2 Lineare Gleichungssysteme

Wir betrachten das lineare Gleichungssystem

$$Ax = b$$

mit der $n \times n$-Koeffizientenmatrix A und der rechten Seite $b \in \mathbb{R}^n$.

Wir leiten zuerst eine Variante des Gauss-Algorithmus (LR-Zerlegung) her zum Lösen von $Ax = b$ und untersuchen, wann dieser Algorithmus ein *guter* Algorithmus ist (Pivot Strategie). Im zweiten Abschnitt dieses Kapitels betrachten wir grosse dünn besetzte Gleichungssysteme, wie sie in Anwendungen auftreten. Für solche Probleme ist der Gauss-Algorithmus nicht geeignet.

2.1 Gauss-Algorithmus, LR-Zerlegung

▷ K.Nipp/D.Stoffler [1], Par. 2.4, 2.5.

A) Ohne Zeilenvertauschungen

Herleitung an einem Beispiel ($n = 3$):

$$A = \begin{pmatrix} 2 & -1 & -3 \\ 6 & 1 & -10 \\ -2 & -7 & 8 \end{pmatrix}, \; b = \begin{pmatrix} 4 \\ -1 \\ 25 \end{pmatrix}$$

Gauss-Algorithmus nur mit Matrix A (Hauptteil), wobei die Nullen ohne Information durch die Quotienten der Hilfsspalte ersetzt werden:

$$\begin{array}{r|ccc} & \mathbf{2} & -1 & -3 \\ 3 & 6 & 1 & -10 \\ -1 & -2 & -7 & 8 \end{array} \xrightarrow{E_1} \begin{array}{c|ccc} & 2 & -1 & -3 \\ \hline 3 & 3 & \mathbf{4} & -1 \\ -1 & -1 & -8 & 5 \end{array} \xrightarrow{E_2} \begin{array}{ccc} 2 & -1 & -3 \\ 3 & 4 & -1 \\ -1 & -2 & \mathbf{3} \end{array}$$

Aus dem Endschema werden zwei Dreiecksmatrizen generiert:

$$\text{obere Dreiecksmatrix } R = \begin{pmatrix} 2 & -1 & -3 \\ 0 & 4 & -1 \\ 0 & 0 & 3 \end{pmatrix};$$

$$\text{untere Dreiecksmatrix } L = \begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ -1 & -2 & 1 \end{pmatrix},$$

wobei bei L in der Diagonalen zusätzlich Einsen eingefügt werden.

Wir bilden

$$LR = \begin{pmatrix} 2 & -1 & -3 \\ 6 & 1 & -10 \\ -2 & -7 & 8 \end{pmatrix}$$

und bemerken, dass $LR = A$ ist.

Satz: *Falls der Gauss-Alogrithmus angewandt auf die $n \times n$-Matrix A ohne Zeilenvertauschungen möglich ist, dann 'liefert' der Gauss-Algorithmus im Endschema eine Linksdreiecksmatrix L (mit Einsen in der Diagonalen) und eine Rechtsdreiecksmatrix R. L ist regulär und R hat 'Zeilenstufenform'. Es gilt: $LR = A$ (LR-Zerlegung von A).*

Bemerkung: Die LR-Zerlegung wird in der englischen Literatur LU-Zerlegung genannt (lower/upper triangular).

Es gilt:

$Ax = b$ ist äquivalent zu $LRx = b$.

Definiert man $c := Rx \in \mathbb{R}^n$, gilt

$$Ax = b \text{ ist äquivalent zu } \begin{cases} Lc = b \\ Rx = c \end{cases} .$$

Algorithmus: (LR-Zerlegung ohne Zeilenvertauschungen)

1. *LR-Zerlegung* von A: Mit dem Gauss-Algorithmus L, R, so dass $LR = A$.
2. *Vorwärtseinsetzen:* Löse $Lc = b$ nach c auf.
3. *Rückwärtseinsetzen:* Bestimme die Lösungsmenge von $Rx = c$.

Beispiel:

$$A = \begin{pmatrix} 2 & -1 & -3 \\ 6 & 1 & -10 \\ -2 & -7 & 8 \end{pmatrix}, \; b = \begin{pmatrix} 4 \\ -1 \\ 25 \end{pmatrix}.$$

LR-Zerlegung:

$$R = \begin{pmatrix} 2 & -1 & -3 \\ 0 & 4 & -1 \\ 0 & 0 & 3 \end{pmatrix}; \; L = \begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ -1 & -2 & 1 \end{pmatrix}.$$

Vorwärtseinsetzen ($Lc = b$):

$$\begin{array}{rrrcrcl} c_1 & & & = & 4 & 1) & c_1 = 4 \\ 3c_1 & +c_2 & & = & -1 & \Rightarrow\ 2) & c_2 = -1 \\ -c_1 & -2c_2 & +c_3 & = & 25 & 3) & c_3 = 3 \end{array} .$$

Rückwärtseinsetzen ($Rx = c$);

$$\begin{array}{rrrcrcl} 2x_1 & -x_2 & -3x_3 & = & 4 & 3) & x_1 = 2 \\ & 4x_2 & -x_3 & = & -13 & \Rightarrow\ 2) & x_2 = -3 \\ & & 3x_3 & = & 3 & 1) & x_3 = 1 \end{array} .$$

Bemerkung: Für A regulär ist der Aufwand gleich wie beim Gauss-Algorithmus mit 1-Spalte; bei mehreren rechten Seiten: n^2 wesentliche Operationen pro zusätzliches b.

B) Mit Zeilenvertauschungen

Definition: Eine $n \times n$-Matrix P mit genau einer 1 in jeder Zeile und in jeder Spalte und sonst 0 heisst *Permutationsmatrix*.

Bemerkung: P ist regulär, da P durch Vertauschen der Zeilen aus I_n erhalten wird. PA hat die gleichen Zeilen-Vertauschungen gegenüber A wie P gegenüber I_n.

