Numerische Verfahren

Jens-Peter M. Zemke
zemke@tu-harburg.de

Institut für Numerische Simulation
Technische Universität Hamburg-Harburg

29.04.2008

TUHH
Technische Universität Hamburg-Harburg

Übersicht

Lineare Systeme

Wir betrachten in diesem Abschnitt das Problem, ein lineares Gleichungssystem

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

Lineare Systeme

Wir betrachten in diesem Abschnitt das Problem, ein lineares Gleichungssystem

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

mit einer gegebenen regulären Matrix

$$\boldsymbol{A} \in \mathbb{R}^{n \times n}$$

Lineare Systeme

Wir betrachten in diesem Abschnitt das Problem, ein lineares Gleichungssystem

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

mit einer gegebenen regulären Matrix

$$\boldsymbol{A} \in \mathbb{R}^{n \times n}$$

und einem gegebenen Vektor

$$\boldsymbol{b} \in \mathbb{R}^{n}$$

zu lösen.

Zerlegung regulärer Matrizen

Wir betrachten das lineare Gleichungssystem

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \quad \boldsymbol{A} \in \mathbb{R}^{n\times n}, \quad \boldsymbol{b} \in \mathbb{R}^n. \tag{4.1}$$

Es sei $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ regulär.

Zerlegung regulärer Matrizen

Wir betrachten das lineare Gleichungssystem

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \quad \boldsymbol{A} \in \mathbb{R}^{n\times n}, \quad \boldsymbol{b} \in \mathbb{R}^n. \tag{4.1}$$

Es sei $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ regulär. Dann existiert (vgl. LA Satz 4.42)

- eine Permutationsmatrix $\boldsymbol{P}$,
- eine normierte untere Dreiecksmatrix $\boldsymbol{L}$, ($\ell_{jj} = 1 \; \forall \, j = 1, \ldots, n$) und
- eine obere Dreiecksmatrix $\boldsymbol{R}$

mit

$$\boldsymbol{P}\boldsymbol{A} = \boldsymbol{L}\boldsymbol{R}. \tag{4.2}$$

Zerlegung regulärer Matrizen

Wir betrachten das lineare Gleichungssystem

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \quad \boldsymbol{A} \in \mathbb{R}^{n\times n}, \quad \boldsymbol{b} \in \mathbb{R}^n. \tag{4.1}$$

Es sei $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ regulär. Dann existiert (vgl. LA Satz 4.42)

- eine Permutationsmatrix $\boldsymbol{P}$,
- eine normierte untere Dreiecksmatrix $\boldsymbol{L}$, ($\ell_{jj} = 1 \; \forall \, j = 1, \ldots, n$) und
- eine obere Dreiecksmatrix $\boldsymbol{R}$

mit

$$\boldsymbol{P}\boldsymbol{A} = \boldsymbol{L}\boldsymbol{R}. \tag{4.2}$$

Diese Zerlegung heißt die LR-Zerlegung (mit partieller Pivotisierung) der Matrix $\boldsymbol{A}$.

Zerlegung regulärer Matrizen

Es gilt $\boldsymbol{P} = \boldsymbol{E}$ genau dann, wenn alle Hauptuntermatrizen

$$\boldsymbol{A}(1:k, 1:k) := (a_{ij})_{i,j=1}^{k}, \quad k = 1, \ldots, n,$$

von $\boldsymbol{A}$ regulär.

Zerlegung regulärer Matrizen

Es gilt $\boldsymbol{P} = \boldsymbol{E}$ genau dann, wenn alle Hauptuntermatrizen

$$\boldsymbol{A}(1:k, 1:k) := (a_{ij})_{i,j=1}^{k}, \quad k = 1, \ldots, n,$$

von $\boldsymbol{A}$ regulär.

Ist dies der Fall, so kann man die LR-Zerlegung von $\boldsymbol{A}$ mit dem Gaußschen Eliminationsverfahren bestimmen.

Zerlegung regulärer Matrizen

Es gilt $\boldsymbol{P} = \boldsymbol{E}$ genau dann, wenn alle Hauptuntermatrizen

$$\boldsymbol{A}(1:k, 1:k) := (a_{ij})_{i,j=1}^{k}, \quad k = 1, \ldots, n,$$

von $\boldsymbol{A}$ regulär.

Ist dies der Fall, so kann man die LR-Zerlegung von $\boldsymbol{A}$ mit dem Gaußschen Eliminationsverfahren bestimmen.

Dabei speichern wir die von Null verschiedenen Elemente von $\boldsymbol{R}$ in dem oberen Dreieck von $\boldsymbol{A}$ und die Elemente unterhalb der Diagonale von $\boldsymbol{L}$ in dem strikten unteren Dreieck von $\boldsymbol{A}$ ab.

Zerlegung regulärer Matrizen

Algorithmus 4.1: (LR-Zerlegung ohne Pivotsuche)

```
for i = 1:n-1
   for j = i+1:n
      % Bestimme die i-te Spalte von L
      a(j,i) = a(j,i)/a(i,i);
      for k = i+1:n
         % Datiere die j-te Zeile auf
         a(j,k) = a(j,k) - a(j,i)*a(i,k);
      end
   end
end
```

Ist die LR-Zerlegung von $\boldsymbol{A}$ bekannt, so kann man die Lösung des Gleichungssystems (4.1) schreiben als

$$\boldsymbol{Ax} = \boldsymbol{LRx} =: \boldsymbol{Ly} = \boldsymbol{b}, \quad \boldsymbol{Rx} = \boldsymbol{y},$$

Zerlegung regulärer Matrizen

. . . und durch Vorwärtseinsetzen

Algorithmus 4.2: (Vorwärtseinsetzen)

```
for j = 1:n
   y(j) = b(j);
   for k = 1:j-1
      y(j) = y(j) - a(j,k)*y(k);
   end

end
```

die Lösung y von $Ly = b$ bestimmen und durch Rückwärtseinsetzen . . .

Zerlegung regulärer Matrizen

. . . und durch Rückwärtseinsetzen

Algorithmus 4.3: (Rückwärtseinsetzen)

```
for j = n:-1:1
   x(j) = y(j);
   for k = j+1:n
      x(j) = x(j) - a(j,k)*x(k);
   end
   x(j) = x(j)/a(j,j);
end
```

die Lösung $\boldsymbol{x}$ von $\boldsymbol{Rx} = \boldsymbol{y}$, d.h. von $\boldsymbol{Ax} = \boldsymbol{b}$.

Zerlegung regulärer Matrizen

Als Aufwand der LR-Zerlegung erhält man

$$\begin{aligned}\sum_{i=1}^{n-1}\Big(\sum_{j=i+1}^{n}(1+\sum_{k=i+1}^{n}2)\Big) &= \sum_{i=1}^{n-1}\Big((n-i)+2(n-i)^2\Big)\\ &= \frac{2}{3}n^3-\frac{1}{2}n^2-\frac{1}{6}n\\ &= \frac{2}{3}n^3+O(n^2).\end{aligned}$$

flops (floating point operations).

Zerlegung regulärer Matrizen

Als Aufwand der LR-Zerlegung erhält man

$$\begin{aligned}\sum_{i=1}^{n-1}\Big(\sum_{j=i+1}^{n}(1+\sum_{k=i+1}^{n}2)\Big) &= \sum_{i=1}^{n-1}\Big((n-i)+2(n-i)^2\Big)\\ &= \frac{2}{3}n^3-\frac{1}{2}n^2-\frac{1}{6}n\\ &= \frac{2}{3}n^3+O(n^2).\end{aligned}$$

flops (floating point operations).

Vorwärts- und Rückwärtseinsetzen erfordert jeweils $O(n^2)$ flops.

Zerlegung regulärer Matrizen

Existiert eine singuläre Hauptuntermatrix $\boldsymbol{A}(1:i,1:i)$ der regulären Matrix $\boldsymbol{A}$, so bricht Algorithmus 4.1 ab, da das Pivotelement $a(i,i)$ bei der Elimination der i-ten Variable Null wird.

Zerlegung regulärer Matrizen

Existiert eine singuläre Hauptuntermatrix $\boldsymbol{A}(1:i,1:i)$ der regulären Matrix $\boldsymbol{A}$, so bricht Algorithmus 4.1 ab, da das Pivotelement $a(i,i)$ bei der Elimination der i-ten Variable Null wird.

In diesem Fall gibt es ein $a_{ij} \neq 0, j > i$ (das im Verlaufe des Algorithmus erzeugt worden ist), und man kann die aktuelle Zeile i mit der Zeile j vertauschen und den Algorithmus fortsetzen.

Zerlegung regulärer Matrizen

Sammelt man diese Vertauschungen in der Permutationsmatrix $\boldsymbol{P}$, so erhält man am Ende des so modifizierten Algorithmus 4.1 eine LR-Zerlegung der Matrix $\boldsymbol{PA}$:

$$\boldsymbol{PA} = \boldsymbol{LR}.$$

Zerlegung regulärer Matrizen

Sammelt man diese Vertauschungen in der Permutationsmatrix $\boldsymbol{P}$, so erhält man am Ende des so modifizierten Algorithmus 4.1 eine LR-Zerlegung der Matrix $\boldsymbol{PA}$:

$$\boldsymbol{PA} = \boldsymbol{LR}.$$

Aus Stabilitätsgründen empfiehlt es sich, auch dann eine Vertauschung vorzunehmen, wenn a_{ii} zwar von Null verschieden ist, in der Restmatrix $\boldsymbol{A}(i:n, i:n)$ aber Elemente vorhanden sind, deren Betrag wesentlich größer ist als der Betrag von a_{ii}.

Zerlegung regulärer Matrizen

Beispiel 4.4: Es sei

$$\boldsymbol{A} = \begin{pmatrix} 10^{-4} & 1 \\ 1 & 1 \end{pmatrix}.$$

Zerlegung regulärer Matrizen

Beispiel 4.4: Es sei

$$\boldsymbol{A} = \begin{pmatrix} 10^{-4} & 1 \\ 1 & 1 \end{pmatrix}.$$

Wir bezeichnen mit $\mathrm{fl}(a)$ die Gleitpunktdarstellung von a. Dann liefert Algorithmus 4.1 bei dreistelliger Rechnung

$$\boldsymbol{L} = \begin{pmatrix} 1 & 0 \\ \mathrm{fl}(1/10^{-4}) & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 10^{4} & 1 \end{pmatrix}$$

und

$$\boldsymbol{R} = \begin{pmatrix} 10^{-4} & 1 \\ 0 & \mathrm{fl}(1 - 10^{4}) \end{pmatrix} = \begin{pmatrix} 10^{-4} & 1 \\ 0 & -10^{4} \end{pmatrix},$$

Zerlegung regulärer Matrizen

Beispiel 4.4: Es sei

$$\boldsymbol{A} = \begin{pmatrix} 10^{-4} & 1 \\ 1 & 1 \end{pmatrix}.$$

Wir bezeichnen mit $\mathrm{fl}(a)$ die Gleitpunktdarstellung von a. Dann liefert Algorithmus 4.1 bei dreistelliger Rechnung

$$\boldsymbol{L} = \begin{pmatrix} 1 & 0 \\ \mathrm{fl}(1/10^{-4}) & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 10^{4} & 1 \end{pmatrix}$$

und

$$\boldsymbol{R} = \begin{pmatrix} 10^{-4} & 1 \\ 0 & \mathrm{fl}(1 - 10^{4}) \end{pmatrix} = \begin{pmatrix} 10^{-4} & 1 \\ 0 & -10^{4} \end{pmatrix},$$

und damit

$$\boldsymbol{LR} = \begin{pmatrix} 10^{-4} & 1 \\ 1 & 0 \end{pmatrix} \neq \boldsymbol{A}.$$

□

Zerlegung regulärer Matrizen

In vielen Fällen reicht es aus, vor dem Annullieren der Elemente der i-ten Spalte $j \in \{i, \ldots, n\}$ zu bestimmen mit $|a_{ji}| \geqslant |a_{ki}|$ für alle $k = i, \ldots, n$ und dann die i-te Zeile mit der j-ten Zeile zu vertauschen.

Zerlegung regulärer Matrizen

In vielen Fällen reicht es aus, vor dem Annullieren der Elemente der i-ten Spalte $j \in \{i, \ldots, n\}$ zu bestimmen mit $|a_{ji}| \geqslant |a_{ki}|$ für alle $k = i, \ldots, n$ und dann die i-te Zeile mit der j-ten Zeile zu vertauschen.

Dieses Vorgehen heißt Spaltenpivotsuche oder partielle Pivotisierung.

Zerlegung regulärer Matrizen

Algorithmus 4.5: (LR-Zerlegung mit Spaltenpivotsuche)

```
for i = 1:n-1
```

Wähle $j \geqslant i$ mit $|a_{ji}| \geqslant |a_{ki}|$ für alle $k \geqslant i$ und vertausche die i-te mit der j-ten Zeile

```
   for j = i+1:n
      a(j,i) = a(j,i)/a(i,i);
      for k = i+1:n
         a(j,k) = a(j,k) - a(j,i)*a(i,k);
      end
   end
end
```

Zerlegung regulärer Matrizen

Beispiel 4.6: Für Beispiel 4.4 erhält man mit Algorithmus 4.5

$$\boldsymbol{L} = \begin{pmatrix} 1 & 0 \\ 10^{-4} & 1 \end{pmatrix}, \quad \boldsymbol{R} = \begin{pmatrix} 1 & 1 \\ 0 & \mathrm{fl}(1 - 10^{-4}) \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

Zerlegung regulärer Matrizen

Beispiel 4.6: Für Beispiel 4.4 erhält man mit Algorithmus 4.5

$$\boldsymbol{L} = \begin{pmatrix} 1 & 0 \\ 10^{-4} & 1 \end{pmatrix}, \quad \boldsymbol{R} = \begin{pmatrix} 1 & 1 \\ 0 & \mathrm{fl}(1 - 10^{-4}) \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

und

$$\boldsymbol{LR} = \begin{pmatrix} 1 & 1 \\ 10^{-4} & 1 + 10^{-4} \end{pmatrix}$$

ist eine gute Approximation von

$$\boldsymbol{PA} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \boldsymbol{A} = \begin{pmatrix} 1 & 1 \\ 10^{-4} & 1 \end{pmatrix}.$$

Zerlegung regulärer Matrizen

Nicht in allen Fällen führt die Spaltenpivotsuche zum Erfolg. Wendet man Algorithmus 4.5 bei dreistelliger Rechnung auf

$$\tilde{\boldsymbol{A}} = \begin{pmatrix} 1 & 10^4 \\ 1 & 1 \end{pmatrix}$$

an

Zerlegung regulärer Matrizen

Nicht in allen Fällen führt die Spaltenpivotsuche zum Erfolg. Wendet man Algorithmus 4.5 bei dreistelliger Rechnung auf

$$\tilde{\boldsymbol{A}} = \begin{pmatrix} 1 & 10^4 \\ 1 & 1 \end{pmatrix}$$

an, so erhält man

$$\boldsymbol{L} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \quad \boldsymbol{R} = \begin{pmatrix} 1 & 10^4 \\ 0 & \mathrm{fl}(1 - 10^4) \end{pmatrix} = \begin{pmatrix} 1 & 10^4 \\ 0 & -10^4 \end{pmatrix}$$

Zerlegung regulärer Matrizen

Nicht in allen Fällen führt die Spaltenpivotsuche zum Erfolg. Wendet man Algorithmus 4.5 bei dreistelliger Rechnung auf

$$\tilde{\boldsymbol{A}} = \begin{pmatrix} 1 & 10^4 \\ 1 & 1 \end{pmatrix}$$

an, so erhält man

$$\boldsymbol{L} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \quad \boldsymbol{R} = \begin{pmatrix} 1 & 10^4 \\ 0 & \mathrm{fl}(1 - 10^4) \end{pmatrix} = \begin{pmatrix} 1 & 10^4 \\ 0 & -10^4 \end{pmatrix}$$

und hiermit

$$\boldsymbol{LR} = \begin{pmatrix} 1 & 10^4 \\ 1 & 0 \end{pmatrix}.$$

□

Zerlegung regulärer Matrizen

Treten, wie in unserem Beispiel, in der Matrix Elemente sehr unterschiedlicher Größenordnung auf, so empfiehlt es sich, eine vollständige Pivotsuche auszuführen.

