Kapitel 4

Lineare Gleichungssysteme

4.1 Problemstellung und Einführung

In diesem Kapitel betrachten wir direkte Verfahren zur Lösung von linearen Gleichungssystemen.

Lineares Gleichungssystem: Gesucht ist eine Lösung $x \in \mathbb{R}^n$ von

(4.1) $$Ax = b.$$

mit
(4.2)
$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \in \mathbb{R}^{n,n}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \in \mathbb{R}^n, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n.$$

Die hier besprochenen direkten Methoden liefern– rundungsfehlerfreie Rechnung vorausgesetzt– die Lösung von (4.1) in endlich vielen Rechenschritten. Bekanntlich ist (4.1) die Matrixschreibweise für

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n = b_i, \quad i = 1, \ldots, n.$$

Lineare Gleichungssysteme treten in der Praxis als Hilfsproblem bei einer Vielzahl von Problemstellungen auf, z.B. bei der Lösung von Rand- und Randanfangswertaufgaben für gewöhnliche und partielle Differentialgleichungen (Schaltkreissimulation, elektromagnetische Felder, ...), in der Bildverarbeitung, usw. . Schätzungen besagen, dass etwa 75% der Rechenzeit im technisch-wissenschaftlichen Bereich auf die Lösung von linearen Gleichungssystemen entfällt.

Wir erinnern zunächst an folgenden Sachverhalt.

Proposition 4.1.1 *Das lineare Gleichungssystem* (4.1) *hat eine Lösung genau dann, wenn gilt*

$$\textit{rang}(A) = \textit{rang}(A, b).$$

Hierbei ist bekanntlich für eine Matrix $B \in \mathbb{R}^{n,m}$ *der Rang definiert durch*

$$\begin{aligned}\mathrm{Rang}(B) &= \textit{Maximalzahl } r \textit{ der linear unabhängigen Zeilenvektoren}\\ &= \textit{Maximalzahl } r \textit{ der linear unabhängigen Spaltenvektoren}.\end{aligned}$$

Das lineare Gleichungssystem (4.1) *hat eine eindeutige Lösung genau dann, wenn* A *invertierbar ist (oder gleichbedeutend:* $\det(A) \neq 0$ *). Die eindeutige Lösung lautet dann*

$$x = A^{-1}b.$$

4.2 Das Gaußsche Eliminationsverfahren, Dreieckszerlegung einer Matrix

Das grundsätzliche Vorgehen der Gauß-Elimination ist aus der Linearen Algebra bekannt. Wir werden das Verfahren kurz wiederholen und zeigen, wie man daraus eine Dreieckszerlegung einer Matrix erhält. Zudem werden wir uns klarmachen, welchen Einfluss Rundungsfehler haben können und wie dieser Einfluss wirksam bekämpft werden kann.

Die Grundidee des Gaußschen Eliminationsverfahrens besteht darin, das Gleichungssystem (4.1) durch die elementaren Operationen

- Addition eines Vielfachen einer Gleichung zu einer anderen,
- Zeilenvertauschungen, d.h. Vertauschen von Gleichungen
- Spaltenvertauschungen, die einer Umnummerierung der Unbekannten entsprechen,

in ein Gleichungssystem der Form

$$Ry = c, \quad y_{\sigma_i} = x_i, \quad i = 1, \ldots, n,$$

mit der durchgeführten Spaltenpermutation $(\sigma_1, \ldots, \sigma_n)$ und einer oberen Dreiecksmatrix

$$R = \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ 0 & & r_{nn} \end{pmatrix}$$

zu überführen, das dieselben Lösungen wie (4.1) besitzt. (4.3) ist ein sogenanntes *gestaffeltes Gleichungssystem*, das man leicht durch Rückwärtssubstitution lösen kann, solange R invertierbar ist. Werden keine Spaltenvertauschungen durchgeführt, dann gilt $x = y$.