Beispiel ($n = 3$):

$$P = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix};\ A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \Rightarrow\ PA = \begin{pmatrix} 4 & 5 & 6 \\ 1 & 2 & 3 \\ 7 & 8 & 9 \end{pmatrix}.$$

Wir betrachten das lineare Gleichungssystem $Ax = b$ mit

$$A = \begin{pmatrix} 0 & 3 & -2 \\ 4 & -2 & 1 \\ 2 & -1 & 1 \end{pmatrix},\ b = \begin{pmatrix} -1 \\ -1 \\ 8 \end{pmatrix}.$$

LR-Zerlegungsvariante des Gauss-Algorithmus für A mit I_3 zur Buchführung der Zeilenvertauschungen:

$$\begin{array}{c|ccc|ccc} 0 & 1 & 0 & 0 & 0 & 3 & -2 \\ 2 & 0 & 1 & 0 & 4 & -2 & 1 \\ & 0 & 0 & 1 & \mathbf{2} & -1 & 1 \end{array} \longrightarrow \begin{array}{c|ccc|c|cc} 0 & 0 & 0 & 1 & 2 & -1 & 1 \\ 0 & 1 & 0 & 0 & 2 & 0 & -1 \\ 1 & 0 & 0 & 0 & & \mathbf{3} & -2 \end{array}$$

$$\longrightarrow \begin{array}{ccc|ccc} 0 & 0 & 1 & \mathbf{2} & -1 & 1 \\ 1 & 0 & 0 & 0 & \mathbf{3} & -2 \\ 0 & 1 & 0 & 2 & 0 & \mathbf{-1} \end{array}$$

Aus dem erweitertem Endschema lesen wir 3 Matrizen ab:

$$L = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 0 & 1 \end{pmatrix}, \; R = \begin{pmatrix} 2 & -1 & 1 \\ 0 & 3 & -2 \\ 0 & 0 & -1 \end{pmatrix}; \; P = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Wir bilden:

$$LR = \begin{pmatrix} 2 & -1 & 1 \\ 0 & 3 & -2 \\ 4 & -2 & 1 \end{pmatrix} \text{ und } PA = \begin{pmatrix} 2 & -1 & 1 \\ 0 & 3 & -2 \\ 4 & -2 & 1 \end{pmatrix}$$

und stellen fest, dass $LR \neq A$, aber $PA = LR$.

Satz: *Das Gaussverfahren angewandt auf die $n \times n$-Matrix A liefert im erweiterten Endschema eine Linksdreiecksmatrix L (mit 1 in der Diagonalen), eine Rechtsdreiecksmatrix R und eine Permutationsmatrix P, so dass $LR = PA$ gilt. L ist regulär und R hat Zeilenstufenform (LR-Zerlegung von A).*

Es gilt:

$Ax = b$ ist äquivalent zu $PAx = Pb$.

Definiert man $c := Rx \in \mathbb{R}^n$, gilt

$Ax = b$ ist äquivalent zu $LRx = Pb$

$$Ax = b \text{ ist äquivalent zu } \begin{cases} Lc = b \\ Rx = c \end{cases}.$$

Algorithmus: (LR-Zerlegung)

1. *LR-Zerlegung* von A: Mit dem Gauss-Algorithmus L, R, P, so dass $LR = PA$.
2. *Vorwärtseinsetzen*: Löse $Lc = Pb$ nach c auf.
3. *Rückwärtseinsetzen*: Bestimme die Lösungsmenge von $Rx = c$.

Beispiel:

$$A = \begin{pmatrix} 0 & 3 & -2 \\ 4 & -2 & 1 \\ 2 & -1 & 1 \end{pmatrix}, \; b = \begin{pmatrix} -1 \\ -1 \\ 8 \end{pmatrix}.$$

LR-Zerlegung:

$$L = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 2 & -1 & 1 \\ 0 & 3 & -2 \\ 0 & 0 & -1 \end{pmatrix}; \; P = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.$$

Vorwärtseinsetzen ($Lc = Pb$):

$$\begin{array}{rrcrcll} c_1 & & = & 8 & & 1) & c_1 = 8 \\ & c_2 & = & -1 & \Rightarrow & 2) & c_2 = -1 \\ 2c_1 & + c_3 & = & -1 & & 3) & c_3 = -17 \; . \end{array}$$

Rückwärtseinsetzen ($Rx = c$):

$$\begin{array}{rrrcrcll} 2x_1 & -x_2 & + x_3 & = & 8 & & 3) & x_1 = 1 \\ & 3x_2 & - 2x_3 & = & -1 & \Rightarrow & 2) & x_2 = 11 \\ & & - x_3 & = & -17 & & 1) & x_3 = 17 \; . \end{array}$$

Bemerkung: In der Praxis (Berechnung der eindeutigen Lösung von $Ax = b$ auf dem Computer) spielt die Pivotwahl eine wichtige Rolle (endliche Arithmetik, Auslöschung). Bis jetzt haben wir nur die *Diagonalstrategie* betrachtet: Diese ist kein guter Algorithmus, ausser z.B. für diagonal dominantes A.

Besser ist die *Spaltenmaximumstrategie*.
Gut ist die *Relative Spaltenmaximumstrategie*.

Nur mit einer guten *Pivotstrategie* ist der Gauss-Algorithmus bzw. die LR-Zerlegung ein guter Algorithmus für das Lösen eines linearen Gleichungssystems (d.h. löst ein gut konditioniertes Problem so gut wie möglich).

Pivotstrategie

Die Pivots müssen so gewählt werden, dass Auslöschung vermieden wird. Wir untersuchen das Problem an einem

Beispiel:

$$\begin{aligned} 0.035x_1 &+ 3.62x_2 &= 9.12 \\ 1.17x_1 &+ 1.42x_2 &= 5.89 \end{aligned}$$

Exakte Lösung: $x_1 = 2,\ x_2 = 2.5$

Annahme: $\mathbb{M}(10; 3, , *)$ (mit Runden).