Zerlegung regulärer Matrizen

Treten, wie in unserem Beispiel, in der Matrix Elemente sehr unterschiedlicher Größenordnung auf, so empfiehlt es sich, eine vollständige Pivotsuche auszuführen.

In diesem Fall werden im i-ten Eliminationsschritt ein Zeilenindex $j \geqslant i$ und ein Spaltenindex $k \geqslant i$ bestimmt mit $|a_{jk}| \geqslant |a_{\ell m}|$ für alle $\ell, m \geqslant i$, und es wird vor der Elimination die i-te mit der j-ten Zeile und die i-te Spalte mit der k-ten Spalte getauscht.

Zerlegung regulärer Matrizen

Treten, wie in unserem Beispiel, in der Matrix Elemente sehr unterschiedlicher Größenordnung auf, so empfiehlt es sich, eine vollständige Pivotsuche auszuführen.

In diesem Fall werden im i-ten Eliminationsschritt ein Zeilenindex $j \geqslant i$ und ein Spaltenindex $k \geqslant i$ bestimmt mit $|a_{jk}| \geqslant |a_{\ell m}|$ für alle $\ell, m \geqslant i$, und es wird vor der Elimination die i-te mit der j-ten Zeile und die i-te Spalte mit der k-ten Spalte getauscht.

Man erhält dann eine Zerlegung

$$\boldsymbol{PAQ} = \boldsymbol{LR},$$

wobei die Permutationsmatrix $\boldsymbol{P}$ die Zeilenvertauschungen in $\boldsymbol{A}$ vornimmt und die Permutationsmatrix $\boldsymbol{Q}$ die Spaltenvertauschungen.

Zerlegung regulärer Matrizen

Ist die reguläre Matrix $\boldsymbol{A}$ symmetrisch und existiert eine LR-Zerlegung, so kann man $\boldsymbol{R}$ als Produkt einer Diagonalmatrix $\boldsymbol{D}$ und einer normierten oberen Dreiecksmatrix $\tilde{\boldsymbol{R}}$ schreiben.

Zerlegung regulärer Matrizen

Ist die reguläre Matrix $\boldsymbol{A}$ symmetrisch und existiert eine LR-Zerlegung, so kann man $\boldsymbol{R}$ als Produkt einer Diagonalmatrix $\boldsymbol{D}$ und einer normierten oberen Dreiecksmatrix $\tilde{\boldsymbol{R}}$ schreiben. Hiermit gilt dann

$$\boldsymbol{A}^T = \tilde{\boldsymbol{R}}^T \boldsymbol{D} \boldsymbol{L}^T$$

mit einer normierten unteren Dreiecksmatrix $\tilde{\boldsymbol{R}}^T$ und einer oberen Dreiecksmatrix $\boldsymbol{D}\boldsymbol{L}^T$.

Zerlegung regulärer Matrizen

Ist die reguläre Matrix $\boldsymbol{A}$ symmetrisch und existiert eine LR-Zerlegung, so kann man $\boldsymbol{R}$ als Produkt einer Diagonalmatrix $\boldsymbol{D}$ und einer normierten oberen Dreiecksmatrix $\tilde{\boldsymbol{R}}$ schreiben. Hiermit gilt dann

$$\boldsymbol{A}^T = \tilde{\boldsymbol{R}}^T \boldsymbol{D} \boldsymbol{L}^T$$

mit einer normierten unteren Dreiecksmatrix $\tilde{\boldsymbol{R}}^T$ und einer oberen Dreiecksmatrix $\boldsymbol{D}\boldsymbol{L}^T$.

Da die LR-Zerlegung einer regulären Matrix eindeutig bestimmt ist, folgt $\tilde{\boldsymbol{R}}^T = \boldsymbol{L}$. Damit kann man $\boldsymbol{A}$ auch schreiben als

$$\boldsymbol{A} = \boldsymbol{L}\boldsymbol{D}\boldsymbol{L}^T$$

mit einer Diagonalmatrix $\boldsymbol{D}$ und $\boldsymbol{L}$ wie oben. Diese Zerlegung heißt die LDLT-Zerlegung von $\boldsymbol{A}$.

Zerlegung regulärer Matrizen

Ist die symmetrische Matrix $\boldsymbol{A}$ zusätzlich positiv definit, so sind alle Hauptuntermatrizen von $\boldsymbol{A}$ ebenfalls positiv definit, also regulär und daher existiert in diesem Fall die LDLT-Zerlegung.

Zerlegung regulärer Matrizen

Ist die symmetrische Matrix $\boldsymbol{A}$ zusätzlich positiv definit, so sind alle Hauptuntermatrizen von $\boldsymbol{A}$ ebenfalls positiv definit, also regulär und daher existiert in diesem Fall die LDLT-Zerlegung.

Ferner sind die Diagonalelemente von $\boldsymbol{D} = diag\{d_1, \ldots, d_n\}$ positiv, und man kann mit

$$\boldsymbol{C} = \boldsymbol{L}\, diag\{\sqrt{d_1}, \ldots, \sqrt{d_n}\}$$

die LDLT-Zerlegung von $\boldsymbol{A}$ schreiben als

$$\boldsymbol{A} = \boldsymbol{C}\boldsymbol{C}^T. \tag{4.3}$$

Zerlegung regulärer Matrizen

Ist die symmetrische Matrix $\boldsymbol{A}$ zusätzlich positiv definit, so sind alle Hauptuntermatrizen von $\boldsymbol{A}$ ebenfalls positiv definit, also regulär und daher existiert in diesem Fall die LDLT-Zerlegung.

Ferner sind die Diagonalelemente von $\boldsymbol{D} = diag\{d_1, \ldots, d_n\}$ positiv, und man kann mit

$$\boldsymbol{C} = \boldsymbol{L}\, diag\{\sqrt{d_1}, \ldots, \sqrt{d_n}\}$$

die LDLT-Zerlegung von $\boldsymbol{A}$ schreiben als

$$\boldsymbol{A} = \boldsymbol{C}\boldsymbol{C}^T. \tag{4.3}$$

Diese Zerlegung heißt die Cholesky-Zerlegung von $\boldsymbol{A}$.

Zerlegung regulärer Matrizen

Prinzipiell kann man die Cholesky-Zerlegung mit Hilfe des Gaußschen Eliminationsverfahrens bestimmen.

Zerlegung regulärer Matrizen

Prinzipiell kann man die Cholesky-Zerlegung mit Hilfe des Gaußschen Eliminationsverfahrens bestimmen.

Man benötigt dann $\frac{2}{3}n^3 + O(n^2)$ Operationen und n^2 Speicherplätze.

Zerlegung regulärer Matrizen

Prinzipiell kann man die Cholesky-Zerlegung mit Hilfe des Gaußschen Eliminationsverfahrens bestimmen.

Man benötigt dann $\frac{2}{3}n^3 + O(n^2)$ Operationen und n^2 Speicherplätze.

Durch direkten Vergleich der Elemente in (4.3) erhält man einen Algorithmus, der mit der Hälfte des Aufwandes auskommt.

Zerlegung regulärer Matrizen

Da C eine untere Dreiecksmatrix ist, gilt $c_{ij} = 0$ für $1 \leqslant i < j \leqslant n$, und daher ist

$$a_{ij} = \sum_{k=1}^{n} c_{ik} c_{jk} = \sum_{k=1}^{i} c_{ik} c_{jk}.$$

Zerlegung regulärer Matrizen

Da C eine untere Dreiecksmatrix ist, gilt $c_{ij} = 0$ für $1 \leqslant i < j \leqslant n$, und daher ist

$$a_{ij} = \sum_{k=1}^{n} c_{ik} c_{jk} = \sum_{k=1}^{i} c_{ik} c_{jk}.$$

Speziell für $i = j = 1$ bedeutet dies

$$a_{11} = c_{11}^2, \quad \text{d.h.} \quad c_{11} = \sqrt{a_{11}},$$

Zerlegung regulärer Matrizen

Da $\boldsymbol{C}$ eine untere Dreiecksmatrix ist, gilt $c_{ij} = 0$ für $1 \leqslant i < j \leqslant n$, und daher ist

$$a_{ij} = \sum_{k=1}^{n} c_{ik} c_{jk} = \sum_{k=1}^{i} c_{ik} c_{jk}.$$

Speziell für $i = j = 1$ bedeutet dies

$$a_{11} = c_{11}^2, \quad \text{d.h.} \quad c_{11} = \sqrt{a_{11}},$$

und für $i = 1$ und $j = 2, \ldots, n$

$$a_{1j} = c_{11} c_{j1}, \quad \text{d.h.} \quad c_{j1} = a_{j1} / c_{11}.$$

Zerlegung regulärer Matrizen

Da $\boldsymbol{C}$ eine untere Dreiecksmatrix ist, gilt $c_{ij} = 0$ für $1 \leqslant i < j \leqslant n$, und daher ist

$$a_{ij} = \sum_{k=1}^{n} c_{ik} c_{jk} = \sum_{k=1}^{i} c_{ik} c_{jk}.$$

Speziell für $i = j = 1$ bedeutet dies

$$a_{11} = c_{11}^2, \quad \text{d.h.} \quad c_{11} = \sqrt{a_{11}},$$

und für $i = 1$ und $j = 2, \ldots, n$

$$a_{1j} = c_{11} c_{j1}, \quad \text{d.h.} \quad c_{j1} = a_{j1} / c_{11}.$$

Damit ist die erste Spalte von $\boldsymbol{C}$ bestimmt.

Zerlegung regulärer Matrizen

Sind schon $c_{\mu\nu}$ für $\nu = 1, \ldots, i-1$ und $\mu = \nu, \nu+1, \ldots, n$ bestimmt, so erhält man aus

$$a_{ii} = c_{ii}^2 + \sum_{k=1}^{i-1} c_{ik}^2$$

Zerlegung regulärer Matrizen

Sind schon $c_{\mu\nu}$ für $\nu = 1, \ldots, i-1$ und $\mu = \nu, \nu+1, \ldots, n$ bestimmt, so erhält man aus

$$a_{ii} = c_{ii}^2 + \sum_{k=1}^{i-1} c_{ik}^2$$

das i-te Diagonalelement

$$c_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} c_{ik}^2}$$

von $\boldsymbol{C}$,

Zerlegung regulärer Matrizen

und

$$a_{ij} = c_{ii}c_{ji} + \sum_{k=1}^{i-1} c_{ik}c_{jk}$$

Zerlegung regulärer Matrizen

und

$$a_{ij} = c_{ii}c_{ji} + \sum_{k=1}^{i-1} c_{ik}c_{jk}$$

liefert die i-te Spalte

$$c_{ji} = \frac{1}{c_{ii}}\left(a_{ij} - \sum_{k=1}^{i-1} c_{ik}c_{jk}\right), \quad j = i+1, \ldots, n.$$

Zerlegung regulärer Matrizen

und

$$a_{ij} = c_{ii}c_{ji} + \sum_{k=1}^{i-1} c_{ik}c_{jk}$$

liefert die i-te Spalte

$$c_{ji} = \frac{1}{c_{ii}}\left(a_{ij} - \sum_{k=1}^{i-1} c_{ik}c_{jk}\right), \quad j = i+1, \ldots, n.$$

Damit erhält man das folgende Verfahren zur Bestimmung der Cholesky-Zerlegung. Dabei überschreiben wir das untere Dreieck von $\boldsymbol{A}$ durch die wesentlichen Elemente von $\boldsymbol{C}$.

Zerlegung regulärer Matrizen

Algorithmus 4.7: (Cholesky-Zerlegung)

```
for i = 1:n
   for k = 1:i-1
      a(i,i) = a(i,i) - a(i,k)*a(i,k);
   end
   a(i,i) = sqrt(a(i,i));
   for j = i+1:n
      for k = 1:i-1
         a(j,i) = a(j,i) - a(i,k)*a(j,k);
      end
      a(j,i) = a(j,i)/a(i,i);
   end
end
```

Zerlegung regulärer Matrizen

Der Aufwand des Cholesky-Verfahrens ist

$$\begin{aligned}\sum_{i=1}^{n}\Big(\sum_{k=1}^{i-1}2+\sum_{j=i+1}^{n}\big(1+\sum_{k=1}^{i-1}2\big)\Big) &= \sum_{i=1}^{n}\Big(2(i-1)+(n-i)(2i-1)\Big)\\ &= \frac{1}{3}n^3+\frac{1}{2}n^2-\frac{5}{6}n\\ &= \frac{1}{3}n^3+O(n^2)\end{aligned}$$

flops und n Quadratwurzeln.

Zerlegung regulärer Matrizen

Der Aufwand des Cholesky-Verfahrens ist

$$\begin{aligned}\sum_{i=1}^{n}\Big(\sum_{k=1}^{i-1}2+\sum_{j=i+1}^{n}\big(1+\sum_{k=1}^{i-1}2\big)\Big) &= \sum_{i=1}^{n}\Big(2(i-1)+(n-i)(2i-1)\Big)\\ &= \frac{1}{3}n^3+\frac{1}{2}n^2-\frac{5}{6}n\\ &= \frac{1}{3}n^3+O(n^2)\end{aligned}$$

flops und n Quadratwurzeln.