4.2.1 Lösung gestaffelter Gleichungssysteme

Gestaffelte Gleichungssysteme

$$Ry = c \tag{4.3}$$

mit einer oberen Dreiecksmatrix

$$R = \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ 0 & & r_{nn} \end{pmatrix}, \tag{4.4}$$

sowie

$$Lz = d \tag{4.5}$$

mit einer unteren Dreiecksmatrix

$$L = \begin{pmatrix} l_{11} & & 0 \\ \vdots & \ddots & \\ l_{n1} & \cdots & l_{nn} \end{pmatrix},$$

lassen sich offensichtlich leicht durch Rückwärts- bzw. Vorwärtssubstitution lösen:

Satz 4.2.1 *Seien $R = (r_{ij}) \in \mathbb{R}^{n,n}$ und $L = (l_{ij}) \in \mathbb{R}^{n,n}$ invertierbare obere bzw. untere Dreiecksmatrizen und $c = (c_1, \ldots, c_n)^T$, $d = (d_1, \ldots, d_n)^T$ Spaltenvektoren. Dann lassen sich die Lösungen von* (4.3) *bzw.* (4.5) *folgendermaßen berechnen:*

a) **Rückwärtssubstitution für obere Dreieckssysteme** (4.3)**:**

$$\boxed{y_i = \frac{c_i - \sum_{j=i+1}^{n} r_{ij} y_j}{r_{ii}}, \quad i = n, n-1, \ldots, 1.}$$

b) **Vorwärtssubstitution für untere Dreieckssysteme** (4.5)**:**

$$\boxed{z_i = \frac{d_i - \sum_{j=1}^{i-1} l_{ij} z_j}{l_{ii}}, \quad i = 1, 2, \ldots, n.}$$

Beweis: zu a): Da R invertierbar ist, gilt

$$\det(R) = r_{11} r_{22} \cdots r_{nn} \neq 0,$$

also $r_{ii} \neq 0$, $i = 1, \ldots, n$. Somit ergibt sich

$$\begin{aligned} y_n &= \frac{c_n}{r_{nn}} \\ y_{n-1} &= \frac{c_{n-1} - r_{n-1,n} y_n}{r_{n-1,n-1}} \\ &\vdots \end{aligned}$$

und somit induktiv a).

zu b): Wegen $\det(L) = l_{11}l_{22}\cdots l_{nn} \neq 0$ gilt $l_{ii} \neq 0$, $i = 1, \ldots, n$. Somit ergibt sich

$$\begin{aligned} z_1 &= \frac{d_1}{l_{11}} \\ z_2 &= \frac{d_2 - l_{2,1}z_1}{l_{22}} \\ &\vdots \end{aligned}$$

und wir erhalten induktiv b). $\Box$

Bemerkung: Der Aufwand für die Rückwärtssubstitution ist $O(n^2)$ an elementaren Rechenoperationen, falls nicht zusätzlich eine spezielle Besetztheitsstruktur vorliegt (Dünnbesetztheit, Bandstruktur). $\Box$

4.2.2 Das Gaußsche Eliminationsverfahrens

Wir erklären nun (die grundsätzliche Vorgehensweise sollte aus der Linearen Algebra bekannt sein), wie man mit dem Gaußschen Eliminationsverfahren ein gestaffeltes Gleichungssystem erhält. Statt mit den Gleichungen (4.1) zu arbeiten, ist es bequemer, die Operationen an der um die rechte Seite erweiterten Koeffizientenmatrix

$$(A, b) = \left(\begin{array}{ccc|c} a_{11} & \cdots & a_{1n} & b_1 \\ \vdots & & \vdots & \vdots \\ a_{n1} & \cdots & a_{nn} & b_n \end{array}\right)$$

durchzuführen.

Beim Gaußschen Eliminationsverfahren geht man nun wie folgt vor:

Grundkonzept des Gaußschen Eliminationsverfahrens:

0. **Initialisierung:** $(A^{(1)}, b^{(1)}) = \left(\begin{array}{ccc|c} a_{11}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ \vdots & & \vdots & \vdots \\ a_{n1}^{(1)} & \cdots & a_{nn}^{(1)} & b_n^{(1)} \end{array}\right) := (A, b).$

