $a_{k,k} = 0$ **then**
    stop
  else
    for $i = k+1 : n$
      $a_{i,k} := a_{i,k}/a_{k,k}$
      $b_i := b_i - a_{i,k} b_k$
      for $j = k+1 : n$
        $a_{i,j} := a_{i,j} - a_{i,k} a_{k,j}$

Spaltenpivotsuche muss (i.A.) immer durchgeführt werden, um kleine Pivotelemente zu vermeiden, die zu ungenauen Ergebnissen führen können:

Beispiel:

$$\begin{bmatrix} 10^{-20} & 1 \\ 1 & 1 \end{bmatrix} \boldsymbol{x} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \text{Lösung } \boldsymbol{x} = \frac{1}{1-10^{-20}} \begin{bmatrix} -1 \\ 1 \end{bmatrix} \approx \begin{bmatrix} -1 \\ 1 \end{bmatrix}.$$

Gauß-Elimination (mit Rundungseinheit $u = 2^{-53} \approx 10^{-16}$):

$$10^{-20} x_1 + x_2 = 1$$

$$(1-10^{20}) x_2 = -10^{20}, \text{ gerundet } -10^{20} \tilde{x}_2 = -10^{20}, \text{ d.h. } \tilde{x}_2 = 1, \ \tilde{x}_1 = 0.$$

(Gauß-Elimination mit Spaltenpivotsuche würde $\tilde{x}_1 = -1$ und $\tilde{x}_2 = 1$ liefern.)

Gauß-Elimination ohne Pivotsuche liefert also bei gerundeter Rechnung die Faktoren

$$\tilde{L} = \begin{bmatrix} 1 & 0 \\ 10^{20} & 1 \end{bmatrix} \qquad \text{und} \qquad \tilde{R} = \begin{bmatrix} 10^{-20} & 1 \\ 0 & -10^{20} \end{bmatrix}$$

anstelle der exakten Faktoren

$$L = \begin{bmatrix} 1 & 0 \\ 10^{20} & 1 \end{bmatrix} \qquad \text{und} \qquad R = \begin{bmatrix} 10^{-20} & 1 \\ 0 & 1 - 10^{20} \end{bmatrix}.$$

Beim Gleichungslösen wurde also verwendet

$$\tilde{L}\tilde{R} = \begin{bmatrix} 10^{-20} & 1 \\ 1 & 0 \end{bmatrix} \quad \text{statt} \quad LR = \begin{bmatrix} 10^{-20} & 1 \\ 1 & 1 \end{bmatrix} = A\,.$$

3.6 Equilibrierung und Nachiteration

Die Spaltenpivotsuche kann leicht ad absurdum geführt werden, wenn man die verschiedenen Gleichungen unterschiedlich wichtet:

Beispiel (aus 3.5, nur erste Gleichung wurde mit 10^{20} multipliziert)

$$\begin{bmatrix} 1 & 10^{20} \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 10^{20} \\ 0 \end{bmatrix}$$ führt bei Rechnung (mit Spaltenpivotsuche),$u \approx 10^{-16}$,

wie oben auf $\tilde{x}_2 = 1$ und $\tilde{x}_1 = 0$.

A heißt zeilenequilibriert, wenn in jeder Zeile die Betragssumme der Einträge gleich ist,

$$\sum_{j=1}^{n} |a_{i,j}| = 1 \quad (i = 1, 2, \ldots, n).$$

Man kann das leicht dadurch erreichen, indem man die i-te Zeile vorab mit $1/\sum_{j=1}^{n} |a_{i,j}|$ multipliziert.

Alternative: Bestimme Pivotelement (in Schritt j) so, dass

$$\frac{|a_{p,j}^{(j)}|}{\sum_{k=1}^{n}|a_{p,k}|} \geq \frac{|a_{i,j}^{(j)}|}{\sum_{k=1}^{n}|a_{i,k}|} \qquad \text{für alle } i = j, j+1, \ldots, n.$$

Man kann die Qualität einer berechneten Lösung $\tilde{\boldsymbol{x}}$ durch Nachiteration verbessern:

a) Berechne Residuum $\boldsymbol{r} = \boldsymbol{b} - A\tilde{\boldsymbol{x}}$ (doppelt genau).

b) Löse $A\boldsymbol{h} = \boldsymbol{r}$ (verwende LR-Zerlegung von A).

c) Setze $\boldsymbol{x} = \tilde{\boldsymbol{x}} + \boldsymbol{h}$.