Da eine Quadratwurzel auf den meisten Rechnern mit IEEE 754-Arithmetik in etwa so teuer ist wie ein flop, ist der Aufwand also vergleichbar mit

$$\frac{1}{3}n^3+\frac{1}{2}n^2+\frac{1}{6}n.$$

Zerlegung regulärer Matrizen

Besitzt die symmetrische Matrix A eine LDLT-Zerlegung, so kann man diese mit einem ähnlichen Verfahren wie Algorithmus 4.7 bestimmen. Man beachte aber, dass eine Pivotsuche wie beim Gaußschen Eliminationsverfahren die Symmetrie der Matrix zerstört.

Zerlegung regulärer Matrizen

Besitzt die symmetrische Matrix $\boldsymbol{A}$ eine LDLT-Zerlegung, so kann man diese mit einem ähnlichen Verfahren wie Algorithmus 4.7 bestimmen. Man beachte aber, dass eine Pivotsuche wie beim Gaußschen Eliminationsverfahren die Symmetrie der Matrix zerstört.

Man muss also gleichlautende Zeilen- und Spaltenvertauschungen vornehmen. Auf diese Weise kann man nicht immer für eine reguläre Matrix ein von Null verschiedenes Pivotelement erzeugen, wie das Beispiel

$$\boldsymbol{A} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

zeigt.

Zerlegung regulärer Matrizen

Besitzt die symmetrische Matrix $\boldsymbol{A}$ eine LDLT-Zerlegung, so kann man diese mit einem ähnlichen Verfahren wie Algorithmus 4.7 bestimmen. Man beachte aber, dass eine Pivotsuche wie beim Gaußschen Eliminationsverfahren die Symmetrie der Matrix zerstört.

Man muss also gleichlautende Zeilen- und Spaltenvertauschungen vornehmen. Auf diese Weise kann man nicht immer für eine reguläre Matrix ein von Null verschiedenes Pivotelement erzeugen, wie das Beispiel

$$\boldsymbol{A} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

zeigt.

Auch wenn die LDLT-Zerlegung existiert, kann die Übertragung von Algorithmus 4.7 instabil sein.

Modifikationen des Gaußschen Verfahrens

In Algorithmus 4.1 haben wir die Elimination durchgeführt, indem wir im i-ten Schritt ein geeignetes Vielfaches der i-ten Zeile von den nachfolgenden Zeilen abgezogen haben. In Vektorschreibweise haben wir also

Algorithmus 4.8: (LR-Zerlegung; zeilenorientiert)

```
for i = 1:n-1
   for j = i+1:n
      a(j,i) = a(j,i)/a(i,i);
      a(j,i+1:n) = a(j,i+1:n) - a(j,i)*a(i,i+1:n);
   end
end
```

Modifikationen des Gaußschen Verfahrens

Vertauscht man die beiden inneren Schleifen, so erhält man eine spaltenorientierte Version des Gaußschen Eliminationsverfahrens.

Algorithmus 4.9: (LR-Zerlegung; spaltenorientiert)

```
for i = 1:n-1
   a(i+1:n,i) = a(i+1:n,i)/a(i,i);
   for k = i+1:n
      a(i+1:n,k) = a(i+1:n,k) - a(i,k)*a(i+1:n,i);
   end
end
```

Modifikationen des Gaußschen Verfahrens

Auch die i-Schleife kann man mit der j-Schleife und/oder der k-Schleife vertauschen. Man erhält insgesamt 6 Varianten der Gauß-Elimination, die alle von der Zahl der Rechenoperationen her denselben Aufwand besitzen.

Modifikationen des Gaußschen Verfahrens

Auch die i-Schleife kann man mit der j-Schleife und/oder der k-Schleife vertauschen. Man erhält insgesamt 6 Varianten der Gauß-Elimination, die alle von der Zahl der Rechenoperationen her denselben Aufwand besitzen.

Sie können sich aber auf verschiedenen Plattformen unter verschiedenen Programmiersprachen sehr unterschiedlich verhalten.

Modifikationen des Gaußschen Verfahrens

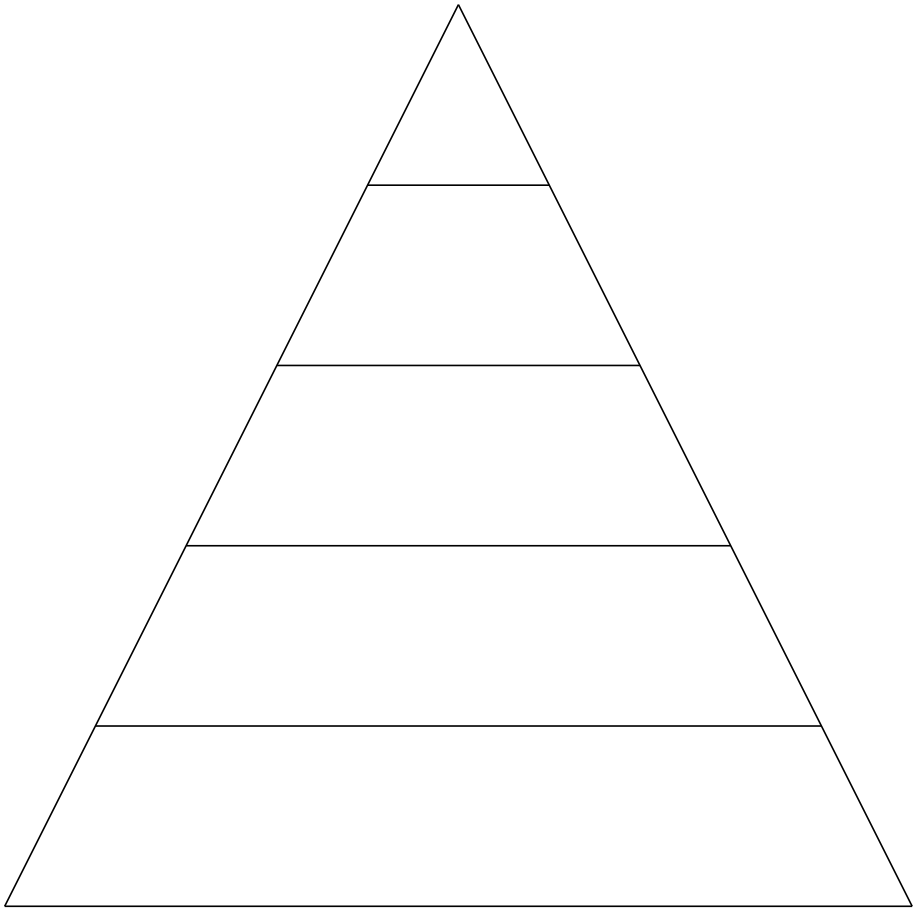
Auch die i-Schleife kann man mit der j-Schleife und/oder der k-Schleife vertauschen. Man erhält insgesamt 6 Varianten der Gauß-Elimination, die alle von der Zahl der Rechenoperationen her denselben Aufwand besitzen.

Sie können sich aber auf verschiedenen Plattformen unter verschiedenen Programmiersprachen sehr unterschiedlich verhalten.

Der Grund hierfür ist, dass Speicher von Rechnern hierarchisch aufgebaut sind, beginnend mit sehr langsamen, sehr großen, sehr billigen Magnetband-Speichern, über schnellere, kleinere, teurere Plattenspeicher, den noch schnelleren, noch kleineren, noch teureren Hauptspeicher, den schnellen, kleinen und teuren Cache und die sehr schnellen, sehr kleinen und sehr teuren Register der CPU. Arithmetische und logische Operationen können nur in den Registern ausgeführt werden.

Hierarchischer Speicher

schnell
& klein
& teuer



langsam
& gross
& billig

Modifikationen des Gaußschen Verfahrens

Daten können nur in den benachbarten Speicher bewegt werden, wobei der Datentransport zwischen den billigen Speicherarten sehr langsam geschieht, während der Transport zwischen den teuren Speichern an der Spitze der Hierarchie sehr schnell geschieht.

Modifikationen des Gaußschen Verfahrens

Daten können nur in den benachbarten Speicher bewegt werden, wobei der Datentransport zwischen den billigen Speicherarten sehr langsam geschieht, während der Transport zwischen den teuren Speichern an der Spitze der Hierarchie sehr schnell geschieht.

Insbesondere ist die Geschwindigkeit, mit der arithmetische Operationen ausgeführt werden, sehr viel höher als die Geschwindigkeit, mit der Daten transportiert werden.

Modifikationen des Gaußschen Verfahrens

Daten können nur in den benachbarten Speicher bewegt werden, wobei der Datentransport zwischen den billigen Speicherarten sehr langsam geschieht, während der Transport zwischen den teuren Speichern an der Spitze der Hierarchie sehr schnell geschieht.

Insbesondere ist die Geschwindigkeit, mit der arithmetische Operationen ausgeführt werden, sehr viel höher als die Geschwindigkeit, mit der Daten transportiert werden.

Faktoren zwischen 10 und 10000 (je nach Speicherebene) sind nicht ungewöhnlich. Algorithmen müssen also so gestaltet werden, dass der Datentransport zwischen verschiedenen Speicherebenen möglichst klein ist.

Modifikationen des Gaußschen Verfahrens

Die Reihenfolge der Schleifen im Gaußschen Eliminationsverfahren hat Einfluss auf die Laufzeit eines Programms.

Modifikationen des Gaußschen Verfahrens

Die Reihenfolge der Schleifen im Gaußschen Eliminationsverfahren hat Einfluss auf die Laufzeit eines Programms.

Eine Matrix wird in einem langen Array gespeichert. In der Sprache C geschieht das zeilenweise:

$$
\begin{array}{cccccccccc}
 & a(1,1) & \rightarrow & a(1,2) & \rightarrow & a(1,3) & \rightarrow & \dots & \rightarrow & a(1,n) \\
\rightarrow & a(2,1) & \rightarrow & a(2,2) & \rightarrow & a(2,3) & \rightarrow & \dots & \rightarrow & a(2,n) \\
\rightarrow & a(3,1) & \rightarrow & a(3,2) & \rightarrow & a(3,3) & \rightarrow & \dots & \rightarrow & a(3,n) \\
 & \vdots & & \vdots & & \vdots & & & & \vdots \\
\rightarrow & a(n,1) & \rightarrow & a(n,2) & \rightarrow & a(n,3) & \rightarrow & \dots & \rightarrow & a(n,n)
\end{array}
$$

Modifikationen des Gaußschen Verfahrens

Die Reihenfolge der Schleifen im Gaußschen Eliminationsverfahren hat Einfluss auf die Laufzeit eines Programms.

Eine Matrix wird in einem langen Array gespeichert. In der Sprache C geschieht das zeilenweise:

$$\begin{array}{cccccccccc}
 & a(1,1) & \to & a(1,2) & \to & a(1,3) & \to & \dots & \to & a(1,n) \\
\to & a(2,1) & \to & a(2,2) & \to & a(2,3) & \to & \dots & \to & a(2,n) \\
\to & a(3,1) & \to & a(3,2) & \to & a(3,3) & \to & \dots & \to & a(3,n) \\
 & \vdots & & \vdots & & \vdots & & & & \vdots \\
\to & a(n,1) & \to & a(n,2) & \to & a(n,3) & \to & \dots & \to & a(n,n)
\end{array}$$

In der zeilenorientierten Version in Algorithmus 4.8 werden in der innersten Schleife Daten verwendet, die im Speicher nahe beieinander liegen.

Modifikationen des Gaußschen Verfahrens

Die Reihenfolge der Schleifen im Gaußschen Eliminationsverfahren hat Einfluss auf die Laufzeit eines Programms.

Eine Matrix wird in einem langen Array gespeichert. In der Sprache C geschieht das zeilenweise:

$$\begin{array}{cccccccccc}
 & a(1,1) & \rightarrow & a(1,2) & \rightarrow & a(1,3) & \rightarrow & \dots & \rightarrow & a(1,n) \\
\rightarrow & a(2,1) & \rightarrow & a(2,2) & \rightarrow & a(2,3) & \rightarrow & \dots & \rightarrow & a(2,n) \\
\rightarrow & a(3,1) & \rightarrow & a(3,2) & \rightarrow & a(3,3) & \rightarrow & \dots & \rightarrow & a(3,n) \\
 & \vdots & & \vdots & & \vdots & & & & \vdots \\
\rightarrow & a(n,1) & \rightarrow & a(n,2) & \rightarrow & a(n,3) & \rightarrow & \dots & \rightarrow & a(n,n)
\end{array}$$

In der zeilenorientierten Version in Algorithmus 4.8 werden in der innersten Schleife Daten verwendet, die im Speicher nahe beieinander liegen.

Es sind daher nur wenige Datentransporte zwischen den verschiedenen Speicherebenen erforderlich.

Modifikationen des Gaußschen Verfahrens

Für die spaltenorientierte Version liegen die nacheinander benutzten Daten (für große n) sehr weit auseinander, und daher ist ein „cache miss“ (das benötigte Wort liegt nicht im Cache und es müssen zwei Blöcke zwischen dem Cache und dem Hauptspeicher ausgetauscht werden) oder sogar ein „page fault“ (das benötigte Wort liegt nicht einmal im Hauptspeicher und es müssen zwei Seiten zwischen dem Hauptspeicher und der Platte ausgetauscht werden) wahrscheinlicher.

Modifikationen des Gaußschen Verfahrens

Für die spaltenorientierte Version liegen die nacheinander benutzten Daten (für große n) sehr weit auseinander, und daher ist ein „cache miss“ (das benötigte Wort liegt nicht im Cache und es müssen zwei Blöcke zwischen dem Cache und dem Hauptspeicher ausgetauscht werden) oder sogar ein „page fault“ (das benötigte Wort liegt nicht einmal im Hauptspeicher und es müssen zwei Seiten zwischen dem Hauptspeicher und der Platte ausgetauscht werden) wahrscheinlicher.

In FORTRAN ist die Situation umgekehrt. Matrizen werden spaltenweise gespeichert, und für Algorithmus 4.9 ist die Datenlokalität hoch, während sie in Algorithmus 4.8 gering ist.

Modifikationen des Gaußschen Verfahrens

Um nicht für jeden neuen Rechner neue Software schreiben zu müssen, um die Modularität und Effizienz von Programmen zu erhöhen und um ihre Pflege zu erleichtern, wurden Standardoperationen der (numerischen) linearen Algebra, die „basic linear algebra subprograms“ (BLAS), definiert und es wurden Standardschnittstellen für ihren Aufruf festgelegt.