1. **Pivotsuche:** Suche eine Gleichung r, die von x_1 abhängt, also mit $a_{r1}^{(1)} \neq 0$ und

vertausche sie mit der ersten Gleichung:

$$(A^{(1)}, b^{(1)}) = \left(\begin{array}{ccc|c} a_{11}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ \vdots & & \vdots & \vdots \\ a_{r1}^{(1)} & \cdots & a_{rn}^{(1)} & b_r^{(1)} \\ \vdots & & \vdots & \vdots \\ a_{n1}^{(1)} & \cdots & a_{nn}^{(1)} & b_n^{(1)} \end{array}\right) \rightsquigarrow \left(\begin{array}{ccc|c} a_{r1}^{(1)} & \cdots & a_{rn}^{(1)} & b_r^{(1)} \\ \vdots & & \vdots & \vdots \\ a_{11}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ \vdots & & \vdots & \vdots \\ a_{n1}^{(1)} & \cdots & a_{nn}^{(1)} & b_n^{(1)} \end{array}\right)$$

$$=: \left(\begin{array}{ccc|c} \tilde{a}_{11}^{(1)} & \cdots & \tilde{a}_{1n}^{(1)} & \tilde{b}_1^{(1)} \\ \vdots & & \vdots & \vdots \\ \tilde{a}_{n1}^{(1)} & \cdots & \tilde{a}_{nn}^{(1)} & \tilde{b}_n^{(1)} \end{array}\right) = (\tilde{A}^{(1)}, \tilde{b}^{(1)}).$$

Ist A invertierbar, dann existiert immer ein solches r, da wegen der Invertierbarkeit von A die erste Spalte nicht verschwinden kann.

2. **Elimination:** Subtrahiere geeignete Vielfache der ersten Gleichung von den übrigen Gleichungen derart, dass die Koeffizienten von x_1 in diesen Gleichungen verschwinden. Offensichtlich muss man hierzu jeweils das l_{i1}-fache mit

$$l_{i1} = \frac{\tilde{a}_{i1}^{(1)}}{\tilde{a}_{11}^{(1)}}$$

der ersten Gleichung von der i-ten Gleichung subtrahieren:

$$(\tilde{A}^{(1)}, \tilde{b}^{(1)}) \quad \rightsquigarrow \quad (A^{(2)}, b^{(2)}) = \left(\begin{array}{cccc|c} \tilde{a}_{11}^{(1)} & \tilde{a}_{12}^{(1)} & \cdots & \tilde{a}_{1n}^{(1)} & \tilde{b}_1^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} & b_n^{(2)} \end{array}\right)$$

$$=: \left(\begin{array}{ccc|c} \tilde{a}_{11}^{(1)} & \cdots & \tilde{a}_{1n}^{(1)} & \tilde{b}_1^{(1)} \\ \hline 0 & & & \\ \vdots & & \hat{A}^{(2)} & \hat{b}^{(2)} \\ 0 & & & \end{array}\right).$$

3. **Iteration:** Wende für $k = 2, \dots, n-1$ Schritt 1. und 2. auf $(\hat{A}^{(k)}, \hat{b}^{(k)})$ an:

   1_k. Wähle ein Pivotelement $a_{rk}^{(k)} \neq 0$, $k \leq r \leq n$, vertausche Zeile k und r $\rightsquigarrow$ $(\tilde{A}^{(k)}, \tilde{b}^{(k)})$

   2_k. Subtrahiere das l_{ik}-fache mit

   $$l_{ik} = \frac{\tilde{a}_{ik}^{(k)}}{\tilde{a}_{kk}^{(k)}}$$

   der k-ten Gleichung von der i-ten Gleichung, $i = k+1, \dots, n$.
   $\rightsquigarrow$ $(A^{(k+1)}, b^{(k+1)})$

Nach k Eliminationsschritten

$$(A,b) =: (A^{(1)}, b^{(1)}) \to (A^{(2)}, b^{(2)}) \to \ldots \to (A^{(k+1)}, b^{(k+1)})$$

erhalten wir also eine Zwischenmatrix der Form

$$(A^{(k+1)}, b^{(k+1)}) = \left(\begin{array}{ccc|ccc|c}
\tilde{a}_{11}^{(1)} & \cdots & \tilde{a}_{1k}^{(1)} & \cdots & & \tilde{a}_{1n}^{(1)} & \tilde{b}_1^{(1)} \\
0 & \ddots & & & & \vdots & \vdots \\
 & & \tilde{a}_{kk}^{(k)} & \cdots & & \tilde{a}_{kn}^{(k)} & \tilde{b}_k^{(k)} \\
\hline
 & & 0 & & & & \\
 & & \vdots & \hat{A}^{(k+1)} & & & \hat{b}^{(k+1)} \\
 & & 0 & & & &
\end{array}\right).$$

Nach $n-1$ Eliminationsschritten liegt somit ein gestaffeltes Gleichungssystem (4.3)

$$Rx = c, \quad \text{mit} \quad R = A^{(n)}, \quad c = b^{(n)}$$

vor.