3.7 Stabilität bei der Gauß-Elimination

Nicht verwechseln darf man:

- **Die Stabilität eines mathematischen Problems** (hier: eines linearen Gleichungssystems mit invertierbarer Koeffizientenmatrix), die beschreibt, wie sich die Lösung **bei exakter Rechnung** verändert, wenn die Daten gestört werden.

- **Die Stabilität eines numerischen Verfahrens** (hier: der Gauß-Elimination mit Spaltenpivotsuche), die beschreibt, wie sich die **in Gleitpunktarithmetik berechnete** Lösung von der exakten Lösung unterscheidet.

Beispiel: (Instabilität des Problems)

$$A = \begin{bmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 32 \\ 23 \\ 33 \\ 31 \end{bmatrix}, \qquad \tilde{\boldsymbol{b}} = \begin{bmatrix} 32.1 \\ 22.9 \\ 33.1 \\ 30.9 \end{bmatrix},$$

$$\boldsymbol{x} := A^{-1}\boldsymbol{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \qquad \tilde{\boldsymbol{x}} := A^{-1}\tilde{\boldsymbol{b}} = \begin{bmatrix} 9.2 \\ -12.6 \\ 4.5 \\ -1.1 \end{bmatrix}$$

Erklärung: $\operatorname{cond}_\infty(A) = 4.488 \cdot 10^3$, also

$$13.6 = \frac{\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|_\infty}{\|\boldsymbol{x}\|_\infty} \lessapprox \operatorname{cond}_\infty(A) \frac{\|\boldsymbol{b} - \tilde{\boldsymbol{b}}\|_\infty}{\|\boldsymbol{b}\|_\infty} = \left(4.488 \cdot 10^3\right) \cdot \frac{0.1}{33} = 13.6\,.$$

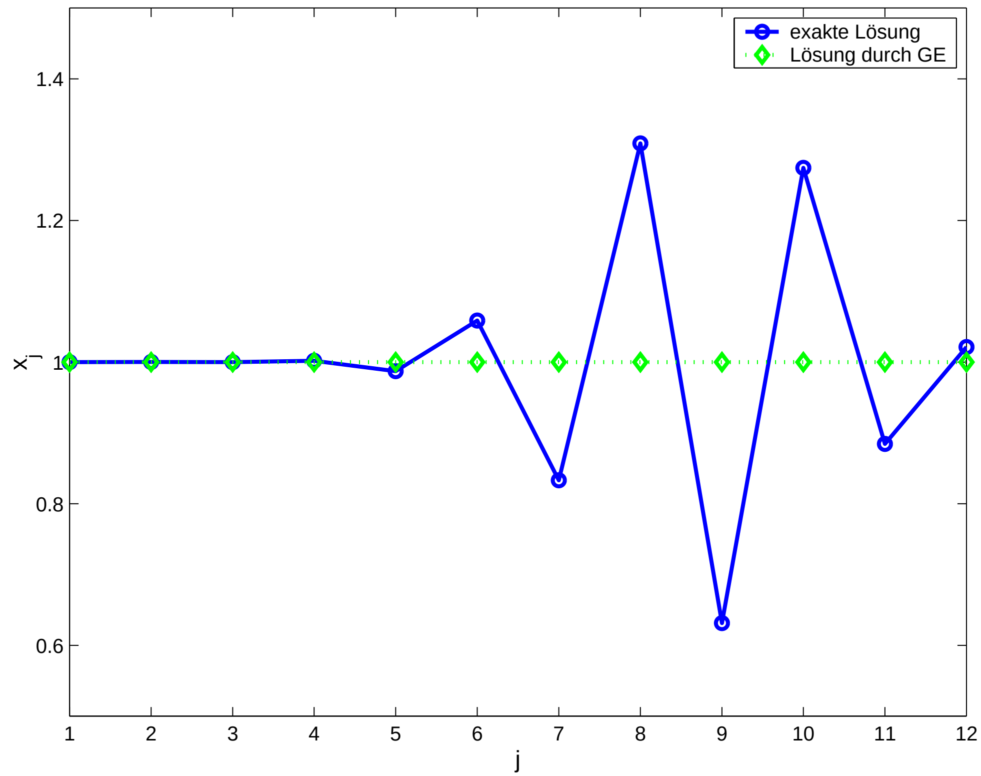
Ein weiteres Beispiel: Die Hilbert-Matrix der Dimension n