Modifikationen des Gaußschen Verfahrens

Um nicht für jeden neuen Rechner neue Software schreiben zu müssen, um die Modularität und Effizienz von Programmen zu erhöhen und um ihre Pflege zu erleichtern, wurden Standardoperationen der (numerischen) linearen Algebra, die „basic linear algebra subprograms“ (BLAS), definiert und es wurden Standardschnittstellen für ihren Aufruf festgelegt.

Diese werden (jedenfalls für Hochleistungsrechner) von den Herstellern auf Hardwareebene realisiert (nicht zuletzt, da die Hersteller wissen, dass die üblichen Benchmarktests Programme enthalten, die aus den BLAS-Routinen aufgebaut sind).

Modifikationen des Gaußschen Verfahrens

Die Benutzung der BLAS in eigenen Programmen bietet eine Reihe von Vorteilen:

- Die Robustheit von Berechnungen der linearen Algebra wird durch die BLAS erhöht, denn in Ihnen werden Details der Algorithmen und der Implementierung berücksichtigt, die bei der Programmierung eines Anwendungsproblems leicht übersehen werden, wie z.B. die Berücksichtigung von Overflow-Problemen.

Modifikationen des Gaußschen Verfahrens

Die Benutzung der BLAS in eigenen Programmen bietet eine Reihe von Vorteilen:

- Die Robustheit von Berechnungen der linearen Algebra wird durch die BLAS erhöht, denn in Ihnen werden Details der Algorithmen und der Implementierung berücksichtigt, die bei der Programmierung eines Anwendungsproblems leicht übersehen werden, wie z.B. die Berücksichtigung von Overflow-Problemen.
- Die Portabilität von Programmen wird erhöht, ohne auf Effizienz zu verzichten, denn es werden optimierte Versionen der BLAS auf Rechnern verwendet, für die diese existieren. Für alle anderen Plattformen existieren kompatible Standard-FORTRAN- oder C-Implementierungen.

Modifikationen des Gaußschen Verfahrens

Diese können als Public Domain Software bezogen werden von

http://www.netlib.org/blas/

bzw.

http://www.netlib.org/clapack/cblas/

Im ATLAS-Projekt („automatically tuned linear algebra software“) wurden und werden empirische Methoden entwickelt, um Bibliotheken hoher Performance zu erzeugen und zu pflegen, und so in der durch die Software diktierten Geschwindigkeit Schritt mit der Hardware-Entwicklung zu halten.

Modifikationen des Gaußschen Verfahrens

Es werden z.B. für den benutzten Rechner die Größen des Registers und Caches ermittelt und für die Level 2 und Level 3 BLAS die Blockgrößen angepasst. Die im ATLAS-Projekt entwickelten BLAS, für die es FORTRAN- und C-Interfaces gibt, können herunter geladen werden von

http://www.netlib.org/atlas/

Modifikationen des Gaußschen Verfahrens

Es werden z.B. für den benutzten Rechner die Größen des Registers und Caches ermittelt und für die Level 2 und Level 3 BLAS die Blockgrößen angepasst. Die im ATLAS-Projekt entwickelten BLAS, für die es FORTRAN- und C-Interfaces gibt, können herunter geladen werden von

http://www.netlib.org/atlas/

– Die Lesbarkeit von Programmen wird dadurch erhöht, dass mnemonische Namen für Standardoperationen verwendet werden und der Programmablauf nicht durch Codierungsdetails unterbrochen wird. Dies erleichtert auch die Dokumentation von Programmen.

Modifikationen des Gaußschen Verfahrens

Die erste Stufe der BLAS-Definitionen (Level 1 BLAS oder BLAS1) wurde 1979 durchgeführt und enthielt Vektor-Vektor-Operationen wie das Skalarprodukt oder die Summe eines Vektors und des Vielfachen eines weiteren Vektors.

Modifikationen des Gaußschen Verfahrens

Die erste Stufe der BLAS-Definitionen (Level 1 BLAS oder BLAS1) wurde 1979 durchgeführt und enthielt Vektor-Vektor-Operationen wie das Skalarprodukt oder die Summe eines Vektors und des Vielfachen eines weiteren Vektors.

Es wurde eine Namenskonvention eingeführt, die einen drei- bis fünfbuchstabigen Namen verwendet, dem ein Buchstabe zur Kennzeichnung des Datentyps vorangestellt wird (`S`, `D`, `C` oder `Z`).

Modifikationen des Gaußschen Verfahrens

Die erste Stufe der BLAS-Definitionen (Level 1 BLAS oder BLAS1) wurde 1979 durchgeführt und enthielt Vektor-Vektor-Operationen wie das Skalarprodukt oder die Summe eines Vektors und des Vielfachen eines weiteren Vektors.

Es wurde eine Namenskonvention eingeführt, die einen drei- bis fünfbuchstabigen Namen verwendet, dem ein Buchstabe zur Kennzeichnung des Datentyps vorangestellt wird (`S`, `D`, `C` oder `Z`).

Als Beispiele nennen wir nur die Function `_DOT`, für die durch

```
ALPHA=DDOT(N,X,1,Y,1)
```

das innere Produkt der doppeltgenauen Vektoren $\boldsymbol{x}$ und $\boldsymbol{y}$ der Dimension n berechnet werden,

Modifikationen des Gaußschen Verfahrens

oder die Subroutine _AXPY („a $\times$ plus y“) mit dem Aufruf

```
CAXPY(N,ALPHA,X,1,Y,1)
```

durch die für die komplexen Vektoren $\boldsymbol{x}$ und $\boldsymbol{y}$ der Dimension n der komplexe Vektor $\alpha\boldsymbol{x} + \boldsymbol{y}$ berechnet wird und im Speicher von $\boldsymbol{y}$ abgelegt wird.

Modifikationen des Gaußschen Verfahrens

oder die Subroutine _AXPY („a $\times$ plus y“) mit dem Aufruf

```
CAXPY(N,ALPHA,X,1,Y,1)
```

durch die für die komplexen Vektoren $\boldsymbol{x}$ und $\boldsymbol{y}$ der Dimension n der komplexe Vektor $\alpha\boldsymbol{x} + \boldsymbol{y}$ berechnet wird und im Speicher von $\boldsymbol{y}$ abgelegt wird.

Statt der Einsen in den Aufrufen kann man Inkremente angeben, so dass auch innere Produkte der Art

$$\sum_{j=0}^{n-1} a_{1+mj} a_{2+mj}$$

berechnet werden können, also bei spaltenweiser Speicherung einer (m, n)-Matrix $\boldsymbol{A}$ das innere Produkt der ersten und zweiten Zeile.

Modifikationen des Gaußschen Verfahrens

Beispiel 4.10: Als Beispiel betrachten wir die Bestimmung des inneren Produktes zweier Vektoren der Dimension $n = 10^7$.

Modifikationen des Gaußschen Verfahrens

Beispiel 4.10: Als Beispiel betrachten wir die Bestimmung des inneren Produktes zweier Vektoren der Dimension $n = 10^7$.

Wir verwenden (wie auch bei den folgenden Beispielen zur Performance der BLAS Routinen) eine HP 9000/785/C3000 Workstation mit einer CPU 2.0.PA8500 mit der Taktfrequenz 400 MHz und den FORTRAN 90 Compiler f90-HP.

Modifikationen des Gaußschen Verfahrens

Beispiel 4.10: Als Beispiel betrachten wir die Bestimmung des inneren Produktes zweier Vektoren der Dimension $n = 10^7$.

Wir verwenden (wie auch bei den folgenden Beispielen zur Performance der BLAS Routinen) eine HP 9000/785/C3000 Workstation mit einer CPU 2.0.PA8500 mit der Taktfrequenz 400 MHz und den FORTRAN 90 Compiler f90-HP.

Die folgende Tabelle enthält die Laufzeiten eines naiven Codes mit einer Laufanweisung, einer BLAS1 Implementierung in FORTRAN 77, die aus der netlib heruntergeladen werden kann, einer BLAS-Implementierung, die mit dem FORTRAN 90 Compiler von HP geliefert wird, der BLAS1-Routine aus dem ATLAS-Projekt und einer optimierten BLAS-Routine der veclib von HP.

Modifikationen des Gaußschen Verfahrens

Implementierung	CPU Zeit	Relation
naiv	0.92	7.9
BLAS (netlib)	0.52	4.5
BLAS (f90-HP)	0.29	2.4
BLAS (ATLAS)	0.23	2.0
veclib	0.12	1.0

Modifikationen des Gaußschen Verfahrens

Die BLAS1 haben zu effizienten Implementierungen von Algorithmen auf skalaren Maschinen geführt, auf Vektorrechnern oder Parallelrechnern werden weitere Standardoperationen benötigt. Man kann z.B. das Produkt einer Matrix $\boldsymbol{A}$ mit einem Vektor $\boldsymbol{x}$ codieren als

Modifikationen des Gaußschen Verfahrens

Die BLAS1 haben zu effizienten Implementierungen von Algorithmen auf skalaren Maschinen geführt, auf Vektorrechnern oder Parallelrechnern werden weitere Standardoperationen benötigt. Man kann z.B. das Produkt einer Matrix $\boldsymbol{A}$ mit einem Vektor $\boldsymbol{x}$ codieren als

Algorithmus 4.11: (Matrix-Vektor Produkt mit DAXPY)

```
Y = zeros(n,1);
for j = 1:n
   DAXPY(n,X(j),A(:,j),1,Y,1)
end
```

Modifikationen des Gaußschen Verfahrens

Die BLAS1 haben zu effizienten Implementierungen von Algorithmen auf skalaren Maschinen geführt, auf Vektorrechnern oder Parallelrechnern werden weitere Standardoperationen benötigt. Man kann z.B. das Produkt einer Matrix A mit einem Vektor x codieren als

Algorithmus 4.11: (Matrix-Vektor Produkt mit DAXPY)

```
Y = zeros(n,1);
for j = 1:n
   DAXPY(n,X(j),A(:,j),1,Y,1)
end
```

Dabei wird aber nicht berücksichtigt, dass der Ergebnisvektor y im Register gehalten werden könnte. BLAS1 hat also den Nachteil, dass zu wenige (nützliche) flops ausgeführt werden im Verhältnis zu (nutzlosen) Datenbewegungen.

Modifikationen des Gaußschen Verfahrens

Für die Ausführung eines _AXPYs sind z. B. $3n + 1$ Speicherzugriffe erforderlich (die Vektoren $\boldsymbol{x}$ und $\boldsymbol{y}$ und der Skalar α müssen gelesen werden, und der Ergebnisvektor $\boldsymbol{y}$ muss geschrieben werden) und es werden $2n$ flops ausgeführt. Das Verhältnis ist also $2/3$.

Modifikationen des Gaußschen Verfahrens

Für die Ausführung eines _AXPYs sind z. B. $3n + 1$ Speicherzugriffe erforderlich (die Vektoren $\boldsymbol{x}$ und $\boldsymbol{y}$ und der Skalar α müssen gelesen werden, und der Ergebnisvektor $\boldsymbol{y}$ muss geschrieben werden) und es werden $2n$ flops ausgeführt. Das Verhältnis ist also $2/3$.

Dies wurde 1988 verbessert mit der Definition den Level 2 BLAS oder BLAS2, die Matrix-Vektor-Operationen enthalten, wie z.B. die Subroutine _GEMV, durch die $\boldsymbol{y} \leftarrow \alpha \boldsymbol{A}\boldsymbol{x} + \beta \boldsymbol{y}$ berechnet wird, das Produkt einer Dreiecksmatrix mit einem Vektor, Rang-1- oder Rang-2-Aufdatierungen von Matrizen wie $\boldsymbol{A} \leftarrow \alpha \boldsymbol{x}\boldsymbol{y}^T + \boldsymbol{A}$, oder Lösungen von Gleichungssystemen mit Dreiecksmatrizen.

Modifikationen des Gaußschen Verfahrens

Für die Ausführung eines _AXPYs sind z. B. $3n + 1$ Speicherzugriffe erforderlich (die Vektoren $\boldsymbol{x}$ und $\boldsymbol{y}$ und der Skalar α müssen gelesen werden, und der Ergebnisvektor $\boldsymbol{y}$ muss geschrieben werden) und es werden $2n$ flops ausgeführt. Das Verhältnis ist also $2/3$.

Dies wurde 1988 verbessert mit der Definition den Level 2 BLAS oder BLAS2, die Matrix-Vektor-Operationen enthalten, wie z.B. die Subroutine _GEMV, durch die $\boldsymbol{y} \leftarrow \alpha \boldsymbol{A}\boldsymbol{x} + \beta \boldsymbol{y}$ berechnet wird, das Produkt einer Dreiecksmatrix mit einem Vektor, Rang-1- oder Rang-2-Aufdatierungen von Matrizen wie $\boldsymbol{A} \leftarrow \alpha \boldsymbol{x}\boldsymbol{y}^T + \boldsymbol{A}$, oder Lösungen von Gleichungssystemen mit Dreiecksmatrizen.

Für die Subroutine _GEMV sind im n-dimensionalen Fall $n^2 + 3n + 2$ Speicherzugriffe erforderlich, und es werden $2n^2$ flops ausgeführt. Das Verhältnis ist also 2.

Modifikationen des Gaußschen Verfahrens

Algorithmus 4.9 ist schon fast eine BLAS2 Implementierung der LR-Zerlegung. Man muss nur noch die Modifikationen der Spalten zu einer Rang-1-Modifikation des unteren $(n-i, n-i)$-Blocks umschreiben.

Algorithmus 4.12: (LR-Zerlegung; BLAS2)

```
for i=1: n-1
   a(i+1:n,i)=a(i+1:n,i)/a(i,i);
   a(i+1:n,i+1:n)=a(i+1:n,i+1:n)-a(i+1:m,i)*a(i,i+1:n);
end
```

Modifikationen des Gaußschen Verfahrens

Beispiel 4.13: Als Beispiel zur Performance der BLAS2 betrachten wir die Bestimmung des Produktes einer $(10^4, 10^3)$-Matrix mit einem Vektor. Hierfür erhält man die Laufzeiten

Implementierung	CPU Zeit	Relation
naiv	4.32	61.7
BLAS2 (netlib)	1.39	19.9
BLAS2 (f90-HP)	0.62	8.9
BLAS2 (ATLAS)	0.17	2.4
veclib	0.07	1.0

□

Modifikationen des Gaußschen Verfahrens

In die Level 3 BLAS oder BLAS3 wurden schließlich Matrix-Matrix-Operationen wie

$$\boldsymbol{C} \leftarrow \alpha \boldsymbol{A}\boldsymbol{B} + \beta \boldsymbol{C}$$

aufgenommen.