Pivotstrategie

Das Element $a_{rk}^{(k)}$, das in Schritt 1_k bestimmt wird, heißt *Pivotelement*. Theoretisch kann man bei der Pivotsuche jedes $a_{rk}^{(k)} \neq 0$ als Pivotelement wählen. Die Wahl kleiner Pivotelemente kann aber zu einer dramatischen Verstärkung von Rundungsfehlern führen. Gewöhnlich trifft man daher die Wahl von $a_{rk}^{(k)}$ durch

Spaltenpivotsuche: Wähle $k \leq r \leq n$ mit $|a_{rk}^{(k)}| = \max_{k \leq i \leq n} |a_{ik}^{(k)}|$.

Hierbei sollten die Zeilen von A "equilibriert" sein, also ihre Normen dieselbe Größenordnung haben.

Beispiel 4.2.1 Betrachte das Beispiel

$$\begin{pmatrix} 1 & 2 & -1 \\ 2 & -2 & 4 \\ 2 & 1 & -2 \end{pmatrix} x = \begin{pmatrix} 2 \\ 10 \\ -2 \end{pmatrix}$$

Dies liefert

$$\left(\begin{array}{rrr|r} 1 & 2 & -1 & 2 \\ 2 & -2 & 4 & 10 \\ 2 & 1 & -2 & -2 \end{array}\right) \quad \overset{\text{Spaltenpivotsuche}}{\longrightarrow} \quad \left(\begin{array}{rrr|r} 2 & -2 & 4 & 10 \\ 1 & 2 & -1 & 2 \\ 2 & 1 & -2 & -2 \end{array}\right)$$

$$\overset{\text{Elimination}}{\longrightarrow} \quad \begin{array}{l} -(\underbrace{1/2}_{=l_{21}}) \cdot \text{Zeile } 1 \\ -(\underbrace{1}_{=l_{31}}) \cdot \text{Zeile } 1 \end{array} \left(\begin{array}{rrr|r} 2 & -2 & 4 & 10 \\ 0 & 3 & -3 & -3 \\ 0 & 3 & -6 & -12 \end{array}\right)$$

$$\overset{\text{Spaltenpivotsuche+Elimination}}{\longrightarrow} \quad -\underbrace{1}_{=l_{32}} \cdot \text{Zeile } 2 \left(\begin{array}{rrr|r} 2 & -2 & 4 & 10 \\ 0 & 3 & -3 & -3 \\ 0 & 0 & -3 & -9 \end{array}\right)$$

4.2.3 Praktische Implementierung des Gauß-Verfahrens

Bei der Realisierung auf einem Rechner speichert man in der Regel auch die verwendeten Multiplikatoren l_{ik}. Wir werden sehen, dass das Gaußsche Eliminationsverfahren dann "gratis" eine Dreieckszerlegung (oder LR-Zerlegung) von A der Form

$$LR = PA \tag{4.6}$$

liefert. Hierbei ist $R \in \mathbb{R}^{n,n}$ eine obere Dreiecksmatrix (4.4), $L \in \mathbb{R}^{n,n}$ eine untere Dreiecksmatrix der Form

$$L = \begin{pmatrix} 1 & & 0 & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ \vdots & & \ddots & \ddots & \\ l_{n1} & & \cdots & l_{n,n-1} & 1 \end{pmatrix}, \tag{4.7}$$

und P eine *Permutationsmatrix*, die lediglich die Zeilen von A permutiert.