A) Diagonalstrategie (Pivot auf Diagonalen falls möglich)

$$\boxed{\begin{array}{rr} \mathbf{0.035} & 3.62 \\ 1.17 & 1.42 \end{array}} \rightarrow \boxed{\begin{array}{rr} 0.035 & 3.62 \\ 33.4 & \mathbf{-120} \end{array}}$$

Daraus ergibt sich:

$$\widehat{L} = \begin{pmatrix} 1 & 0 \\ 33.4 & 1 \end{pmatrix},\ \widehat{R} = \begin{pmatrix} 0.035 & 3.62 \\ 0 & -120 \end{pmatrix}$$

Beispielsweise ergibt sich der Wert für $\widehat{r}_{22}$ wie folgt:

$$\begin{aligned} \widehat{r}_{22} &= \rho(1.42 - \rho(33.4 \cdot 3.62)) \\ &= \rho(1.42 - \rho(120.908)) = \rho(1.42 - 121) \\ &= \rho(-11.58) = -120\ . \end{aligned}$$

Kontrolle:

$$\widehat{L}\widehat{R} = \begin{pmatrix} 0.035 & 3.62 \\ 1.169 & 0.908 \end{pmatrix}\ .$$

Der Eintrag 0.908 (anstatt 1.42), sehr ungenau!

Vorwärtseinsetzen ($\widehat{L}\widehat{c} = b$):

$$\begin{aligned} \widehat{c}_1 &= 9.12 \\ \widehat{c}_2 &= \rho(5.89 - \rho(33.4 \cdot 9.12)) = \rho(5.89 - 305) = -299 \end{aligned}$$

Rückwärtseinsetzen ($\widehat{R}\widehat{x} = \widehat{c}$):

$$\begin{aligned} \widehat{x}_2 &= \rho\left(\frac{-299}{-120}\right) = 2.49 \\ \widehat{x}_1 &= \rho\left(\frac{\rho(9.12 - \rho(3.62 \cdot 2.49))}{0.035}\right) = \underbrace{\rho\left(\frac{9.12 - 9.01}{0.035}\right)}_{\text{Auslöschung}} = 3.14\ . \end{aligned}$$

Der relative Fehler von $\widehat{x}$ ist $5.7 \cdot 10^{-1}$: Der Grund der Auslöschung ist die Wahl eines *kleinen* Pivots.

B) Spaltenmaximumstrategie (grösstmögliches Pivot)

$$\begin{aligned} \mathbf{1.17}x_1 &+ 1.42x_2 &=& \quad 5.894 \\ 0.035x_1 &+ 3.62x_2 &=& \quad 9.12 \end{aligned}$$

$$\widehat{L} = \begin{pmatrix} 1 & 0 \\ 0.0299 & 1 \end{pmatrix}, \ \widehat{R} = \begin{pmatrix} 1.17 & 1.42 \\ 0 & 3.58 \end{pmatrix}$$

Kontrolle:

$$\widehat{L}\widehat{R} = \begin{pmatrix} 1.17 & 1.42 \\ 0.035 & 3.62 \end{pmatrix} .$$

Dies ist sehr gut: $\widehat{L}\widehat{R} = A$ in $\mathbb{M}(10; 3, *)$.

Vorwärtseinsetzen ($\widehat{L}\widehat{c} = b$):

$$\begin{aligned} \widehat{c}_1 &= 5.89 \\ \widehat{c}_2 &= \rho(9.12 - \rho(0.0299 \cdot 5.89)) = \rho(9.12 - 0.176) = 8.94 \end{aligned}$$

Rückwärtseinsetzen ($\widehat{R}\widehat{x} = \widehat{c}$):

$$\begin{aligned} \widehat{x}_2 &= \rho(\frac{8.94}{3.58}) = 2.50 \\ \widehat{x}_1 &= \rho\left(\frac{\rho(5.89 - \rho(2.50 \cdot 1.42))}{1.17}\right) = \underbrace{\rho\left(\frac{5.89 - 3.55}{1.17}\right)}_{\text{keine Auslöschung}} = 2 \end{aligned}$$

Die numerische Lösung ist exakt.

Wir rechnen das Ganze nochmals, multiplizieren aber die erste Gleichung mit hundert:

$$\begin{aligned} \mathbf{3.5}\ x_1 &+ 362\ x_2 &=& \quad 912 \\ 1.17\ x_1 &+ 1.42\ x_2 &=& \quad 5.89 \end{aligned}$$

Dieses Gleichungssystem hat die gleiche (exakte) Lösung wie das ursprüngliche System: $x_1 = 2$, $x_2 = 2.6$.

Spaltenmaximumstrategie:

$$\widehat{L} = \begin{pmatrix} 1 & 0 \\ 0.334 & 1 \end{pmatrix}, \ \widehat{R} = \begin{pmatrix} 3.5 & 362 \\ 0 & -120 \end{pmatrix}$$

$$\widehat{c} = \begin{pmatrix} 912 \\ -299 \end{pmatrix}, \ \widehat{x} = \begin{pmatrix} 3.14 \\ 2.49 \end{pmatrix}$$

Diese numerische Lösung ist wieder sehr ungenau (wie im Falle der Diagonalstrategie).

C) Relative Spaltenmaximumstrategie

Wir skalieren zuerst jede Gleichung so, dass $\max_k |a_{ik}| = 1$.

Dann wählen wir das maximal skalierte Pivot:

$$\begin{aligned} 1.17\, x_1 \quad &+ 1.42 x_2 \quad = \quad 5.89\,; \qquad q_1 = \tfrac{1.17}{1.42} = \mathbf{0.824} \\ 3.5 x_1 \quad &+ 362\, x_2 \quad = \quad 912\,; \qquad q_2 = \tfrac{3.5}{362} = 0.00967\,. \end{aligned}$$

Da $0.824 > 0.00967$, wählen wir im 1. Gauss-Schritt 1.17 als Pivot. Das ergibt:

$$\widehat{L} = \begin{pmatrix} 1 & 0 \\ 2.99 & 1 \end{pmatrix},\ \widehat{R} = \begin{pmatrix} 1.17 & 1.42 \\ 0 & 358 \end{pmatrix}\ .$$

$$\widehat{c} = \begin{pmatrix} 5.89 \\ 894 \end{pmatrix},\ \widehat{x} = \begin{pmatrix} 2 \\ 2.5 \end{pmatrix}$$

Die numerische Lösung ist also wieder exakt.