$$H_n = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{n+1} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \cdots & \frac{1}{n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \cdots & \frac{1}{2n-1} \end{bmatrix}$$

ist symmetrisch und positiv definit, insbesondere also invertierbar.
Ihre Konditionszahl $\text{cond}_2(H_n)$ wächst wie $e^{7n/2} \approx 33^n$:

n	4	6	8	10	12	14
$\text{cond}_2(H_n)$	$1.6 \cdot 10^{4}$	$1.5 \cdot 10^{7}$	$1.5 \cdot 10^{10}$	$1.6 \cdot 10^{13}$	$1.7 \cdot 10^{16}$	$1.9 \cdot 10^{19}$
$\text{cond}_\infty(H_n)$	$2.8 \cdot 10^{4}$	$2.9 \cdot 10^{7}$	$3.4 \cdot 10^{10}$	$3.5 \cdot 10^{13}$	$4.1 \cdot 10^{16}$	$4.5 \cdot 10^{19}$

Wir lösen $H_{12}\boldsymbol{x} = \boldsymbol{b}$ mit Rundungseinheit $u = 2^{-53} \approx 1.1 \cdot 10^{-16}$ (die rechte Seite $\boldsymbol{b}$ wurde so gewählt, dass $A^{-1}\boldsymbol{b} = [1, \ldots, 1]^\top$ gilt).



Bezeichnungen. Für $A = [a_{i,j}], B = [b_{i,j}] \in \mathbb{R}^{n \times n}$ bedeute $A \leq B$, dass $a_{i,j} \leq b_{i,j}$ für alle $1 \leq i, j \leq n$ erfüllt ist. Außerdem sei $|A| := [|a_{i,j}|] \in \mathbb{R}^{n \times n}$.

Satz 3.16 (Wilkinson, 1963)**.** *Es seien $A \in \mathbb{C}^{n \times n}$ regulär und $\boldsymbol{b} \in \mathbb{C}^n$. In Gleitpunktarithmetik (mit Rundungseinheit u) werde (mit beliebiger Pivotstrategie) eine LR-Zerlegung $\tilde{L}\tilde{R} \sim PA$ von A sowie eine Lösung $\tilde{\boldsymbol{x}}$ des Systems $A\boldsymbol{x} = \boldsymbol{b}$ berechnet. Dann gelten:*

(a) $\tilde{L}\tilde{R} = PA + E$ *mit* $|E| \leq n\,u\,|\tilde{L}|\,|\tilde{R}| + O\left(u^2\right)$.
(b) $(A + \Delta A)\tilde{\boldsymbol{x}} = \boldsymbol{b}$ *mit* $|\Delta A| \leq 3\,n\,u\,|\tilde{L}|\,|\tilde{R}| + O\left(u^2\right)$.

Bemerkung 3.17.

(a) Für absolute Normen $\|\cdot\|$ gelten somit

$$\|E\| \leq nu\|\tilde{L}\|\|\tilde{R}\| + O(u^2) \quad \textit{sowie} \quad \|\Delta A\| \leq 3nu\|\tilde{L}\|\tilde{R}\| + O(u^2).$$

(b) Die Konstante $3\gamma_n + \gamma_n^2$ im Beweis von Satz 3.16 lässt sich verbessern zu $2\gamma_n$.

Korollar 3.18. *Bei Spaltenpivotsuche gilt mit den Bezeichnungen aus Satz 3.16*

$$\|\Delta A\|_\infty \leq 3\,\rho\, n^3\, \|A\|_\infty\, u + O(u^2),$$

wobei

$$\rho := \frac{\max_{1\leq i,j\leq n} |\tilde{r}_{i,j}|}{\max_{1\leq i,j\leq n} |a_{i,j}|}$$

der sog. Wachstumsfaktor bei Gauß-Elimination mit Spaltenpivotsuche ist. Es gilt

$$\rho \leq 2^{n-1}$$

und diese obere Schranke wird auch angenommen.