Wir erhalten die folgende Implementierung des Gauß-Verfahrens mit Spaltenpivotsuche:

Algorithmus 1 Gaußsches Eliminationsverfahren mit Spaltenpivotsuche

Setze $(A^{(1)}, b^{(1)}) = (A, b)$ *und* $L^{(1)} = 0 \in \mathbb{R}^{n,n}$.

Für $k = 1, 2, \ldots, n-1$*:*

1. **Spaltenpivotsuche:** *Bestimme* $k \leq r \leq n$ *mit*

$$|a_{rk}^{(k)}| = \max_{k \leq i \leq n} |a_{ik}^{(k)}|.$$

Falls $a_{rk}^{(k)} = 0$: STOP, A ist singulär.

Vertausche die Zeilen r und k von $(A^{(k)}, b^{(k)})$ und von $L^{(k)}$. Das Ergebnis sei formal mit $(\tilde{A}^{(k)}, \tilde{b}^{(k)})$, $\tilde{L}^{(k)}$ bezeichnet.

2. **Elimination:** *Subtrahiere für $i = k+1, \ldots, n$ das l_{ik}-fache, $l_{ik} = \frac{\tilde{a}_{ik}^{(k)}}{\tilde{a}_{kk}^{(k)}}$, der k-ten Zeile von $(\tilde{A}^{(k)}, \tilde{b}^{(k)})$ von der i-ten Zeile und füge die Multiplikatoren l_{ik} in $\tilde{L}^{(k)}$ ein. Das Ergebnis sei formal mit $(A^{(k+1)}, b^{(k+1)})$ und $L^{(k+1)}$ bezeichnet.*

   Im Detail: *Initialisiere $(A^{(k+1)}, b^{(k+1)}) := (\tilde{A}^{(k)}, \tilde{b}^{(k)})$, $\quad L^{(k+1)} := \tilde{L}^{(k)}$.*

   Für $i = k+1, \ldots, n$;

$$\begin{aligned} l_{ik} &= \frac{\tilde{a}_{ik}^{(k)}}{\tilde{a}_{kk}^{(k)}}, \\ b_i^{(k+1)} &= \tilde{b}_i^{(k)} - l_{ik}\tilde{b}_k^{(k)}, \\ a_{ik}^{(k+1)} &= 0, \\ l_{ik}^{(k+1)} &= l_{ik} \quad \textit{(Multiplikator speichern).} \end{aligned}$$

   Für $j = k+1, \ldots, n$:

$$a_{ij}^{(k+1)} = \tilde{a}_{ij}^{(k)} - l_{ik}\tilde{a}_{kj}^{(k)}$$

Ergebnis: *$R := A^{(n)}$, $c := b^{(n)}$, $L := I + L^{(n)}$ mit der Einheitsmatrix $I \in \mathbb{R}^{n,n}$.*

Bei einer Implementierung auf dem Rechner kann man für die Speicherung aller $A^{(k)}$, $b^{(k)}$, $\tilde{A}^{(k)}$, $\tilde{b}^{(k)}$ die Felder verwenden, in denen A und b gespeichert waren. $L^{(k)}$ kann man anstelle der enstehenden Nullen im strikten unteren Dreieck platzsparend speichern.

Bemerkung: Der Rechenaufwand ist $n^3/3 - n/3$ an elementaren Rechenoperationen, falls nicht zusätzlich eine spezielle Besetztheitsstruktur vorliegt. □

Vollständige Pivotsuche

Anstelle der Spaltenpivotsuche kann man auch *vollständige Pivotsuche* durchführen, bei der man die Pivotsuche nicht auf die erste Spalte beschränkt. Schritt 1 in Algorithmus 1 ist dann wie folgt zu modifizieren:

Algorithmus 2 Gaußsches Eliminationsverfahren mit vollständiger Pivotsuche *Algorithmus 1 mit folgender Modifikation von Schritt 1:*

1.′ **Vollständige Pivotsuche:** *Bestimme $k \leq r \leq n$, $k \leq s \leq n$ mit*

$$|a_{rs}^{(k)}| = \max_{k \leq i,j \leq n} |a_{ij}^{(k)}|.$$

Falls $a_{rs}^{(k)} = 0$*: STOP,* A *ist singulär.*

Vertausche die Zeilen r *und* k *sowie die Spalten* s *und* k *von* $(A^{(k)}, b^{(k)})$ *und von* $L^{(k)}$*. Das Ergebnis sei formal mit* $(\tilde{A}^{(k)}, \tilde{b}^{(k)})$*,* $\tilde{L}^{(k)}$ *bezeichnet.*

Achtung: Bei jeder Spaltenvertauschung müssen die Komponenten von x entsprechend umnumeriert werden, d.h. nach Lösen von (4.3) müssen die Komponenten des Ergebnisvektors x zurückgetauscht werden.