Bemerkungen:

- Der Gauss-Algorithmus mit relativer Spaltenmaximumstrategie ist ein guter Algorithmus.
- Die Skalierung wird nicht durchgeführt. In jedem Schritt werden die q_i nur berechnet zur Wahl des Pivots.

2.2 Iterative Methoden für grosse lineare Gleichungssysteme

Motivation: In zahlreichen Anwendungen (Differenzenmethoden oder Finite-Elemente-Methoden zur approximativen Lösung von partiellen Differentialgleichungen) treten grosse lineare Gleichungssysteme auf (10^5-10^7 Unbekannte), deren Koeffizientenmatrix dünn besetzt (sparse) ist. Für solche Probleme sind die herkömmlichen direkten Methoden im allgemeinen nicht geeignet ('fill-in', d.h. Nullelemente werden zu Nichtnullelementen infolge der Pivotierung, weil die ganze Matrix gespeichert werden muss und wegen der grossen Komplexität $O(n^3)$).

Ausweg: Iterative Lösungsmethoden

Beispiele sind:

- Jacobi-Verfahren
- Gauss-Seidel-Verfahren
- SOR (sukzessive Überrelaxation)
- Methode des steilsten Abstiegs
- Methode der konjugierten Gradienten (CG)
- ...

Gemeinsame Merkmale sind:

- die Matrix A braucht nicht als Ganzes gespeichert zu werden.
- pro Iterationsschritt ist der Aufwand grob eine Matrix×Vektor-Multiplikation (nur die Nichtnullelemente).
- lineare Konvergenz.

A) Stationäre Iterationsverfahren

Sei A eine reelle, reguläre $n \times n$-Matrix und $b \in \mathbb{R}^n$. Wir betrachten das lineare Gleichungssystem

$$Ax = b \, . \tag{1}$$

Gesucht ist eine Approximation $\tilde{x}$ der eindeutigen Lösung x^*.

Wir betrachten die Iteration

$$x^{k+1} = f(x^k), \; k = 0, 1, 2, \ldots$$

Da (1) ein lineares Problem ist, machen wir einen linearen Ansatz

$$f(x) = Tx + c \, ,$$

wobei T eine $n \times n$-Matrix ist und $c \in \mathbb{R}^n$.

Damit gilt:

$$x^{k+1} = Tx^k + c, \quad k = 0, 1, 2, \tag{2}$$

Falls $\{x^k\}$ konvergiert, d.h. $\lim_{k\to\infty} x^k = x$, gilt

$$x = Tx + c\,, \tag{3}$$

und x ist eindeutig bestimmt, falls $I - T$ regulär ist:

$$x = (I - T)^{-1}c\,.$$

Damit x Lösung von (1) ist, muss gelten:

$$(I - T)^{-1}c = A^{-1}b \text{ (\textit{Konsistenz-Bedingung})} \tag{4}$$

Frage: Wann konvergiert $x^{k+1} = Tx^k + c\,, \quad k \to \infty$?

Wir betrachten den Fehlervektor

$$e^k := x^k - x\,.$$

Aus (2) und (3) folgt $e^{k+1} = Te^k$, und somit erhalten wir

$$e^k = T^k e^0, \quad k = 1, 2, \ldots$$

Satz: *Falls $||T|| < 1$, konvergiert das durch (2) definierte Iterationsverfahren für einen beliebigen Startvektor x^0.*

Beweis:

$$||e^k|| = ||T^k e^0|| \leq ||T||^k ||e^0||. \qquad \square$$

Bemerkungen:

- Der Satz gilt, falls $||T|| < 1$ in einer beliebigen Norm gilt; und aus $||T|| < 1$ folgt, dass $I - T$ regulär ist.
- Aus der Voraussetzung vom Satz und aus (4) folgt die Konvergenz von (2) gegen x^*.
- Die Konvergenz ist linear mit dem Abklingfaktor $\sim ||T||$.

Definition: Seien $\lambda_i, i = 1, \ldots, n$, Eigenwerte der Matrix T.
Dann heisst $\rho(T) := \max |\lambda_i|$ der *Spektralradius* von T.

Es gilt:

1. Für jede Norm ist $\rho(T) \leq ||T||$.
2. Für alle $\epsilon > 0$ existiert eine Norm $||.||_\epsilon$, so dass $||T||_\epsilon \leq \rho(T) + \epsilon$.

Beweis von 1. : Sei λ ein Eigenwert von T, und sei v der zugehörige Eigenvektor mit $||v|| = 1$. Dann gilt:

$$||T|| \cdot \underbrace{||v||}_{=1} \geq ||Tv|| = ||\lambda v|| = |\lambda| \cdot \underbrace{||v||}_{=1} = |\lambda|. \qquad \square$$

Folgerung: Falls $\rho(T) < 1$, konvergiert (2) für ein beliebiges x^0, und zwar in jeder Norm. (Hier wurde auch benutzt, dass aus $\rho(T) < 1$ folgt, dass die Matrix $I - T$ regulär ist.)

Ein geeignetes *Abbruchkriterium* für das Iterationsverfahren (2) ist

$$\frac{||x^k - x^{k-1}||}{||x^k||} \leq TOL + \frac{TOL}{||x^k||} \, .$$

Dieses Abbruchkriterium kontrolliert den absoluten Fehler (für $||x^k|| << 1$) und den relativen Fehler (sonst).