Modifikationen des Gaußschen Verfahrens

In die Level 3 BLAS oder BLAS3 wurden schließlich Matrix-Matrix-Operationen wie

$$\boldsymbol{C} \leftarrow \alpha \boldsymbol{A}\boldsymbol{B} + \beta \boldsymbol{C}$$

aufgenommen.

Um die Wiederverwendung von Daten in den Registern oder im Cache in möglichst hohem Maße zu erreichen, werden die beteiligten Matrizen in Blöcke unterteilt und die Operationen blockweise ausgeführt.

Modifikationen des Gaußschen Verfahrens

In die Level 3 BLAS oder BLAS3 wurden schließlich Matrix-Matrix-Operationen wie

$$\boldsymbol{C} \leftarrow \alpha \boldsymbol{A}\boldsymbol{B} + \beta \boldsymbol{C}$$

aufgenommen.

Um die Wiederverwendung von Daten in den Registern oder im Cache in möglichst hohem Maße zu erreichen, werden die beteiligten Matrizen in Blöcke unterteilt und die Operationen blockweise ausgeführt.

Auf diese Weise erreicht man, dass für die obige Operation bei $4n^2 + 2$ Speicherzugriffen $2n^3 + O(n^2)$ flops ausgeführt werden, das Verhältnis von nützlicher Arbeit zu Speicherzugriffen also auf $n/2$ steigt.

Modifikationen des Gaußschen Verfahrens

Beispiel 4.14: Als Beispiel zur Performance der BLAS3 betrachten wir die Bestimmung des Produktes zweier $(10^3, 10^3)$ Matrizen. Hierfür erhält man die Laufzeiten

Implementierung	CPU Zeit	Relation
naiv	462.42	247.34
BLAS3 (netlib)	320.00	171.1
BLAS3 (f90-HP)	7.25	3.9
BLAS3 (ATLAS)	4.26	2.3
veclib	1.87	1.0

□

Störungen linearer Systeme

Wir wollen nun den Begriff der Kondition einer Matrix einführen, deren Wert — wie oben angedeutet — verantwortlich ist für die numerische Behandelbarkeit eines linearen Gleichungssystems.

Störungen linearer Systeme

Wir wollen nun den Begriff der Kondition einer Matrix einführen, deren Wert — wie oben angedeutet — verantwortlich ist für die numerische Behandelbarkeit eines linearen Gleichungssystems.

Wir betrachten dazu neben dem linearen Gleichungssystem

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{4.4}$$

mit der regulären Matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$

Störungen linearer Systeme

Wir wollen nun den Begriff der Kondition einer Matrix einführen, deren Wert — wie oben angedeutet — verantwortlich ist für die numerische Behandelbarkeit eines linearen Gleichungssystems.

Wir betrachten dazu neben dem linearen Gleichungssystem

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{4.4}$$

mit der regulären Matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ ein gestörtes System

$$(\boldsymbol{A} + \Delta\boldsymbol{A})(\boldsymbol{x} + \Delta\boldsymbol{x}) = \boldsymbol{b} + \Delta\boldsymbol{b} \tag{4.5}$$

und fragen uns, wie die Lösung des ursprünglichen Systems auf diese Störungen reagiert.

Störungen linearer Systeme

Bemerkung 4.15: Kleine Störungen kann man bei der praktischen Lösung linearer Gleichungssysteme grundsätzlich nicht ausschließen.

Störungen linearer Systeme

Bemerkung 4.15: Kleine Störungen kann man bei der praktischen Lösung linearer Gleichungssysteme grundsätzlich nicht ausschließen.

Einerseits können die Eingangsdaten des Systems aus Messungen herrühren und somit a priori fehlerbehaftet sein. Andererseits wird man bei der Benutzung eines elektronischen Rechners durch die endliche Genauigkeit der Zahlendarstellungen auf dem Rechner Eingabefehler machen müssen.

Störungen linearer Systeme

Bemerkung 4.15: Kleine Störungen kann man bei der praktischen Lösung linearer Gleichungssysteme grundsätzlich nicht ausschließen.

Einerseits können die Eingangsdaten des Systems aus Messungen herrühren und somit a priori fehlerbehaftet sein. Andererseits wird man bei der Benutzung eines elektronischen Rechners durch die endliche Genauigkeit der Zahlendarstellungen auf dem Rechner Eingabefehler machen müssen.

Man muss also grundsätzlich davon ausgehen, dass man mit gestörten Systemen anstelle der wirklich zu lösenden Systeme rechnen muss. Allerdings kann man meistens annehmen, dass die Störungen klein sind. □

Störungen linearer Systeme

Bevor wir die Wirkung von Störungen untersuchen können, benötigen wir noch einige Aussagen über sogenannte Matrixnormen, durch die wir die Größe einer Matrix messen.

Störungen linearer Systeme

Bevor wir die Wirkung von Störungen untersuchen können, benötigen wir noch einige Aussagen über sogenannte Matrixnormen, durch die wir die Größe einer Matrix messen.

Definition 4.16: Es sei $A \in \mathbb{C}^{(m \times n)}$ eine Matrix, $\|\cdot\|_n$ eine Vektornorm auf $\mathbb{C}^n$ und $\|\cdot\|_m$ eine Vektornorm auf $\mathbb{C}^m$.

Störungen linearer Systeme

Bevor wir die Wirkung von Störungen untersuchen können, benötigen wir noch einige Aussagen über sogenannte Matrixnormen, durch die wir die Größe einer Matrix messen.

Definition 4.16: Es sei $\boldsymbol{A} \in \mathbb{C}^{(m \times n)}$ eine Matrix, $\|\cdot\|_n$ eine Vektornorm auf $\mathbb{C}^n$ und $\|\cdot\|_m$ eine Vektornorm auf $\mathbb{C}^m$.

Dann heißt

$$\|\boldsymbol{A}\|_{m,n} := \max_{\boldsymbol{x} \neq \boldsymbol{o}} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_m}{\|\boldsymbol{x}\|_n}$$

die Matrixnorm von $\boldsymbol{A}$, die den Normen $\|\cdot\|_n$ und $\|\cdot\|_m$ zugeordnet ist.

Störungen linearer Systeme

Bevor wir die Wirkung von Störungen untersuchen können, benötigen wir noch einige Aussagen über sogenannte Matrixnormen, durch die wir die Größe einer Matrix messen.

Definition 4.16: Es sei $\boldsymbol{A} \in \mathbb{C}^{(m \times n)}$ eine Matrix, $\|\cdot\|_n$ eine Vektornorm auf $\mathbb{C}^n$ und $\|\cdot\|_m$ eine Vektornorm auf $\mathbb{C}^m$.

Dann heißt

$$\|\boldsymbol{A}\|_{m,n} := \max_{\boldsymbol{x} \neq \boldsymbol{o}} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_m}{\|\boldsymbol{x}\|_n}$$

die Matrixnorm von $\boldsymbol{A}$, die den Normen $\|\cdot\|_n$ und $\|\cdot\|_m$ zugeordnet ist.

Eine zugeordnete Matrixnorm „misst“ also die maximale Verlängerung, die ein Vektor $\boldsymbol{x}$ durch Anwendung von $\boldsymbol{A}$ erfahren kann.

Störungen linearer Systeme

Bemerkung 4.17: Wegen

$$\frac{\|\boldsymbol{A}\boldsymbol{x}\|_m}{\|\boldsymbol{x}\|_n} = \left\| \boldsymbol{A} \left(\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_n} \right) \right\|_m$$

Störungen linearer Systeme

Bemerkung 4.17: Wegen

$$\frac{\|\boldsymbol{A}\boldsymbol{x}\|_m}{\|\boldsymbol{x}\|_n} = \left\| \boldsymbol{A}\left(\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_n}\right)\right\|_m$$

genügt es, das Maximum über Vektoren der Länge 1 zu erstrecken:

$$\|\boldsymbol{A}\|_{m,n} = \max\{\|\boldsymbol{A}\boldsymbol{y}\|_m \ : \ \|\boldsymbol{y}\|_n = 1\}.$$

Störungen linearer Systeme

Bemerkung 4.17: Wegen

$$\frac{\|\boldsymbol{A}\boldsymbol{x}\|_m}{\|\boldsymbol{x}\|_n} = \left\| \boldsymbol{A}\left(\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_n}\right)\right\|_m$$

genügt es, das Maximum über Vektoren der Länge 1 zu erstrecken:

$$\|\boldsymbol{A}\|_{m,n} = \max\{\|\boldsymbol{A}\boldsymbol{y}\|_m \ : \ \|\boldsymbol{y}\|_n = 1\}.$$

Die Existenz des Maximums (daß es also einen Vektor $\boldsymbol{x}$ mit $\|\boldsymbol{x}\|_n = 1$ und $\|\boldsymbol{A}\boldsymbol{x}\|_m = \|\boldsymbol{A}\|_{m,n}$ gibt) folgt aus der Stetigkeit der Abbildung $\boldsymbol{x} \mapsto \|\boldsymbol{A}\boldsymbol{x}\|$. □

Störungen linearer Systeme

Bemerkung 4.18: $\|\cdot\|_{m,n}$ ist eine Norm auf $\mathbb{C}^{(m\times n)}$, denn

$$\begin{aligned}\|\boldsymbol{A}\|_{m,n} = 0 &\Leftrightarrow \|\boldsymbol{A}\boldsymbol{x}\|_m = 0 \quad \forall \boldsymbol{x} \in \mathbb{C}^n \\ &\Leftrightarrow \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0} \quad \forall \boldsymbol{x} \in \mathbb{C}^n \\ &\Leftrightarrow \boldsymbol{A} = \boldsymbol{O},\end{aligned}$$

$$\begin{aligned}\|\lambda\boldsymbol{A}\|_{m,n} &= \max\{\|\lambda\boldsymbol{A}\boldsymbol{x}\|_m \,:\, \|\boldsymbol{x}\|_n = 1\} \\ &= \max\{|\lambda| \cdot \|\boldsymbol{A}\boldsymbol{x}\|_m \,:\, \|\boldsymbol{x}\|_n = 1\} \\ &= |\lambda| \cdot \|\boldsymbol{A}\|_{m,n},\end{aligned}$$

$$\begin{aligned}\|\boldsymbol{A}+\boldsymbol{B}\|_{m,n} &= \max\{\|\boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{x}\|_m \,:\, \|\boldsymbol{x}\|_n = 1\} \\ &\leqslant \max\{\|\boldsymbol{A}\boldsymbol{x}\|_m + \|\boldsymbol{B}\boldsymbol{x}\|_m \,:\, \|\boldsymbol{x}\|_n = 1\} \\ &\leqslant \max\{\|\boldsymbol{A}\boldsymbol{x}\|_m \,:\, \|\boldsymbol{x}\|_n = 1\} + \max\{\|\boldsymbol{B}\boldsymbol{x}\|_m \,:\, \|\boldsymbol{x}\|_n = 1\} \\ &= \|\boldsymbol{A}\|_{m,n} + \|\boldsymbol{B}\|_{m,n}.\end{aligned}$$

Störungen linearer Systeme

Diese ist zusätzlich submultiplikativ im folgenden Sinne:

$$\|\boldsymbol{A}\boldsymbol{x}\|_m \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{x}\|_n \quad \forall\, \boldsymbol{x} \in \mathbb{C}^n, \boldsymbol{A} \in \mathbb{C}^{(m\times n)}, \tag{4.6}$$

$$\|\boldsymbol{A}\boldsymbol{B}\|_{m,p} \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{B}\|_{n,p} \quad \forall\, \boldsymbol{A} \in \mathbb{C}^{(m\times n)}, \boldsymbol{B} \in \mathbb{C}^{(n,p)}. \tag{4.7}$$

Störungen linearer Systeme

Diese ist zusätzlich submultiplikativ im folgenden Sinne:

$$\|\boldsymbol{A}\boldsymbol{x}\|_m \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{x}\|_n \quad \forall\, \boldsymbol{x} \in \mathbb{C}^n, \boldsymbol{A} \in \mathbb{C}^{(m\times n)}, \tag{4.6}$$

$$\|\boldsymbol{A}\boldsymbol{B}\|_{m,p} \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{B}\|_{n,p} \quad \forall\, \boldsymbol{A} \in \mathbb{C}^{(m\times n)}, \boldsymbol{B} \in \mathbb{C}^{(n,p)}. \tag{4.7}$$

Die Ungleichung (4.6) folgt sofort aus der Definition 4.16; denn es ist

$$\|\boldsymbol{A}\|_{m,n} \geqslant \frac{\|\boldsymbol{A}\boldsymbol{x}\|_m}{\|\boldsymbol{x}\|_n}$$

für alle $\boldsymbol{x} \in \mathbb{C}^n$.

Störungen linearer Systeme

Diese ist zusätzlich submultiplikativ im folgenden Sinne:

$$\|\boldsymbol{A}\boldsymbol{x}\|_m \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{x}\|_n \quad \forall\, \boldsymbol{x} \in \mathbb{C}^n, \boldsymbol{A} \in \mathbb{C}^{(m\times n)}, \tag{4.6}$$

$$\|\boldsymbol{A}\boldsymbol{B}\|_{m,p} \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{B}\|_{n,p} \quad \forall\, \boldsymbol{A} \in \mathbb{C}^{(m\times n)}, \boldsymbol{B} \in \mathbb{C}^{(n,p)}. \tag{4.7}$$

Die Ungleichung (4.6) folgt sofort aus der Definition 4.16; denn es ist

$$\|\boldsymbol{A}\|_{m,n} \geqslant \frac{\|\boldsymbol{A}\boldsymbol{x}\|_m}{\|\boldsymbol{x}\|_n}$$

für alle $\boldsymbol{x} \in \mathbb{C}^n$.

