4 Methode der konjugierten Gradienten

Als außerordentlich wirksam und den angegebenen iterativen Verfahren vielfach überlegen hat sich das Verfahren der *konjugierten Gradienten* erwiesen. Dieses Verfahren beruht darauf, daß nicht das Gleichungssystem $\boldsymbol{Ax} = \boldsymbol{b}$ direkt gelöst wird, sondern ein äquivalentes Optimierungsproblem. Diese Ideen gehen zurück auf Hestenes und Stiefel [1952].

Zur Beschreibung müssen wir etwas ausholen. Sei $f : \mathbb{R}^n \to \mathbb{R}$ eine Funktion von der wir ein (oder das) Minimum berechnen wollen. Eine sehr weit verbreitete Technik, auf die fast alle gängigen Optimierungsverfahren zurückzuführen sind, ist die folgende:

(a) Man wähle einen *Startpunkt* $\boldsymbol{x}_0 \in \mathbb{R}^n$ und eine *Startrichtung* $\boldsymbol{h}_0 \in \mathbb{R}^n$ mit $\boldsymbol{h}_0 \neq \boldsymbol{0}$.

(b) Man löse das eindimensionale Minimierungsproblem

$$\min_{\alpha \in \mathbb{R}} f(\boldsymbol{x}_0 + \alpha \boldsymbol{h}_0) = f(\boldsymbol{x}_0 + \alpha_0 \boldsymbol{h}_0). \tag{4.1}$$

(c) Man setze $\boldsymbol{x}_1 = \boldsymbol{x}_0 + \alpha_0 \boldsymbol{h}_0$ und wähle eine neue Richtung $\boldsymbol{h}_1$ und wiederhole (b) mit $\boldsymbol{x}_1, \boldsymbol{h}_1$ statt mit $\boldsymbol{x}_0, \boldsymbol{h}_0$, usw. Es gilt also i. a.

$$\boldsymbol{x}_{j+1} = \boldsymbol{x}_j + \alpha_j \boldsymbol{h}_j \implies \boldsymbol{x}_{j+1} = \boldsymbol{x}_0 + \sum_{\ell=0}^{j} \alpha_\ell \boldsymbol{h}_\ell. \tag{4.2}$$

Die Freiheit bei diesem Verfahren liegt abgesehen vom Startpunkt $\boldsymbol{x}_0$ bei der Wahl der Richtungen in jedem Schritt. Wir nennen alle diese Verfahren *Richtungsverfahren*. Sie sind bestimmt durch $\boldsymbol{x}_0, \boldsymbol{h}_0, \boldsymbol{h}_1, \ldots$, wenn wir unterstellen, daß alle Schritte eindeutig ausführbar sind.

Wir gehen in diesem ganzen Abschnitt aus von einer gegebenen Matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ und einem gegebenen Vektor $\boldsymbol{b} \in \mathbb{R}^n$, den wir auch *rechte Seite* nennen. Wir werden die weiteren Untersuchungen nur für symmetrische, positiv definite Matrizen $\boldsymbol{A}$ durchführen (sogenannte SPD-Matrizen), die in den Anwendungen oft vorkommen und für die die Theorie relativ glatt durchläuft. Eine quadratische Matrix $\boldsymbol{A}$ ist *symmetrisch*, wenn für alle Zeilen- und Spaltennummern j, die j-te Zeile mit der j-ten Spalte übereinstimmt, in Formeln $\boldsymbol{A} = \boldsymbol{A}^{\mathrm{T}}$, wobei das hochgestellte $^{\mathrm{T}}$ die Vertauschung von allen Zeilen mit allen Spalten derselben Nummer bedeutet. Eine symmetrische Matrix $\boldsymbol{A}$ ist *positiv definit*, wenn die quadratische Form $q(\boldsymbol{x}) := \boldsymbol{x}^{\mathrm{T}}\boldsymbol{Ax} > 0$ ist für alle $\boldsymbol{x} \neq \boldsymbol{0}$. Für eine 2×2-Matrix $\boldsymbol{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix}$ wäre mit $\boldsymbol{x} := (x_1, x_2)^{\mathrm{T}}$ entsprechend $q(\boldsymbol{x}) = a_{11}x_1^2 + 2a_{12}x_1x_2 + a_{22}x_2^2$. Wenn wir in diesem Abschnitt von positiv definiten Matrizen sprechen, meinen wir damit immer auch symmetrische Matrizen. Wir erinnern daran, daß eine positiv definite Matrix nicht singulär ist, denn für singuläre Matrizen $\boldsymbol{A}$ gibt es $\boldsymbol{x} \neq \boldsymbol{0}$ mit $\boldsymbol{Ax} = \boldsymbol{0}$, also auch $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Ax} = 0$, ein Widerspruch zur positiven Definitheit von $\boldsymbol{A}$.