Spezielle Verfahren (Wahl der Iterationsmatrix T)

Wir zerlegen A wie folgt: $A = D + L + R$

$$A = \begin{pmatrix} \diagdown & & 0 \\ & \diagdown & \\ 0 & & \diagdown \end{pmatrix} + \begin{pmatrix} 0 & & 0 \\ & \diagdown & \\ * & & 0 \end{pmatrix} + \begin{pmatrix} 0 & & * \\ & \diagdown & \\ 0 & & 0 \end{pmatrix} .$$

Damit gilt:

$$\begin{aligned} Ax = b \quad &\Leftrightarrow \quad Dx + (L + R)x = b \\ &\Leftrightarrow \quad Dx = -(L + R)x + b \, . \end{aligned} \tag{5}$$

Die folgende Iterationsvorschrift heisst *Jacobi-Verfahren*:

$$Dx^{k+1} = -(L + R)x^k + b \, , \quad k = 0, 1, 2, \ldots \tag{6}$$

Es gilt: Falls D regulär, ist das Jacobi-Verfahren konsistent.

Beweis: Dies folgt aus (5) für $x^{k+1} = x^k = x$. $\square$

Nach obigem Satz und obiger Bemerkung **gilt**: Für D regulär konvergiert das Jacobi-Verfahren (6) für einen beliebigen Startvektor x^0 gegen die Lösung x^* von $Ax = b$, falls $\rho(D^{-1}(L+R)) < 1$.

Verbesserung der Konvergenz:

Wir berücksichtigen bereits berechnete Näherungen wie folgt:

$$Dx^{k+1} = -Lx^{k+1} - Rx^k + b \; .$$

Für D regulär ergibt dies das *Gauss-Seidelverfahren*:

$$x^{k+1} = -(D+L)^{-1}R\,x^k + (D+L)^{-1}b \; , \quad k = 0, 1, 2, \ldots \tag{7}$$

Es gilt: Für D regulär konvergiert das Gauss-Seidelverfahren (7) für ein beliebiges x^0 gegen x^*, falls $\rho((D+L)^{-1}R) < 1$.

Beweis: Die Konsistenz folgt direkt aus der Beziehung (7) für $x^{k+1} = x^k = x$. □

Definition: Eine Matrix A heisst *strikt diagonal dominant*, falls

$$|a_{ji}| > \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}|, \; i = 1, \ldots, n \; .$$

Es gilt: Falls A strikt diagonal dominant ist, konvergiert das Jacobi-Verfahren.

Beweis: Für die Maximum-Norm $||T||_\infty = \max_i \sum_{j=1}^n |t_{ij}|$ mit $T = -D^{-1}(L+R)$ gilt:

$$t_{ii} = 0, \; i = 1, \ldots, n \; , \text{ und } \sum_{j=1}^{n} |t_{ij}| = \sum_{\substack{j=1 \\ j \neq i}}^{n} \left| \frac{a_{ij}}{a_{ii}} \right| < 1 \; , \quad i = 1, \ldots, n \; . \qquad \square$$

Es gilt: Falls

- A strikt diagonal dominant
- A symmetrisch positiv definit, d.h. $\lambda_i(A) > 0, \; i = 1, \ldots, n$,

so konvergiert das Gauss-Seidel-Verfahren.

Variante des Gauss-Seidel-Verfahrens:
Die Kombination zwischen altem und neuem Gauss-Seidel-Vektor (wobei $w \in \mathbb{R}^+$):

$$Dx^{k+1} = w(-Lx^{k+1} - Rx^k + b) + (1-w)Dx^k, \quad k = 0, 1, 2, \ldots \tag{8}$$

ergibt das *SOR-Verfahren:*

$$x^{k+1} = \underbrace{(D + wL)^{-1}\,[-wR + (1-w)D]}_{=T(w)}\,x^k + (D + wL)^{-1} w\, b \tag{9}$$

Es gilt: Für D regulär konvergiert das SOR-Verfahren (9) für ein beliebiges x^0 gegen x^*, falls $\rho(T(\omega)) < 1$. Für den 'besten' Relaxations-Parameter

$$w_b \cong \frac{2}{1 + \sqrt{1 - \rho(L+R)^2}}$$

konvergiert das SOR-Verfahren für gewisse Modellprobleme schneller als das Gauss-Seidel-Verfahren.

Bemerkung: Die stationären Iterationsverfahren (mit dem gleichen T in jedem Schritt k) sind nicht geeignet für ganz grosse Probleme, da die Konvergenz zu langsam ist. (Für grosse Matrizen ist der Abklingfaktor $||T||$ nahe bei 1.) Wir betrachten im nächsten Abschnitt ein *nichtstationäres Verfahren*, das heute für symmetrisch positiv definites A wohl am häufigsten verwendet wird.

B) Die Methode der konjugierten Gradienten (CG)

(vorgeschlagen von Stiefel/Hestenes 1952)

Definition: Die Matrix A heisst *positiv definit*, falls

$$u^T A u > 0 \text{ für alle } u \in \mathbb{R}^n,\ u \neq 0\ .$$

Sei A eine reelle $n \times n$-Matrix, symmetrisch und positiv definit. Wir betrachten $Ax + b = 0$.

Es gilt:

1. $\lambda_i > 0,\ i = 1, \ldots, n$. (Dies impliziert, dass A regulär ist.)
2. Es existiert eine orthogonale Matrix T, so dass

$$T^T A T = D = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)\ .$$

Beweis:

1. Für λ Eigenwert von A und v zugehöriger Eigenvektor gilt:

$$0 < v^T Av = v^T \lambda v = \lambda ||v||^2, \text{ und das impliziert } \lambda > 0 .$$

2. Siehe K.Nipp/D.Stoffer [1], Par. 7.3. □

Wir betrachten

$$F(u) := \frac{1}{2} u^T Au + u^T b .$$

Da $A = A^T$, gilt:

$$\text{grad } F := \begin{pmatrix} \frac{\partial F}{\partial u_1} \\ \vdots \\ \frac{\partial F}{\partial u_n} \end{pmatrix} = Au + b =: r(u) \text{ (Residuum)} .$$

Behauptung: x erfüllt $Ax + b = 0 \Leftrightarrow x$ erfüllt $F(x) = \min_{u \in \mathbb{R}^n} F(u)$.