Die Ungleichung (4.7) erschließt man mit (4.6) wie folgt: Für alle $\boldsymbol{x} \in \mathbb{C}^p$ ist

$$\|\boldsymbol{A}\boldsymbol{B}\boldsymbol{x}\|_m = \|\boldsymbol{A}(\boldsymbol{B}\boldsymbol{x})\|_m \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{B}\boldsymbol{x}\|_n \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{B}\|_{n,p} \cdot \|\boldsymbol{x}\|_p,$$

Störungen linearer Systeme

Diese ist zusätzlich submultiplikativ im folgenden Sinne:

$$\|\boldsymbol{A}\boldsymbol{x}\|_m \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{x}\|_n \quad \forall\, \boldsymbol{x} \in \mathbb{C}^n, \boldsymbol{A} \in \mathbb{C}^{(m\times n)}, \tag{4.6}$$

$$\|\boldsymbol{A}\boldsymbol{B}\|_{m,p} \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{B}\|_{n,p} \quad \forall\, \boldsymbol{A} \in \mathbb{C}^{(m\times n)}, \boldsymbol{B} \in \mathbb{C}^{(n,p)}. \tag{4.7}$$

Die Ungleichung (4.6) folgt sofort aus der Definition 4.16; denn es ist

$$\|\boldsymbol{A}\|_{m,n} \geqslant \frac{\|\boldsymbol{A}\boldsymbol{x}\|_m}{\|\boldsymbol{x}\|_n}$$

für alle $\boldsymbol{x} \in \mathbb{C}^n$.

Die Ungleichung (4.7) erschließt man mit (4.6) wie folgt: Für alle $\boldsymbol{x} \in \mathbb{C}^p$ ist

$$\|\boldsymbol{A}\boldsymbol{B}\boldsymbol{x}\|_m = \|\boldsymbol{A}(\boldsymbol{B}\boldsymbol{x})\|_m \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{B}\boldsymbol{x}\|_n \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{B}\|_{n,p} \cdot \|\boldsymbol{x}\|_p,$$

Also ist $\|\boldsymbol{A}\boldsymbol{B}\|_{m,p} = \max\{\|\boldsymbol{A}\boldsymbol{B}\boldsymbol{x}\|_m \,:\, \|\boldsymbol{x}\|_p = 1\} \leqslant \|\boldsymbol{A}\|_{m,n} \cdot \|\boldsymbol{B}\|_{n,p}$. □

Störungen linearer Systeme

Bemerkung 4.19: Die Matrixnorm $\|\boldsymbol{A}\|_{m,n}$ ist die kleinste nichtnegative reelle Zahl μ, mit der

$$\|\boldsymbol{A}\boldsymbol{x}\|_m \leqslant \mu \cdot \|\boldsymbol{x}\|_n \quad \forall \boldsymbol{x} \in \mathbb{C}^n$$

ist. $\|\boldsymbol{A}\|_{m,n}$ ist demnach die maximale Verlängerung, die ein $\boldsymbol{x}$ durch Abbildung mit $\boldsymbol{A}$ erfahren kann, wobei $\boldsymbol{x}$ selbst in der $\|\cdot\|_n$-Norm und $\boldsymbol{A}\boldsymbol{x}$ in der Norm $\|\cdot\|_m$ des Bildraumes gemessen wird. □

Störungen linearer Systeme

Bemerkung 4.19: Die Matrixnorm $\|\boldsymbol{A}\|_{m,n}$ ist die kleinste nichtnegative reelle Zahl μ, mit der

$$\|\boldsymbol{A}\boldsymbol{x}\|_m \leqslant \mu \cdot \|\boldsymbol{x}\|_n \quad \forall \boldsymbol{x} \in \mathbb{C}^n$$

ist. $\|\boldsymbol{A}\|_{m,n}$ ist demnach die maximale Verlängerung, die ein $\boldsymbol{x}$ durch Abbildung mit $\boldsymbol{A}$ erfahren kann, wobei $\boldsymbol{x}$ selbst in der $\|\cdot\|_n$-Norm und $\boldsymbol{A}\boldsymbol{x}$ in der Norm $\|\cdot\|_m$ des Bildraumes gemessen wird. □

Wir betrachten nun nur noch den Fall, daß im Urbildraum und im Bildraum dieselbe Norm verwendet wird, auch wenn beide Räume verschiedene Dimension haben, und verwenden für die Matrixnorm dasselbe Symbol wie für die Vektornorm.

Störungen linearer Systeme

Bemerkung 4.19: Die Matrixnorm $\|A\|_{m,n}$ ist die kleinste nichtnegative reelle Zahl μ, mit der

$$\|Ax\|_m \leqslant \mu \cdot \|x\|_n \quad \forall x \in \mathbb{C}^n$$

ist. $\|A\|_{m,n}$ ist demnach die maximale Verlängerung, die ein x durch Abbildung mit A erfahren kann, wobei x selbst in der $\|\cdot\|_n$-Norm und Ax in der Norm $\|\cdot\|_m$ des Bildraumes gemessen wird. □

Wir betrachten nun nur noch den Fall, daß im Urbildraum und im Bildraum dieselbe Norm verwendet wird, auch wenn beide Räume verschiedene Dimension haben, und verwenden für die Matrixnorm dasselbe Symbol wie für die Vektornorm.

Für $A \in \mathbb{R}^{(5,9)}$ bezeichnet also $\|A\|_\infty$ die Matrixnorm von A gemäß Definition 4.16, wenn sowohl im Urbildraum $\mathbb{R}^9$ als auch im Bildraum $\mathbb{R}^5$ die Maximumnorm verwendet wird.

Störungen linearer Systeme

Satz 4.20

Es sei $\boldsymbol{A} \in \mathbb{C}^{(m\times n)}$ gegeben.

- Die Zeilensummennorm

$$\|\boldsymbol{A}\|_\infty := \max_{i=1,\ldots,m} \sum_{j=1}^{n} |a_{ij}|$$

ist der Maximumnorm $\|\boldsymbol{x}\|_\infty := \max\limits_{i=1,\ldots,n} |x_i|$ zugeordnet.

- Die Spektralnorm

$$\|\boldsymbol{A}\|_2 := \max\{\sqrt{\lambda} : \lambda \quad \text{ist Eigenwert von} \quad \boldsymbol{A}^H\boldsymbol{A}\}$$

ist der Euklidischen Norm zugeordnet.

Störungen linearer Systeme

Satz 4.20 (Fortsetzung)

- Die Spaltensummennorm

$$\|\boldsymbol{A}\|_1 := \max_{j=1,\ldots,n} \sum_{i=1}^{m} |a_{ij}|$$

ist der Summennorm $\|\boldsymbol{x}\|_1 := \sum_{i=1}^{n} |x_i|$ zugeordnet.

Störungen linearer Systeme

Satz 4.20 (Fortsetzung)

- Die Spaltensummennorm

$$\|A\|_1 := \max_{j=1,\ldots,n} \sum_{i=1}^{m} |a_{ij}|$$

ist der Summennorm $\|x\|_1 := \sum_{i=1}^{n} |x_i|$ zugeordnet.

Beweis.

Zeilensummennorm: Für alle $x \in \mathbb{C}^n$ gilt

$$\begin{aligned}\|Ax\|_\infty &= \max_{i=1,\ldots,m} |\sum_{j=1}^{n} a_{ij}x_j| \leqslant \max_{i=1,\ldots,m} \sum_{j=1}^{n} |a_{ij}| \cdot |x_j| \\ &\leqslant \|x\|_\infty \cdot \max_{i=1,\ldots,m} \sum_{j=1}^{n} |a_{ij}|,\end{aligned}$$

Störungen linearer Systeme

Beweis.

und daher

$$\|\boldsymbol{A}\|_\infty \leqslant \max_{i=1,\ldots,m} \sum_{j=1}^{n} |a_{ij}|. \tag{4.8}$$

Störungen linearer Systeme

Beweis.

und daher

$$\|\boldsymbol{A}\|_\infty \leqslant \max_{i=1,\ldots,m} \sum_{j=1}^{n} |a_{ij}|. \tag{4.8}$$

Sei $k \in \{1, \ldots, m\}$ *mit*

$$\sum_{j=1}^{n} |a_{ij}| \leqslant \sum_{j=1}^{n} |a_{kj}|$$

für alle i.

Störungen linearer Systeme

Beweis.

und daher

$$\|\boldsymbol{A}\|_\infty \leqslant \max_{i=1,\ldots,m} \sum_{j=1}^{n} |a_{ij}|. \tag{4.8}$$

Sei $k \in \{1, \ldots, m\}$ *mit*

$$\sum_{j=1}^{n} |a_{ij}| \leqslant \sum_{j=1}^{n} |a_{kj}|$$

für alle i.

Wir definieren $\boldsymbol{x} \in \mathbb{C}^n$ *durch* $x_j := 1$, *falls* $a_{kj} = 0$, *und* $x_j := \overline{a_{kj}}/|a_{kj}|$, *sonst.*

Störungen linearer Systeme

Beweis.

Dann gilt $\|\boldsymbol{x}\|_\infty = 1$ *und*

Störungen linearer Systeme

Beweis.

Dann gilt $\|\boldsymbol{x}\|_\infty = 1$ *und*

$$\begin{aligned}\|\boldsymbol{Ax}\|_\infty &= \max_{i=1,\dots,m} |\sum_{j=1}^{n} a_{ij}x_j| \geqslant |\sum_{j=1}^{n} a_{kj}x_j| \\ &= \left|\sum_{j=1}^{n} |a_{kj}|\right| = \max_{i=1,\dots,m} \sum_{j=1}^{n} |a_{ij}|,\end{aligned}$$

Störungen linearer Systeme

Beweis.

Dann gilt $\|\boldsymbol{x}\|_\infty = 1$ *und*

$$\begin{aligned}\|\boldsymbol{Ax}\|_\infty &= \max_{i=1,\dots,m} |\sum_{j=1}^{n} a_{ij}x_j| \geqslant |\sum_{j=1}^{n} a_{kj}x_j| \\ &= \left|\sum_{j=1}^{n} |a_{kj}|\right| = \max_{i=1,\dots,m} \sum_{j=1}^{n} |a_{ij}|,\end{aligned}$$

und daher

$$\begin{aligned}\|\boldsymbol{A}\|_\infty &= \max\{\|\boldsymbol{Ay}\|_\infty \,:\, \|\boldsymbol{y}\|_\infty = 1\} \\ &\geqslant \|\boldsymbol{Ax}\|_\infty \geqslant \max_{i=1,\dots,m} \sum_{j=1}^{n} |a_{ij}|,\end{aligned}$$

zusammen mit (4.8) *also die Behauptung.*

Störungen linearer Systeme

Beweis.

Spektralnorm: Es ist $A^H A \in \mathbb{C}^{(n \times n)}$ eine Hermitesche Matrix, und daher ist nach dem Rayleighschen Prinzip

Störungen linearer Systeme

Beweis.

Spektralnorm: Es ist $\boldsymbol{A}^H\boldsymbol{A} \in \mathbb{C}^{(n\times n)}$ eine Hermitesche Matrix, und daher ist nach dem Rayleighschen Prinzip

$$\max_{\boldsymbol{x}\neq\boldsymbol{o}} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_2^2}{\|\boldsymbol{x}\|_2^2} = \max_{\boldsymbol{x}\neq\boldsymbol{o}} \frac{\boldsymbol{x}^H\boldsymbol{A}^H\boldsymbol{A}\boldsymbol{x}}{\boldsymbol{x}^H\boldsymbol{x}}$$

der maximale Eigenwert von $\boldsymbol{A}^H\boldsymbol{A}$.

Störungen linearer Systeme

Beweis.

Spektralnorm: Es ist $\boldsymbol{A}^H\boldsymbol{A} \in \mathbb{C}^{(n\times n)}$ eine Hermitesche Matrix, und daher ist nach dem Rayleighschen Prinzip

$$\max_{\boldsymbol{x}\neq\boldsymbol{o}} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_2^2}{\|\boldsymbol{x}\|_2^2} = \max_{\boldsymbol{x}\neq\boldsymbol{o}} \frac{\boldsymbol{x}^H\boldsymbol{A}^H\boldsymbol{A}\boldsymbol{x}}{\boldsymbol{x}^H\boldsymbol{x}}$$

der maximale Eigenwert von $\boldsymbol{A}^H\boldsymbol{A}$.

Spaltensummennorm: Zeigt man ähnlich wie bei der Zeilensummennorm. □

Störungen linearer Systeme

Beispiel 4.21: Sei

$$\boldsymbol{A} = \begin{pmatrix} 1 & 0.1 & -0.1 \\ 0.1 & 2 & -0.4 \\ 0.2 & 0.4 & 3 \end{pmatrix}.$$

Störungen linearer Systeme

Beispiel 4.21: Sei

$$\boldsymbol{A} = \begin{pmatrix} 1 & 0.1 & -0.1 \\ 0.1 & 2 & -0.4 \\ 0.2 & 0.4 & 3 \end{pmatrix}.$$

Es ist

$$\|\boldsymbol{A}\|_\infty = \max\{1.2, 2.5, 3.6\} = 3.6$$

$$\|\boldsymbol{A}\|_1 = \max\{1.3, 2.5, 3.5\} = 3.5$$

Störungen linearer Systeme

Die Spektralnorm von $\boldsymbol{A}$ ist die Quadratwurzel aus dem maximalen Eigenwert von

$$\boldsymbol{A}^T\boldsymbol{A} = \begin{pmatrix} 1.05 & 0.38 & 0.46 \\ 0.38 & 4.17 & 0.39 \\ 0.46 & 0.39 & 9.17 \end{pmatrix}.$$

Störungen linearer Systeme

Die Spektralnorm von $\boldsymbol{A}$ ist die Quadratwurzel aus dem maximalen Eigenwert von

$$\boldsymbol{A}^T\boldsymbol{A} = \begin{pmatrix} 1.05 & 0.38 & 0.46 \\ 0.38 & 4.17 & 0.39 \\ 0.46 & 0.39 & 9.17 \end{pmatrix}.$$

Nach einiger Rechnung ergibt dies

$$\|\boldsymbol{A}\|_2 \approx \sqrt{9.2294} \approx 3.04. \qquad \square$$

Störungen linearer Systeme

Die Spektralnorm von $\boldsymbol{A}$ ist die Quadratwurzel aus dem maximalen Eigenwert von

$$\boldsymbol{A}^T\boldsymbol{A} = \begin{pmatrix} 1.05 & 0.38 & 0.46 \\ 0.38 & 4.17 & 0.39 \\ 0.46 & 0.39 & 9.17 \end{pmatrix}.$$

Nach einiger Rechnung ergibt dies

$$\|\boldsymbol{A}\|_2 \approx \sqrt{9.2294} \approx 3.04. \qquad \square$$

Die Spektralnorm ist die genaueste (im Sinne der Euklidischen Geometrie des $\mathbb{C}^n$) und teuerste Norm (im Sinne der Berechenbarkeit) der drei vorgestellten Normen.