In der Regel wird vollständige Pivotsuche nur bei "fast singulären" Matrizen angewandt, um den Rundungsfehlerinfluß minimal zu halten. □

4.2.4 Gewinnung einer Dreieckszerlegung

Wir zeigen nun, dass das Gaußsche Eliminationsverfahren mit Spaltenpivotsuche (Algorithmus 1) eine Dreieckszerlegung (oder LR-Zerlegung) von A der Form

$$LR = PA \tag{4.6}$$

liefert. Hierbei ist $R \in \mathbb{R}^{n,n}$ bzw. $L \in \mathbb{R}^{n,n}$ die von Algorithmus 1 gelieferte obere Dreiecksmatrix der Form (4.4) bzw. untere Dreiecksmatrix der Form (4.7) und P ist eine ***Permutationsmatrix*** für die durchgeführten Zeilenvertauschungen.

Das Gaußsche Eliminationsverfahren mit vollständiger Pivotsuche (Algorithmus 2) liefert eine Dreieckszerlegung

$$LR = PAQ, \tag{4.8}$$

wobei P bzw. Q Permutationsmatrizen für die durchgeführten Zeilen- bzw. Spaltenvertauschungen sind.

Wir betrachten im Folgenden nur die Spaltenpivotsuche genauer, die vollständige Pivotsuche kann ähnlich behandelt werden.

Bemerkung: Eine Dreieckszerlegung (4.6) oder (4.8) ist sehr nützlich, wenn man (4.1) für mehrere rechte Seiten lösen will. Tatsächlich gilt

$$Ax = b \quad \Longleftrightarrow \quad PAQy = Pb,\ \ x = Qy \quad \Longleftrightarrow \quad L\underbrace{Ry}_{=:z} = Pb,\ \ x = Qy,$$

wobei $Q = I$ bei Spaltenpivotsuche. Man erhält nun x durch folgende Schritte:

Vorwärts-Rückwärtssubstitution für Dreieckszerlegung:

Löse $Lz = Pb$ nach z durch Vorwärtssubstitution gemäß Satz 4.2.1.

Löse $Ry = z$ nach y durch Rückwärtssubstitution gemäß Satz 4.2.1.

Lösung: $x = Qy$.

Liegt also die Dreieckszerlegung vor, dann kann (4.1) für jede rechte Seite in $O(n^2)$ Operationen berechnet werden. □

Wir überlegen uns nun, dass das Gaußsche Eliminationsverfahren mit Spaltenpivotsuche eine Dreieckszerlegung (4.6) liefert. Die vollständige Pivotsuche kann analog untersucht werden.

Matrixdarstellung der Eliminationsschritte

Wir betrachten das Gaußsche Eliminationsverfahren mit Spaltenpivotsuche (Algorithmus 1). Formal läßt sich der Übergang $(A^{(k)}, b^{(k)}) \to (\tilde{A}^{(k)}, \tilde{b}^{(k)}) \to (A^{(k+1)}, b^{(k+1)})$ durch Multiplikation mit Matrizen darstellen. Tatsächlich gilt

$$\begin{aligned} (\tilde{A}^{(k)}, \tilde{b}^{(k)}) &= P_k(A^{(k)}, b^{(k)}) \quad (\mathit{Zeilenvertauschung}) \\ (A^{(k+1)}, b^{(k+1)}) = L_k(\tilde{A}^{(k)}, \tilde{b}^{(k)}) &= L_k P_k (A^{(k)}, b^{(k)}) \quad (\mathit{Elimination}) \end{aligned}$$

mit der *elementaren Permutationsmatrix* (vertausche Zeile k und r der Einheitsmatrix)