Beweis:

$'\Leftarrow'$: x Minimum $\Rightarrow \underbrace{\text{grad } F(x)}_{=Ax+b} = 0$.

$'\Rightarrow'$: $Ax + b = 0 \Rightarrow \text{grad } F = 0 \Rightarrow x$ ist stationärer Punkt.

Für $v \in \mathbb{R}^n,\ v \neq 0$, betrachten wir

$$\begin{aligned} F(x+v) &= \frac{1}{2}(x+v)^T A(x+v) + (x+v)^T b \\ &= F(x) + \frac{1}{2}\left[x^T Av + v^T Ax + v^T Av\right] + v^T b \\ &= F(x) + \frac{1}{2}\left[A(x)^T v - v^T b + v^T Av\right] + v^T b \\ &= F(x) + \frac{1}{2}\left[-b^T v - b^T v + v^T Av\right] + b^T v \\ &= F(x) + \frac{1}{2} v^T Av > F(x) . \end{aligned}$$

Da $v^T Av > 0$ nach Voraussetzung, folgt, dass x Minimum ist von $F(u),\ u \in \mathbb{R}^n$. □

Dieses Resultat legt das folgende Vorgehen nahe: Anstatt die Lösung von $Ax + b = 0$ zu bestimmen, bestimmen wir das Minimum von $F(u)$.

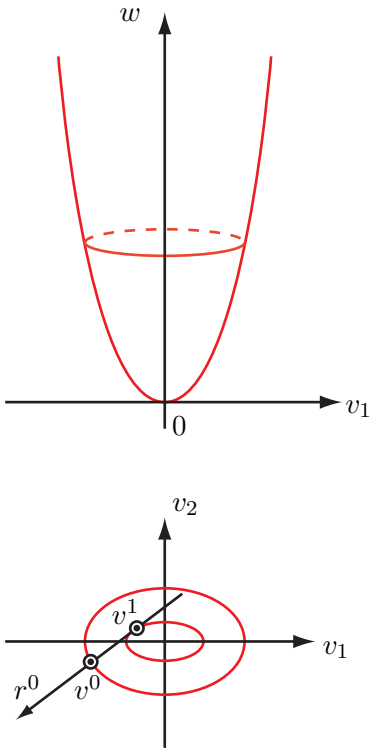
Geometrische Motivation:

Sei $n = 2$. Für eine gegebene Funktion F definiert $y = F(u)$ eine Fläche in $\mathbb{R}^3$, über der u_1-u_2-Ebene.

Sei $F(x) =: m$. Aus der Transformation $y = m + v,\ u = x + v$, ergibt sich

$$w = G(v) := \frac{1}{2} v^T A v,\ \text{grad}\ G = Av\ .$$

Daraus folgt, dass das Minimum in $v = 0$ ist und den Wert 0 hat. $w = G(v)$ beschreibt ein Paraboloid (siehe Skizze); die Niveaulinien $G(v) = \text{const}$ sind konzentrische Ellipsen.



Es gilt: $r :=$ grad G zeigt in Richtung der stärksten Zunahme und steht senkrecht auf den Niveaulinien.

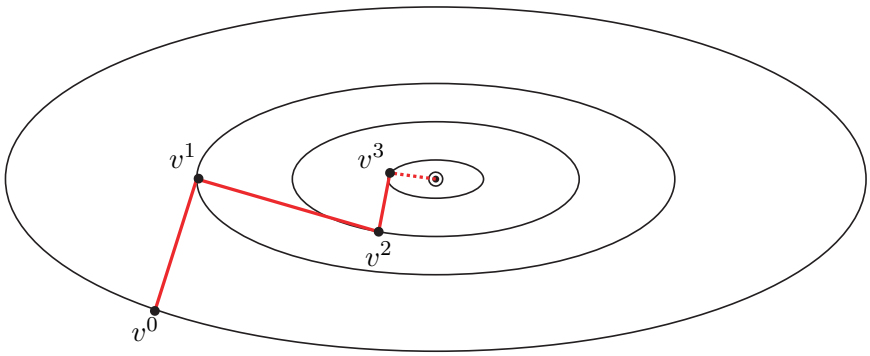
Idee: Wir betrachten einen Startpunkt v^0 und suchen das Minimum in Richtung $-\text{grad}\ G(v^0) =: -r^0$. Eine vertikale Ebene durch r^0 schneidet eine Parabel aus; $G(v)$ über r^0 ist minimal im Scheitel der Parabel. Die Niveaulinie durch diesen Punkt berührt also in der Projektion r^0; dies ergibt ein eindeutiges v^1 auf $(-r^0)$.

Fortsetzung dieses Prozesses: *Methode des steilsten Abstiegs.*

Nachteil der Methode des steilsten Abstiegs: Langsame Annäherung an 0, falls die Kondition von A gross ist:

$$\kappa(A) = ||A||_2 \cdot ||A^{-1}||_2 = \frac{\lambda_{max}}{\lambda_{min}}\ .$$

(Geometrisch betrachtet: langgestreckte Ellipsen)



Neue Idee: Die zur Richtung $-r^0$ in v^1 (Tangente an Niveaulinie) konjugierte Richtung geht durch 0. Bestimmt man das Minimum von $G(v)$ in dieser Richtung, ist man fertig (n=2).

Definition: Zwei Richtungen p, q sind konjugiert falls gilt $p^T Aq = 0$.