Störungen linearer Systeme

Die Spektralnorm von $\boldsymbol{A}$ ist die Quadratwurzel aus dem maximalen Eigenwert von

$$\boldsymbol{A}^T\boldsymbol{A} = \begin{pmatrix} 1.05 & 0.38 & 0.46 \\ 0.38 & 4.17 & 0.39 \\ 0.46 & 0.39 & 9.17 \end{pmatrix}.$$

Nach einiger Rechnung ergibt dies

$$\|\boldsymbol{A}\|_2 \approx \sqrt{9.2294} \approx 3.04. \qquad \square$$

Die Spektralnorm ist die genaueste (im Sinne der Euklidischen Geometrie des $\mathbb{C}^n$) und teuerste Norm (im Sinne der Berechenbarkeit) der drei vorgestellten Normen.

Wir werden noch eine bessere Deutung der Spektralnorm (mittels der sogenannten SVD) kennenlernen, aber oft begnügt man sich mit einer billigeren verwandten Norm, der Schur-Norm.

Störungen linearer Systeme

Satz 4.22

Für alle $\boldsymbol{A} \in \mathbb{C}^{(m \times n)}$ gilt

$$\|\boldsymbol{A}\|_2 \leqslant \|\boldsymbol{A}\|_S := \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2}.$$

Störungen linearer Systeme

Satz 4.22

Für alle $A \in \mathbb{C}^{(m\times n)}$ gilt

$$\|A\|_2 \leqslant \|A\|_S := \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2}.$$

Bemerkung 4.23: $\|A\|_S$ heißt die Schur-Norm oder Erhard-Schmidt-Norm (oder — speziell in der anglo-amerikanischen Literatur — Frobenius-Norm der Matrix A).

Störungen linearer Systeme

Satz 4.22

Für alle $A \in \mathbb{C}^{(m\times n)}$ gilt

$$\|A\|_2 \leqslant \|A\|_S := \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2}.$$

Bemerkung 4.23: $\|A\|_S$ heißt die Schur-Norm oder Erhard-Schmidt-Norm (oder — speziell in der anglo-amerikanischen Literatur — Frobenius-Norm der Matrix A). Diese ist zwar eine Vektornorm auf dem $\mathbb{C}^{(m\times n)}$ (die Euklidische Norm auf dem $\mathbb{C}^{m\cdot n}$), aber keine einer Vektornorm zugeordnete Matrixnorm, denn für eine solche gilt im Fall $n = m$ stets

$$\|E\| = \max\{\|Ex\| \,:\, \|x\| = 1\} = 1.$$

Es ist aber $\|E\|_S = \sqrt{n}$. □

Störungen linearer Systeme

Beweis.

Wegen der Cauchy-Schwarzschen Ungleichung gilt für alle $\boldsymbol{x} \in \mathbb{C}^n$

$$\|\boldsymbol{A}\boldsymbol{x}\|_2^2 = \sum_{i=1}^{m} \left| \sum_{j=1}^{n} a_{ij} x_j \right|^2 \leqslant \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right) \sum_{j=1}^{n} |x_j|^2 = \|\boldsymbol{A}\|_S^2 \cdot \|\boldsymbol{x}\|_2^2,$$

und daher gilt $\|\boldsymbol{A}\|_2 \leqslant \|\boldsymbol{A}\|_S$. □

Störungen linearer Systeme

Beweis.

Wegen der Cauchy-Schwarzschen Ungleichung gilt für alle $\boldsymbol{x} \in \mathbb{C}^n$

$$\|\boldsymbol{A}\boldsymbol{x}\|_2^2 = \sum_{i=1}^{m} \left| \sum_{j=1}^{n} a_{ij} x_j \right|^2 \leqslant \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right) \sum_{j=1}^{n} |x_j|^2 = \|\boldsymbol{A}\|_S^2 \cdot \|\boldsymbol{x}\|_2^2,$$

und daher gilt $\|\boldsymbol{A}\|_2 \leqslant \|\boldsymbol{A}\|_S$. □

Beispiel 4.24: Für die Matrix $\boldsymbol{A}$ aus Beispiel 4.21 erhält man

$$\|\boldsymbol{A}\|_2 \leqslant \|\boldsymbol{A}\|_S = \sqrt{14.39} \leqslant 3.80.$$ □

Störungen linearer Systeme

Wir betrachten nun das gestörte System (4.5) und nehmen an, dass die Störungen der Matrixelemente so klein sind, dass auch die Matrix $\boldsymbol{A} + \Delta\boldsymbol{A}$ regulär ist (dass dies für hinreichend kleine Störungen bei regulärer Matrix $\boldsymbol{A}$ überhaupt stets möglich ist, wird aus dem Satz 4.25 unten folgen).

Störungen linearer Systeme

Wir betrachten nun das gestörte System (4.5) und nehmen an, dass die Störungen der Matrixelemente so klein sind, dass auch die Matrix $\boldsymbol{A} + \Delta\boldsymbol{A}$ regulär ist (dass dies für hinreichend kleine Störungen bei regulärer Matrix $\boldsymbol{A}$ überhaupt stets möglich ist, wird aus dem Satz 4.25 unten folgen).

Löst man mit dieser Annahme (4.5) nach $\Delta\boldsymbol{x}$ auf, so erhält man für den durch die Störungen $\Delta\boldsymbol{A}$ und $\Delta\boldsymbol{b}$ hervorgerufenen absoluten Fehler wegen $\boldsymbol{Ax} = \boldsymbol{b}$

$$\begin{aligned}\Delta\boldsymbol{x} &= (\boldsymbol{A} + \Delta\boldsymbol{A})^{-1}(\Delta\boldsymbol{b} - \Delta\boldsymbol{A}\boldsymbol{x}) \\ &= (\boldsymbol{E} + \boldsymbol{A}^{-1}\Delta\boldsymbol{A})^{-1}\boldsymbol{A}^{-1}(\Delta\boldsymbol{b} - \Delta\boldsymbol{A}\boldsymbol{x}).\end{aligned}$$

Störungen linearer Systeme

Wir betrachten nun das gestörte System (4.5) und nehmen an, dass die Störungen der Matrixelemente so klein sind, dass auch die Matrix $\boldsymbol{A} + \Delta\boldsymbol{A}$ regulär ist (dass dies für hinreichend kleine Störungen bei regulärer Matrix $\boldsymbol{A}$ überhaupt stets möglich ist, wird aus dem Satz 4.25 unten folgen).

Löst man mit dieser Annahme (4.5) nach $\Delta\boldsymbol{x}$ auf, so erhält man für den durch die Störungen $\Delta\boldsymbol{A}$ und $\Delta\boldsymbol{b}$ hervorgerufenen absoluten Fehler wegen $\boldsymbol{Ax} = \boldsymbol{b}$

$$\begin{aligned}\Delta\boldsymbol{x} &= (\boldsymbol{A} + \Delta\boldsymbol{A})^{-1}(\Delta\boldsymbol{b} - \Delta\boldsymbol{A}\boldsymbol{x}) \\ &= (\boldsymbol{E} + \boldsymbol{A}^{-1}\Delta\boldsymbol{A})^{-1}\boldsymbol{A}^{-1}(\Delta\boldsymbol{b} - \Delta\boldsymbol{A}\boldsymbol{x}).\end{aligned}$$

Mit einer beliebigen Vektornorm $\|\cdot\|$ auf dem $\mathbb{R}^n$ und der gleich bezeichneten zugeordneten Matrixnorm kann man also abschätzen:

$$\|\Delta\boldsymbol{x}\| \leqslant \|(\boldsymbol{E} + \boldsymbol{A}^{-1}\Delta\boldsymbol{A})^{-1}\| \cdot \|\boldsymbol{A}^{-1}\| \left(\|\Delta\boldsymbol{b}\| + \|\Delta\boldsymbol{A}\| \cdot \|\boldsymbol{x}\|\right).$$

Störungen linearer Systeme

Für $\boldsymbol{b} \neq \boldsymbol{0}$ und folglich $\boldsymbol{x} \neq \boldsymbol{0}$ erhält man daraus für den relativen Fehler von $\boldsymbol{x}$, also $\|\Delta \boldsymbol{x}\| / \|\boldsymbol{x}\|$, die Abschätzung

$$\frac{\|\Delta \boldsymbol{x}\|}{\|\boldsymbol{x}\|} \leqslant \|(\boldsymbol{E} + \boldsymbol{A}^{-1}\Delta \boldsymbol{A})^{-1}\| \cdot \|\boldsymbol{A}^{-1}\| \left(\frac{\|\Delta \boldsymbol{b}\|}{\|\boldsymbol{x}\|} + \|\Delta \boldsymbol{A}\| \right). \tag{4.9}$$

Störungen linearer Systeme

Für $\boldsymbol{b} \neq \boldsymbol{0}$ und folglich $\boldsymbol{x} \neq \boldsymbol{0}$ erhält man daraus für den relativen Fehler von $\boldsymbol{x}$, also $\|\Delta \boldsymbol{x}\|/\|\boldsymbol{x}\|$, die Abschätzung

$$\frac{\|\Delta \boldsymbol{x}\|}{\|\boldsymbol{x}\|} \leqslant \|(\boldsymbol{E} + \boldsymbol{A}^{-1}\Delta \boldsymbol{A})^{-1}\| \cdot \|\boldsymbol{A}^{-1}\| \left(\frac{\|\Delta \boldsymbol{b}\|}{\|\boldsymbol{x}\|} + \|\Delta \boldsymbol{A}\| \right). \tag{4.9}$$

Um in dieser Ungleichung $\|(\boldsymbol{E} + \boldsymbol{A}^{-1}\Delta \boldsymbol{A})^{-1}\|$ weiter abzuschätzen, benötigen wir das folgende Störungslemma, das gleichzeitig darüber Auskunft gibt, unter wie großen Störungen der Matrixelemente die Existenz der Inversen für die gestörte Matrix noch gesichert ist.

Störungen linearer Systeme

Satz 4.25 (Störungslemma)

Es sei $\boldsymbol{B} \in \mathbb{R}^{n\times n}$ und es gelte für eine beliebige einer Vektornorm zugeordneten Matrixnorm die Ungleichung $\|\boldsymbol{B}\| < 1$. Dann ist die Matrix $\boldsymbol{E} - \boldsymbol{B}$ regulär, und es gilt

$$\|(\boldsymbol{E} - \boldsymbol{B})^{-1}\| \leqslant \frac{1}{1 - \|\boldsymbol{B}\|}.$$

Störungen linearer Systeme

Satz 4.25 (Störungslemma)

Es sei $\boldsymbol{B} \in \mathbb{R}^{n\times n}$ und es gelte für eine beliebige einer Vektornorm zugeordneten Matrixnorm die Ungleichung $\|\boldsymbol{B}\| < 1$. Dann ist die Matrix $\boldsymbol{E} - \boldsymbol{B}$ regulär, und es gilt

$$\|(\boldsymbol{E} - \boldsymbol{B})^{-1}\| \leqslant \frac{1}{1 - \|\boldsymbol{B}\|}.$$

Beweis.

Für alle $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{x} \neq \boldsymbol{0}$, *gilt*

$$\|(\boldsymbol{E} - \boldsymbol{B})\boldsymbol{x}\| \geqslant \|\boldsymbol{x}\| - \|\boldsymbol{B}\boldsymbol{x}\| \geqslant \|\boldsymbol{x}\| - \|\boldsymbol{B}\|\|\boldsymbol{x}\| = (1 - \|\boldsymbol{B}\|)\|\boldsymbol{x}\| > 0,$$

d.h. $(\boldsymbol{E} - \boldsymbol{B})\boldsymbol{x} = \boldsymbol{0}$ *ist nur für* $\boldsymbol{x} = \boldsymbol{0}$ *lösbar und daher ist* $\boldsymbol{E} - \boldsymbol{B}$ *regulär.*

Störungen linearer Systeme

Beweis.

Die Abschätzung der Norm der Inversen von $\boldsymbol{E} - \boldsymbol{B}$ erhält man so:

$$\begin{aligned}
1 &= \|(\boldsymbol{E} - \boldsymbol{B})^{-1}(\boldsymbol{E} - \boldsymbol{B})\| \\
&= \|(\boldsymbol{E} - \boldsymbol{B})^{-1} - (\boldsymbol{E} - \boldsymbol{B})^{-1}\boldsymbol{B}\| \\
&\geqslant \|(\boldsymbol{E} - \boldsymbol{B})^{-1}\| - \|(\boldsymbol{E} - \boldsymbol{B})^{-1}\boldsymbol{B}\| \\
&\geqslant \|(\boldsymbol{E} - \boldsymbol{B})^{-1}\| - \|(\boldsymbol{E} - \boldsymbol{B})^{-1}\| \cdot \|\boldsymbol{B}\| \\
&= (1 - \|\boldsymbol{B}\|) \cdot \|(\boldsymbol{E} - \boldsymbol{B})^{-1}\|.
\end{aligned}$$

Störungen linearer Systeme

Mit dem Störungslemma (mit $\boldsymbol{B} = -\boldsymbol{A}^{-1}\Delta\boldsymbol{A}$) können wir nun den relativen Fehler des gestörten Problems aus (4.9) weiter abschätzen, wobei wir $\|\boldsymbol{A}^{-1}\Delta\boldsymbol{A}\| \leqslant \|\boldsymbol{A}^{-1}\| \cdot \|\Delta\boldsymbol{A}\|$ und $\|\boldsymbol{b}\| = \|\boldsymbol{A}\boldsymbol{x}\| \leqslant \|\boldsymbol{A}\| \cdot \|\boldsymbol{x}\|$ beachten:

$$\begin{aligned}\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} &\leqslant \frac{\|\boldsymbol{A}^{-1}\|}{1 - \|\boldsymbol{A}^{-1}\| \cdot \|\Delta\boldsymbol{A}\|}\left(\|\boldsymbol{A}\|\frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|} + \|\Delta\boldsymbol{A}\|\right) \\ &\leqslant \frac{\|\boldsymbol{A}^{-1}\| \cdot \|\boldsymbol{A}\|}{1 - \|\boldsymbol{A}^{-1}\| \cdot \|\boldsymbol{A}\|\dfrac{\|\Delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|}}\left(\frac{\|\Delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|} + \frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|}\right). \qquad (4.10)\end{aligned}$$

Störungen linearer Systeme

Mit dem Störungslemma (mit $\boldsymbol{B} = -\boldsymbol{A}^{-1}\Delta\boldsymbol{A}$) können wir nun den relativen Fehler des gestörten Problems aus (4.9) weiter abschätzen, wobei wir $\|\boldsymbol{A}^{-1}\Delta\boldsymbol{A}\| \leqslant \|\boldsymbol{A}^{-1}\| \cdot \|\Delta\boldsymbol{A}\|$ und $\|\boldsymbol{b}\| = \|\boldsymbol{A}\boldsymbol{x}\| \leqslant \|\boldsymbol{A}\| \cdot \|\boldsymbol{x}\|$ beachten:

$$\begin{aligned}\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} &\leqslant \frac{\|\boldsymbol{A}^{-1}\|}{1 - \|\boldsymbol{A}^{-1}\| \cdot \|\Delta\boldsymbol{A}\|}\left(\|\boldsymbol{A}\|\frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|} + \|\Delta\boldsymbol{A}\|\right) \\ &\leqslant \frac{\|\boldsymbol{A}^{-1}\| \cdot \|\boldsymbol{A}\|}{1 - \|\boldsymbol{A}^{-1}\| \cdot \|\boldsymbol{A}\|\dfrac{\|\Delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|}}\left(\frac{\|\Delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|} + \frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|}\right). \qquad (4.10)\end{aligned}$$

Aus der Abschätzung (4.10) liest man ab, dass für kleine Störungen der Matrixelemente (so dass der Nenner nicht wesentlich von 1 abweicht) der relative Fehler der rechten Seite und der Matrixelemente um den Faktor $\|\boldsymbol{A}^{-1}\| \cdot \|\boldsymbol{A}\|$ verstärkt wird. Diesen Verstärkungsfaktor nennen wir die Kondition der Matrix $\boldsymbol{A}$.