$$(4.9) \qquad \begin{array}{r} \\ \\ \\ k \rightarrow \\ P_k = \\ \\ \\ r \rightarrow \\ \\ \\ \end{array} \begin{pmatrix} 1 & & & & & & & & & \\ & \ddots & & & & & & & & \\ & & 1 & & & & & & & \\ & & & 0 & & & & 1 & & \\ & & & & 1 & & & & & \\ & & & & & \ddots & & & & \\ & & & & & & 1 & & & \\ & & & 1 & & & & 0 & & \\ & & & & & & & & 1 & \\ & & & & & & & & & \ddots \end{pmatrix}$$

und der *elementaren Eliminationsmatrix*

$$(4.10) \qquad L_k = \begin{pmatrix} 1 & & & 0 & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & -l_{k+1,k} & 1 & & \\ 0 & & \vdots & & \ddots & \\ & & -l_{nk} & & 0 & 1 \end{pmatrix}$$

Man prüft leicht, dass P_k, L_k invertierbar sind mit

$$(4.11) \qquad P_k^{-1} = P_k, \quad L_k^{-1} = \begin{pmatrix} 1 & & & & & 0 & & \\ & \ddots & & & & & & \\ & & & 1 & & & & \\ & & & l_{k+1,k} & 1 & & & \\ 0 & & & \vdots & & \ddots & & \\ & & & l_{nk} & & & 0 & 1 \end{pmatrix}$$

Sei nun die Ausgangsmatrix $A = A^{(1)}$ invertierbar. Dann ist also nach jedem Schritt (Pivotsuche + Elimination) das Ergebnis $A^{(k+1)} = L_k P_k A^{(k)}$ wieder invertierbar und wir erhalten, wie wir uns schon überlegt haben, die Struktur

$$(4.12) \quad (A^{(k+1)}, b^{(k+1)}) = L_k P_k (A^{(k)}, b^{(k)}) = \left(\begin{array}{ccc|cc|c} r_{11} & \cdots & r_{1k} & \cdots & r_{1n} & c_1 \\ 0 & \ddots & & & \vdots & \vdots \\ & & r_{kk} & \cdots & r_{kn} & c_k \\ \hline & & 0 & & & \\ & & \vdots & \hat{A}^{(k+1)} & & \hat{b}^{(k+1)} \\ & & 0 & & & \end{array}\right).$$

Nach den $n-1$ Schritten des Gaußschen Algorithmus erhalten wir somit

$$R = A^{(n)} = L_{n-1} P_{n-1} \cdots L_1 P_1 A.$$

Multiplikation mit

$$(L_{n-1} P_{n-1} \cdots L_1 P_1)^{-1} = P_1^{-1} L_1^{-1} \cdots P_{n-1}^{-1} L_{n-1}^{-1}$$

ergibt wegen $P_k^{-1} = P_k$

$$A = P_1 L_1^{-1} \cdots P_{n-1} L_{n-1}^{-1} R.$$

Sind im Eliminationsverfahren keine Zeilenvertauschungen nötig, dann erhalten wir

$$A = L_1^{-1} \cdots L_{n-1}^{-1} R =: LR.$$

und man rechnet mit (4.11) leicht nach, dass gilt

$$L = L_1^{-1} \cdots L_{n-1}^{-1} = \begin{pmatrix} 1 & & 0 & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ \vdots & & \ddots & \ddots & \\ l_{n1} & & \cdots & l_{n,n-1} & 1 \end{pmatrix} = I + L^{(n)}.$$

Mit den vom Gauß-Verfahren gelieferten Matrizen L und R gilt also ohne Pivotsuche

$$A = LR.$$

Allgemein gilt der folgende Satz.

Satz 4.2.2 *Es sei $A \in \mathbb{R}^{n,n}$ nichtsingulär. Dann gilt:*

i) Das Gaußsche Eliminationsverfahren aus Algorithmus 1 liefert eine untere Dreiecksmatrix L der Form (4.7) *und eine obere Dreiecksmatrix R mit*

$$LR = PA.$$

Hierbei ist $P = P_{n-1} \cdots P_1$ eine Permutationsmatrix, wobei jeweils P_k die Permutationsmatrix für die Zeilenvertauschung im k-ten Schritt ist.

ii) Algorithmus 2 liefert eine Dreieckszerlegung

$$LR = PAQ$$

Hierbei ist P wie eben und $Q = Q_1 \cdots Q_{n-1}$, wobei jeweils Q_k die Permutationsmatrix für die Spaltenvertauschung im k-ten Schritt ist.