Begründung:

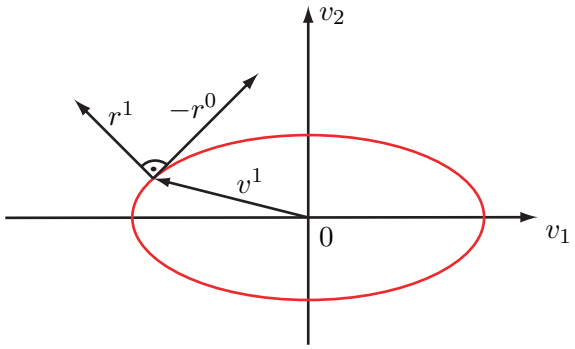
$$r^1 = grad\, G(v^1) = Av^1$$

Tangente an Ellipse in v^1:

$$p \perp r^1 \Leftrightarrow p^T r^1 = 0$$

$$\Rightarrow p = \pm r^0 \Rightarrow -r^{0T} Av^1 = 0$$

$$\Rightarrow -r^0, v^1 \text{ sind konjugiert (auch geometrisch).}$$

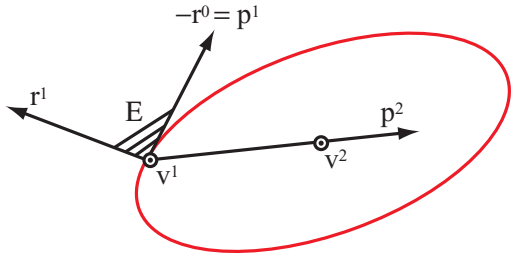


Sei nun $n = 3$: Niveauflächen $G(v) = \text{const}$ sind Ellipsoide.

1. Schritt: wie vorher (steilster Abstieg).

2. Schritt: suche Minimum in der durch $-r^0$ und r^1 aufgespannten Ebene.

Diese Ebene schneidet konzentrische Ellipsen heraus. Das Minimum liegt im gemeinsamen Mittelpunkt. Die zu $p^1(=-r^0)$ konjugierte Richtung p^2 zeigt dorthin.



3. Schritt: r^2 (= grad $G(v^2)$); suche Minimum in der durch p^2 und r^2 aufgespannten Ebene.

Der so bestimmte Punkt v^3 (analog Schritt 2) ist der Nullpunkt, und man ist fertig $(n = 3)$.

Bemerkung: Es gilt allgemein: Nach n Schritten ist das Minimum erreicht.

Notation: Skalarprodukt: $(p, q) = p^T q$.

Für das Skalarprodukt gilt:

$(p, q) = (q, p)$ (symmetrisch)
$(Ap, q) = (p, Aq)$, da $A = A^T$
$(Ap, p) > 0$ für $p \neq 0$, da A positiv definit ist (definiert A-Norm)
$(p,\ p) > 0$ für $p \neq 0$ (definiert 2-Norm).

Allgemeines n (ursprüngliche Koordinaten):

1. Schritt (steilster Abstieg):

$$\begin{aligned} u^0;\ r^0 &:= \text{grad } F(u^0),\ p^1 := -r^0; \\ u^1 &:= u^0 + \rho_1 p^1,\ \rho_1 := \frac{-(r^0, p^1)}{(Ap^1, p^1)}\ . \end{aligned}$$

Beweis, dass $F(u)$ in u^1 minimal über r^0 (lasse obere Indizes weg):

$$\begin{aligned} \frac{d}{d\rho}F(u+\rho p) &= \frac{d}{d\rho}\left[\frac{1}{2}(u+\rho p)^T A(u+\rho p) + (u+\rho p)^T b\right] \\ &= \frac{d}{d\rho}\left[\frac{1}{2}\left[u^T Au + \rho p^T Au + \rho u^T Ap + \rho^2 p^T Ap\right] + \rho p^T b\right] \\ &= p^T Au + \rho p^T Ap + p^T b \stackrel{!}{=} 0 \end{aligned}$$

$$\Rightarrow \rho = -\frac{p^T Au + p^T b}{p^T Ap} = -\frac{p^T(Au+b)}{p^T Ap} = -\frac{(p, r)}{(Ap, p)}\ .$$ □

k. Schritt ($k = 2, 3, \ldots$):

Ansatz: $p^k = -r^{k-1} + e_{k-1} p^{k-1}$

$$p^k \text{ konjugiert zu } p^{k-1}: \quad (p^k, Ap^{k-1}) = 0$$

$$\Rightarrow \qquad 0 = (p^k, Ap^{k-1}) = -(r^{k-1}, Ap^{k-1}) + e_{k-1}(p^{k-1}, Ap^{k-1})$$
$$\Rightarrow \qquad e_{k-1} = \frac{(r^{k-1}, Ap^{k-1})}{(p_{k-1}, Ap^{k-1})} \, .$$

In Richtung p^k bis zum Minimum (Beweis wie vorher):

$$u^k = u^{k-1} + \rho_k p^k, \; \rho_k = \frac{-(r^{k-1}, p^k)}{(Ap^k, p^k)} \, .$$

Gradient in u^k: $r^k = \text{grad } F(u^k) = Au^k + b$.

Es gilt:

$$i) \qquad (r^k, r^{k-1}) = 0$$
$$ii) \qquad (r^k, p^k) = 0, (r^k, p^{k-1}) = 0 \, .$$

Geometrische Begründung: u^k ist das Minimum von F über der von r^{k-1} und p^{k-1} aufgespannten Ebene E. Daraus folgt, dass diese tangential an das Niveau-Ellipsoid in u^k ist; r^k steht senkrecht auf dem Niveau-Ellipsoid in u^k; p^k liegt in E; also ist $r^k \perp r^{k-1}$ und $r^k \perp p^{k-1}$, $r^k \perp p^k$.