Bemerkung 3.19. *Mit Hife von Bemerkung 3.17a lässt sich zeigen, dass sogar* $\|\Delta A\|_\infty \leq 2n^2\gamma_n\rho\|A\|_\infty$.

Allerdings beobachtet man in der Praxis keineswegs, dass ρ wie 2^{n-1} wächst, sondern eher ein Anwachsen wie $n^{2/3}$.

Rückwärtsanalyse (interpretiere Rundefehler als Datenfehler)

$$(A + \Delta A)\tilde{\boldsymbol{x}} = \boldsymbol{b} \;\text{ mit }\; \|\Delta A\|_\infty \lessapprox 3\,\rho\,n^3\,\|A\|_\infty u$$

und Konditionsanalyse (wie wirken sich Datenfehler in *exakter Arithmetik* auf das Ergebnis aus)

$$\frac{\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|}{\|\boldsymbol{x}\|} \lessapprox \text{cond}\,(A)\left[\frac{\|A - \tilde{A}\|}{\|A\|} + \frac{\|\boldsymbol{b} - \tilde{\boldsymbol{b}}\|}{\|\boldsymbol{b}\|}\right] \;\text{ mit }\; \text{cond}(A) := \|A\|\,\|A^{-1}\|$$

liefern insgesamt

$$\frac{\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|_\infty}{\|\boldsymbol{x}\|_\infty} \lessapprox 3\,n^3\,\rho\;\text{cond}_\infty(A)\,u.$$

Gauß-Elimination mit Spaltenpivotsuche ist stabil, wenn Wachstumsfaktor ρ und Konditionszahl $\text{cond}(A)$ klein sind.

Wir haben gesehen, dass

$$\rho \leq 2^{n-1} \text{ für alle } A \in \mathbb{R}^{n \times n}$$

gilt und dass diese Schranke angenommen wird. In der ‘Praxis’ wächst ρ aber wesentlich langsamer mit n, so dass Gauß-Elimination mit Spaltenpivotsuche praktisch rückwärtsstabil ist.

Ob dieser Algorithmus auch vorwärtsstabil ist, hängt von A bzw. von $\text{cond}(A)$ ab. Wie wir am Beispiel der Hilbert-Matrix gesehen haben, gibt es invertierbare Matrizen A, die extrem schlecht konditioniert sind, aber auf den ersten Blick ganz unverdächtig aussehen.

Vollständige Pivotsuche (complete pivoting): Hier zeigte Wilkinson 1961 für den Wachstumsfaktor $\rho = \rho_{CP}(n)$

$$\rho^{CP}(n) \leq \sqrt{n \cdot 2 \cdot 3^{1/2} \cdots n^{1/(n-1)}} \sim cn^{1/2}\, n^{\frac{1}{4}\log n},$$

und dass diese Schranke nicht scharf ist. Diese Schranke wächst wesentlich langsamer als 2^{n-1}, kann aber auch beträchtlich werden: sie beträgt z.B. 3570 für $n = 100$. Bekannte Teilergebnisse:

- $\rho^{CP}(2) = 2$
- $\rho^{CP}(3) = 9/4$
- $\rho^{CP}(4) = 4$
- $\rho^{CP}(5) < 5.005$

Die lang anhaltende Vermutung $\rho^{CP}(n) \leq n$ wurde 1991 von Gould durch ein (13×13) Gegenbeispiel widerlegt.

Die Lücke zwischen dem theoretisch schlimmsten Fall und dem in der Praxis beobachteten Wachstumsverhalten von ρ (beide Pivotstrategien) ist nach wie vor verblüffend:

> “It is rare for the growth factor to exceed 10 on ‘real’ problems, regardless of the size of n.”
>
> P. Gill, W. Murray & M. Wright,
> *Numerical Linear Algebra and Optimization* (1991)

> “In practice g_{PP} is always n or less.”
>
> J. Demmel, *Applied Numerical Linear Algebra* (1997)

> “The growth suggested by the bound rarely occurs in practice. The reasons are not well understood.”
>
> G. W. Stewart, *Matrix Algorithms I* (1998)

> “Anyone that unlucky has already been hit by a truck.”
>
> wird J. Wilkinson zugeschrieben