Störungen linearer Systeme

Definition 4.26: Es sei $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ regulär und $\|\cdot\|_p$ eine einer (gleichbezeichneten) Vektornorm zugeordnete Matrixnorm auf $\mathbb{R}^{n \times n}$.

Störungen linearer Systeme

Definition 4.26: Es sei $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ regulär und $\|\cdot\|_p$ eine einer (gleichbezeichneten) Vektornorm zugeordnete Matrixnorm auf $\mathbb{R}^{n\times n}$.

Dann heißt

$$\kappa_p(\boldsymbol{A}) := \|\boldsymbol{A}^{-1}\|_p \cdot \|\boldsymbol{A}\|_p$$

die Kondition der Matrix $\boldsymbol{A}$ (oder des linearen Gleichungssystems (4.4)) bezüglich der Norm $\|\cdot\|_p$.

Störungen linearer Systeme

Definition 4.26: Es sei $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ regulär und $\|\cdot\|_p$ eine einer (gleichbezeichneten) Vektornorm zugeordnete Matrixnorm auf $\mathbb{R}^{n\times n}$.

Dann heißt

$$\kappa_p(\boldsymbol{A}) := \|\boldsymbol{A}^{-1}\|_p \cdot \|\boldsymbol{A}\|_p$$

die Kondition der Matrix $\boldsymbol{A}$ (oder des linearen Gleichungssystems (4.4)) bezüglich der Norm $\|\cdot\|_p$.

Wir fassen unsere Überlegungen zusammen:

Störungen linearer Systeme

Satz 4.27

Es seien $\boldsymbol{A}, \Delta\boldsymbol{A} \in \mathbb{R}^{n\times n}$ und $\boldsymbol{b}, \Delta\boldsymbol{b} \in \mathbb{R}^n$, $\boldsymbol{b} \neq \boldsymbol{0}$, gegeben, so dass $\boldsymbol{A}$ regulär ist und $\|\boldsymbol{A}^{-1}\| \cdot \|\Delta\boldsymbol{A}\| < 1$ mit einer Vektornorm zugeordneten Matrixnorm $\|\cdot\|$ gilt. Dann existiert neben der Lösung des linearen Gleichungssystems (4.4) auch die Lösung $\boldsymbol{x} + \Delta\boldsymbol{x}$ des gestörten Systems (4.5) und es gilt mit der Kondition $\kappa(\boldsymbol{A}) := \|\boldsymbol{A}\| \cdot \|\boldsymbol{A}^{-1}\|$ die Abschätzung

$$\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} \leqslant \frac{\kappa(\boldsymbol{A})}{1 - \kappa(\boldsymbol{A}) \cdot \frac{\|\Delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|}} \left(\frac{\|\Delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|} + \frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|} \right).$$

Störungen linearer Systeme

Satz 4.27

Es seien $\boldsymbol{A}, \Delta\boldsymbol{A} \in \mathbb{R}^{n\times n}$ und $\boldsymbol{b}, \Delta\boldsymbol{b} \in \mathbb{R}^n$, $\boldsymbol{b} \neq \boldsymbol{0}$, gegeben, so dass $\boldsymbol{A}$ regulär ist und $\|\boldsymbol{A}^{-1}\| \cdot \|\Delta\boldsymbol{A}\| < 1$ mit einer Vektornorm zugeordneten Matrixnorm $\|\cdot\|$ gilt. Dann existiert neben der Lösung des linearen Gleichungssystems (4.4) auch die Lösung $\boldsymbol{x} + \Delta\boldsymbol{x}$ des gestörten Systems (4.5) und es gilt mit der Kondition $\kappa(\boldsymbol{A}) := \|\boldsymbol{A}\| \cdot \|\boldsymbol{A}^{-1}\|$ die Abschätzung

$$\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} \leqslant \frac{\kappa(\boldsymbol{A})}{1 - \kappa(\boldsymbol{A}) \cdot \frac{\|\Delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|}} \left(\frac{\|\Delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|} + \frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|} \right).$$

Bemerkung 4.28: Für jede reguläre Matrix $\boldsymbol{A}$ und jede Norm $\|\cdot\|$ gilt $\kappa(\boldsymbol{A}) \geqslant 1$, denn

$$1 = \|\boldsymbol{E}\| = \|\boldsymbol{A}\boldsymbol{A}^{-1}\| \leqslant \|\boldsymbol{A}\| \cdot \|\boldsymbol{A}^{-1}\| = \kappa(\boldsymbol{A}). \qquad \square$$

Störungen linearer Systeme

Bemerkung 4.29: Werden die Rechnungen bei der Lösung eines linearen Gleichungssystems mit der Mantissenlänge ℓ ausgeführt, so haben die Daten von $\boldsymbol{A}$ und $\boldsymbol{b}$ bereits einen relativen Eingabefehler der Größe $5 \cdot 10^{-\ell}$.

Störungen linearer Systeme

Bemerkung 4.29: Werden die Rechnungen bei der Lösung eines linearen Gleichungssystems mit der Mantissenlänge ℓ ausgeführt, so haben die Daten von $\boldsymbol{A}$ und $\boldsymbol{b}$ bereits einen relativen Eingabefehler der Größe $5 \cdot 10^{-\ell}$.

Gilt $\kappa(\boldsymbol{A}) = 10^{\gamma}$, so ist (abgesehen von Rundungsfehlern, die sich im numerischen Lösungsalgorithmus ergeben) für die numerische Lösung mit einem relativen Fehler der Größe $5 \cdot 10^{\gamma-\ell}$ zu rechnen.

Störungen linearer Systeme

Bemerkung 4.29: Werden die Rechnungen bei der Lösung eines linearen Gleichungssystems mit der Mantissenlänge ℓ ausgeführt, so haben die Daten von $\boldsymbol{A}$ und $\boldsymbol{b}$ bereits einen relativen Eingabefehler der Größe $5 \cdot 10^{-\ell}$.

Gilt $\kappa(\boldsymbol{A}) = 10^{\gamma}$, so ist (abgesehen von Rundungsfehlern, die sich im numerischen Lösungsalgorithmus ergeben) für die numerische Lösung mit einem relativen Fehler der Größe $5 \cdot 10^{\gamma-\ell}$ zu rechnen.

Grob gesprochen verliert man also beim Lösen eines linearen Gleichungssystems γ Stellen, wenn die Koeffizientenmatrix eine Kondition der Größenordnung 10^{γ} besitzt. Dieser Verlust von Stellen ist nicht dem jeweilig verwendeten Algorithmus zuzuschreiben. Er ist problemimmanent. □

Störungen linearer Systeme

Beispiel 4.30: Wir betrachten das lineare Gleichungssystem

$$\begin{pmatrix} 1 & 1 \\ 1 & 0.999 \end{pmatrix} \boldsymbol{x} = \begin{pmatrix} 2 \\ 1.999 \end{pmatrix},$$

das offensichtlich die Lösung $\boldsymbol{x} = (1, 1)^T$ besitzt.

Störungen linearer Systeme

Beispiel 4.30: Wir betrachten das lineare Gleichungssystem

$$\begin{pmatrix} 1 & 1 \\ 1 & 0.999 \end{pmatrix} \boldsymbol{x} = \begin{pmatrix} 2 \\ 1.999 \end{pmatrix},$$

das offensichtlich die Lösung $\boldsymbol{x} = (1, 1)^T$ besitzt.

Für den Vektor $\boldsymbol{x} + \Delta\boldsymbol{x} := (5, -3.002)^T$ gilt

$$\boldsymbol{A}(\boldsymbol{x} + \Delta\boldsymbol{x}) = \begin{pmatrix} 1.998 \\ 2.001002 \end{pmatrix} =: \boldsymbol{b} + \Delta\boldsymbol{b}.$$

Störungen linearer Systeme

Beispiel 4.30: Wir betrachten das lineare Gleichungssystem

$$\begin{pmatrix} 1 & 1 \\ 1 & 0.999 \end{pmatrix} \boldsymbol{x} = \begin{pmatrix} 2 \\ 1.999 \end{pmatrix},$$

das offensichtlich die Lösung $\boldsymbol{x} = (1, 1)^T$ besitzt.

Für den Vektor $\boldsymbol{x} + \Delta\boldsymbol{x} := (5, -3.002)^T$ gilt

$$\boldsymbol{A}(\boldsymbol{x} + \Delta\boldsymbol{x}) = \begin{pmatrix} 1.998 \\ 2.001002 \end{pmatrix} =: \boldsymbol{b} + \Delta\boldsymbol{b}.$$

Es ist also

$$\frac{\|\Delta\boldsymbol{b}\|_\infty}{\|\boldsymbol{b}\|_\infty} = 1.001 \cdot 10^{-3} \quad \text{und} \quad \frac{\|\Delta\boldsymbol{x}\|_\infty}{\|\boldsymbol{x}\|_\infty} = 4.002,$$

Störungen linearer Systeme

. . . und daher gilt für die Kondition

$$\kappa_\infty(\boldsymbol{A}) \geqslant \frac{4.002}{1.001}\, 10^3 = 3998.$$

Störungen linearer Systeme

. . . und daher gilt für die Kondition

$$\kappa_\infty(\boldsymbol{A}) \geqslant \frac{4.002}{1.001}\, 10^3 = 3998.$$

Tatsächlich gilt

$$\boldsymbol{A}^{-1} = \begin{pmatrix} -999 & 1000 \\ 1000 & -1000 \end{pmatrix}$$

und daher $\kappa_\infty(\boldsymbol{A}) = 4000$.

Störungen linearer Systeme

. . . und daher gilt für die Kondition

$$\kappa_\infty(\boldsymbol{A}) \geqslant \frac{4.002}{1.001}\, 10^3 = 3998.$$

Tatsächlich gilt

$$\boldsymbol{A}^{-1} = \begin{pmatrix} -999 & 1000 \\ 1000 & -1000 \end{pmatrix}$$

und daher $\kappa_\infty(\boldsymbol{A}) = 4000$.

Man sieht an diesem Beispiel, dass die Abschätzung in (4.10) scharf ist. □

Störungen linearer Systeme

Der nächste Satz gibt eine geometrische Charakterisierung der Konditionszahl. Er sagt, dass der relative Abstand einer regulären Matrix zur nächsten singulären Matrix in der Euklidischen Norm gleich dem Reziproken der Kondition ist.

Störungen linearer Systeme

Der nächste Satz gibt eine geometrische Charakterisierung der Konditionszahl. Er sagt, dass der relative Abstand einer regulären Matrix zur nächsten singulären Matrix in der Euklidischen Norm gleich dem Reziproken der Kondition ist.

Satz 4.31

Es sei $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ regulär. Dann gilt

$$\min \left\{ \frac{\|\Delta \boldsymbol{A}\|_2}{\|\boldsymbol{A}\|_2} \, : \, \boldsymbol{A} + \Delta \boldsymbol{A} \text{ singulär} \right\} = \left(\kappa_2(\boldsymbol{A})\right)^{-1}.$$

Störungen linearer Systeme

Der nächste Satz gibt eine geometrische Charakterisierung der Konditionszahl. Er sagt, dass der relative Abstand einer regulären Matrix zur nächsten singulären Matrix in der Euklidischen Norm gleich dem Reziproken der Kondition ist.

Satz 4.31

Es sei $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ regulär. Dann gilt

$$\min \left\{ \frac{\|\Delta \boldsymbol{A}\|_2}{\|\boldsymbol{A}\|_2} \, : \, \boldsymbol{A} + \Delta \boldsymbol{A} \text{ singulär} \right\} = \left(\kappa_2(\boldsymbol{A})\right)^{-1}.$$

Beweis.

Siehe dazu das Skript „Grundlagen der Numerischen Mathematik“ von Heinrich Voß, Abschnitt 4.4. □

Software für lineare Gleichungssysteme

Sehr hochwertige Public Domain Software ist in den Bibliotheken LAPACK und ScaLAPACK erhältlich unter der Adresse

http://www.netlib.org/lapack/

bzw.

http://www.netlib.org/scalapack/

Software für lineare Gleichungssysteme

Die FORTRAN 77-Bibliothek LAPACK (und die Übertragungen in andere Sprachen: die C-Version CLAPACK, die C++ Version LAPACK++ und die FORTRAN 90 Version LAPACK90, die ebenfalls in der Netlib frei erhältlich sind) ist für PCs, Workstations, Vektorrechner oder Parallelrechner mit gemeinsamen Speicher geeignet, ScaLAPACK für Parallelrechner mit verteiltem Speicher oder vernetzte Workstations.

Software für lineare Gleichungssysteme

Die FORTRAN 77-Bibliothek LAPACK (und die Übertragungen in andere Sprachen: die C-Version CLAPACK, die C++ Version LAPACK++ und die FORTRAN 90 Version LAPACK90, die ebenfalls in der Netlib frei erhältlich sind) ist für PCs, Workstations, Vektorrechner oder Parallelrechner mit gemeinsamen Speicher geeignet, ScaLAPACK für Parallelrechner mit verteiltem Speicher oder vernetzte Workstations.

Beide Bibliotheken wurden unter Benutzung von BLAS3 geschrieben. Der Quellcode ist frei zugänglich und sehr gut dokumentiert. Die kommerziellen Bibliotheken IMSL oder NAG verwenden (zum Teil geringfügig modifizierte) LAPACK-Routinen. MATLAB verwendet ab Version 6.1 LAPACK-Routinen.