Daraus finden wir die geeigneteren Formeln:

$$\begin{aligned} (r^{k-1}, p^k) &= (r^{k-1}, -r^{k-1} + e_{k-1} p^{k-1}) \\ &= -(r^{k-1}, r^{k-1}) + e_{k-1} \underbrace{(r^{k-1}, p^{k-1})}_{=0} \end{aligned}$$

$$\Rightarrow \rho_k = \frac{(r^{k-1}, r^{k-1})}{(Ap^k, p^k)} \; (> 0) \, ;$$

$$r^k = Au^k + b = A(u^{k-1} + \rho_k p^k) + b = r^{k-1} \rho_k A p^k$$

$$\Rightarrow \qquad Ap^{k-1} = \frac{1}{\rho_{k-1}} \left[r^{k-1} - r^{k-2} \right]$$
$$\Rightarrow \qquad (r^{k-1}, Ap^{k-1}) = \frac{1}{\rho_{k-1}} (r^{k-1}, r^{k-1}) = (Ap^{k-1}, p^{k-1}) \frac{(r^{k-1}, r^{k-1})}{(r^{k-2}, r^{k-2})}$$
$$\Rightarrow \qquad e_{k-1} = \frac{(r^{k-1}, r^{k-1})}{r^{k-2}, r^{k-2}} \; (> 0) \, .$$

Algorithmus: (Methode der konjugierten Gradienten, (CG))

Start: $u^0;\ r^0 = Au^0 + b,\ p^1 = -r^0;\ TOL.$

Schritt k: $(k = 1, 2, \ldots)$

$$\begin{aligned} e_{k-1} &= \frac{(r^{k-1}, r^{k-1})}{(r^{k-2}, r^{k-2})} \text{ für } k \geq 2 \\ p^k &= -r^{k-1} + e_{k-1}p^{k-1} \text{ für } k \geq 2 \\ \rho_k &= \frac{(r^{k-1}, r^{k-1})}{(p^k, Ap^k)} \\ u^k &= u^{k-1} + \rho_k p^k \\ r^k &= r^{k-1} + \rho_k Ap^k\ . \end{aligned}$$

Abbruchkriterium: $||r^k|| \leq TOL$

Bemerkung: Pro Schritt sind nur eine Matrix×Vektor-Multiplikation und zwei Skalarprodukte nötig (Aufwand $O(n^2)$ bzw. $O(n)$; falls die Matrix dünn besetzt ist, beträgt der Aufwand je $O(n)$).

Satz: *Die* $p^j,\ 0 \leq j \leq k$, *sind paarweise konjugiert; die* r^j *sind paarweise orthogonal.*

(*Beweis* mit vollständiger Induktion nach dem Schritt k.)

Folgerung: Das CG-Verfahren liefert die Lösung von $Ax + b = 0$ für eine n×n-Matrix A nach höchstens n Schritten (exakte Rechnung vorausgesetzt).

Beweis: $r^0, r^1, \ldots, r^{n-1}$ sind paarweise orthogonal;

$$r^n \text{ ist orthogonal zu } r^0, \ldots, r^{n-1} \Leftrightarrow \underbrace{r^n}_{=Au^n+b} = 0 \Rightarrow u^n = x\ . \qquad \square$$

Bemerkungen:

1. Wegen der Rundungsfehler stehen die r^j nicht exakt senkrecht aufeinander. Das Minimum von $F(u)$ nimmt jedoch in jedem Schritt ab, d.h. man kann mehr als n Schritte machen.
2. In Anwendungen (PDEs), wo typischerweise n gross ist und A dünn besetzt (sparse), möchte man viel weniger als n Schritte machen.
3. Für n gross ist das CG-Verfahren nur direkt anwendbar, falls $\kappa(A)$ klein ist. Falls $\kappa(A)$ gross ist, braucht es eine *Vorkonditionierung* der Matrix A.

Es gilt:

$$\left\|u^j - x\right\|_A \leq 2\alpha^j \left\|u^0 - x\right\|_A, \quad \text{wobei } \|u\|_A^2 = (u, Au) = u^T Au \text{ und}$$

$$\alpha = \frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} - 1}\ (< 1)\ .$$

Beispiel:

$$\begin{aligned} \kappa_a = 9: &\quad \alpha = \tfrac{1}{2}; \quad \alpha_{st.Abst} = \tfrac{4}{5} \\ \kappa_a = 100: &\quad \alpha = \tfrac{9}{11}; \quad \alpha_{st.Abst} = \tfrac{99}{101} \ . \end{aligned}$$

Idee der Vorkonditionierung:

Sei $C = C^T$, positiv definit, und sei $C = HH^T$ (z.B: $H = L$ aus der *Cholesky-Zerlegung*, d.h. einer LR-Zerlegung mit $R = L^T$). Das lineare Gleichungssystem

$$Ax + b = 0$$

ist äquivalent zu

$$\underbrace{H^{-1}AH^{-T}}_{=:\widetilde{A}} \underbrace{H^T x}_{=:\widetilde{x}} + \underbrace{H^{-1}b}_{=:\widetilde{b}} = 0 \ ;$$

und $\widetilde{A}$ ist ähnlich zu $C^{-1}A$, da

$$H^{-T}\widetilde{A}H^T = H^{-T}H^{-1}A = (HH^T)^{-1}A = C^{-1}A \ .$$

Würde man also $C := A$ wählen, dann wäre $\widetilde{A}$ ähnlich zu I, d.h. $\kappa(\widetilde{A}) = 1$. Das ist aber nicht sinnvoll, da dies die direkte Lösung (Cholesky-Zerlegung) des ursprünglichen Problems bedeuten würde.

Folgerung: C sollte Approximation von A sein.

Möglichkeit: Gewinne H aus *unvollständiger Cholesky-Zerlegung* von A. Für dünn besetzte A gibt es bei $A = LL^T$ ein 'fill in' (Nullelemente werden zu Nichtnullelementen). Stattdessen unterdrückt man das 'fill-in': $A \cong HH^T =: C$ (wobei H dieselbe Struktur hat wie A). Diese Zerlegung existiert z.B. bei sogenannten M-Matrizen.

Der ursprüngliche CG-Algorithmus kann so angepasst werden, dass die Vorkonditionierung implizit berücksichtigt wird. Beim Start muss die unvollständige Cholesky-Zerlegung durchgeführt werden; in jedem Schritt muss zusätzlich das lineare Gleichungssystem $HH^T v^k = r^k$ durch effizientes Vor- und Rückwärtseinsetzen gelöst werden.

Bemerkungen:

- Die Matrix A muss immer als 'dünn besetzt' gespeichert werden.
- Falls A nicht symmetrisch positiv definit ist, gibt es auch iterative Methoden wie beispielsweise:
  - $BICG$ (Variante von CG)
  - $GMRES$

  Diese Verfahren sind im allgemeinen aber weniger robust.