Beweis: Finden keine Zeilenvertauschungen statt, dann haben wir die Behauptung bereits gezeigt.

Für Interessierte: Der allgemeine Fall ist etwas technisch. Setze $C_k = L_k^{-1} - I$. Man prüft leicht die Gültigkeit der Formel

$$L_k^{-1} P_i = P_i (P_i C_k + I), \quad i \geq k.$$

Dies ergibt

$$P_1 L_1^{-1} \cdots P_{n-1} L_{n-1}^{-1} = P_1 \cdots P_{n-1} (P_{n-1} \cdots P_2 C_1 + I) \cdots (P_{n-1} C_{n-2} + I) L_{n-1}^{-1} = P^{-1} L,$$

wobei die Faktoren

$$\tilde{L}_k = P_{n-1} \cdots P_{k+1} C_k + I$$

aus L_k^{-1} durch Permutation der Einträge l_{ik} entstehen, die sich aus den nachfolgenden Pivotschritten ergeben. Dieselben Einträge l_{ik} werden vom Gauß-Verfahren in $L^{(k+1}$ eingetragen und bis zum Ergebnis $L^{(n)}$ genauso vertauscht.

Algorithmus 2 ist nichts anderes als Algorithmus 1 angewendet auf AQ. □

Es gibt einige wichtige Teilklassen von Matrizen, bei denen auf die Pivotsuche verzichtet werden kann:

Matrizenklassen, die keine Pivotsuche erfordern

- $A = A^T$ ist symmetrisch posititv definit, also

  $$x^T A x > 0 \quad \forall\, x \in \mathbb{R}^n \setminus \{0\}.$$

  Wir gehen hierauf noch ein.

- A ist strikt diagonaldominant, d.h.

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}|, \quad i = 1, \ldots, n,$$

  siehe Übung.

- A ist M-Matrix, d.h. es gilt

$$\begin{aligned} & a_{ii} > 0, \;\; i = 1, \ldots, n, \\ & a_{ij} \leq 0, \;\; i \neq j \\ & D^{-1}(A - D), \;\; D = \operatorname{diag}(a_{11}, \ldots, a_{nn}) \text{ hat lauter Eigenwerte vom Betrag } < 1. \end{aligned}$$

Nach Satz 4.2.2 gibt es für eine invertierbare Matrix A immer eine Permutationsmatrix P, so dass eine Dreieckszerlegung

$$LR = PA$$

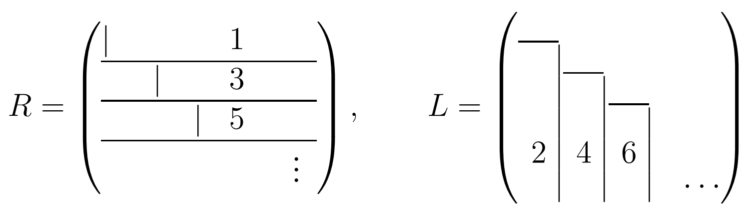
mit L, R der Form (4.7), (4.4) existiert. Tatsächlich sind L, R eindeutig bestimmt:

Satz 4.2.3 *Sei $A \in \mathbb{R}^{n,n}$ invertierbar und sei $P \in \mathbb{R}^{n,n}$ eine Permutationsmatrix, so dass eine Dreieckszerlegung* (4.6) *mit L, R der Form* (4.7), (4.4) *existiert. Dann sind L und R eindeutig bestimmt.*

Beweis: **Für Interessierte:** Die Beziehung $LR = B$ liefert die folgenden n^2 Gleichungen für die n^2 unbekannten Einträge in L und R

$$b_{ij} = \sum_{k=1}^{\min(i,j)} l_{ik} r_{kj}, \quad (l_{ii} = 1).$$

Hieraus lassen sich l_{ik} und r_{kj} zum Beispiel in folgender Reihenfolge berechnen:



□