Satz 4.1. Sei $\boldsymbol{A}$ eine SPD-Matrix. Dann sind die folgenden beiden Probleme (4.3) und (4.4) äquivalent. D. h., jedes Problem hat genau eine Lösung und die Lösung von (4.3) ist auch die Lösung von (4.4) und umgekehrt.

$$\begin{aligned} \boldsymbol{A}\boldsymbol{x} &= \boldsymbol{b}, && (4.3)\\ \min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) \text{ mit } f(\boldsymbol{x}) &:= \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{x}. && (4.4) \end{aligned}$$

Beweis: (i) Durch Nachrechnen oder Anwenden des Satzes von Taylor erhält man die für diesen Abschnitt zentrale Formel

$$f(\boldsymbol{x}+\boldsymbol{h}) = f(\boldsymbol{x}) + \boldsymbol{h}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}) + \frac{1}{2}\boldsymbol{h}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h} \text{ für alle } \boldsymbol{x},\boldsymbol{h}\in\mathbb{R}^n. \tag{4.5}$$

Sei $\hat{\boldsymbol{x}}$ die Lösung von (4.3). Dann folgt also nach (4.5) $f(\hat{\boldsymbol{x}}+\boldsymbol{h}) = f(\hat{\boldsymbol{x}}) + 0.5\boldsymbol{h}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}$. Daraus folgt wegen $\boldsymbol{h}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h} > 0$ für alle $\boldsymbol{h}\neq\boldsymbol{0}$ die Ungleichung $f(\hat{\boldsymbol{x}}) < f(\hat{\boldsymbol{x}}+\boldsymbol{h})$, d. h., $\hat{\boldsymbol{x}}$ ist eindeutiges Minimum.
(ii) Sei $\hat{\boldsymbol{x}}$ eine Lösung von (4.4). Dann ist nach der Definition eines Minimums und nach den bereits angestellten Rechnungen $f(\hat{\boldsymbol{x}}) \le f(\hat{\boldsymbol{x}}+\boldsymbol{h}) = f(\hat{\boldsymbol{x}}) + \boldsymbol{h}^{\mathrm{T}}(\boldsymbol{A}\hat{\boldsymbol{x}}-\boldsymbol{b}) + 0.5\boldsymbol{h}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}$ und also $\boldsymbol{h}^{\mathrm{T}}(\boldsymbol{A}\hat{\boldsymbol{x}}-\boldsymbol{b}) + 0.5\boldsymbol{h}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h} \ge 0$ für alle $\boldsymbol{h}$. Sei $\boldsymbol{y} = \boldsymbol{A}\hat{\boldsymbol{x}}-\boldsymbol{b}\neq\boldsymbol{0}$. Dann können wir $\boldsymbol{h} = -\dfrac{\boldsymbol{y}^{\mathrm{T}}\boldsymbol{y}}{\boldsymbol{y}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{y}}\boldsymbol{y}$ setzen, und wir erhalten $\boldsymbol{h}^{\mathrm{T}}(\boldsymbol{A}\hat{\boldsymbol{x}}-\boldsymbol{b}) + 0.5\boldsymbol{h}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h} = -0.5\dfrac{(\boldsymbol{y}^{\mathrm{T}}\boldsymbol{y})^2}{\boldsymbol{y}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{y}} < 0$, einen Widerspruch, d. h., $\boldsymbol{y} = \boldsymbol{A}\hat{\boldsymbol{x}}-\boldsymbol{b} = \boldsymbol{0}$, und $\hat{\boldsymbol{x}}$ ist (eindeutige) Lösung von (4.3). □

Im folgenden sei immer f wie in (4.4), Satz 4.1 definiert.

Definition 4.2. Bei gegebenen $\boldsymbol{x}, \boldsymbol{x}_j$, $j \ge 0$ heißen die im folgenden definierten Größen $\boldsymbol{r}$, bzw. $\boldsymbol{r}_j$ das *Residuum* und die Größen $\boldsymbol{e}$ bzw. $\boldsymbol{e}_j$ die *Fehler* von $\boldsymbol{x}$, bzw. von $\boldsymbol{x}_j$, wobei $\hat{\boldsymbol{x}}$ die Lösung von $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ bedeutet:

$$\boldsymbol{r} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x} \text{ bzw. } \boldsymbol{r}_j = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_j, \quad \boldsymbol{e} = \hat{\boldsymbol{x}} - \boldsymbol{x} \text{ bzw. } \boldsymbol{e}_j = \hat{\boldsymbol{x}} - \boldsymbol{x}_j. \tag{4.6}$$

Das Residuum wird vielfach zur Beurteilung der Güte einer Näherungslösung von $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ verwendet. Im Gegensatz zum Fehler kann es auch berechnet werden, wenn die Lösung $\hat{\boldsymbol{x}}$ des Gleichungssystems noch nicht bekannt ist. Wichtiger in diesem Zusammenhang ist jedoch auch die Interpretation des Residuums als *Richtung des steilsten Abstiegs*. Unter der Richtung des steilsten Abstiegs versteht man i. a. die Richtung $-f'(\boldsymbol{x}) = -(\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b})$, die hier gerade auf das Residuum von $\boldsymbol{x}$ führt. Ein Richtungsverfahren, das auf diesen Richtungen basiert, funktioniert aber oft nicht gut. Um dies zu erkennen, verfolge man was passiert, wenn man sich selbst in einem flachen Tal nach dieser Regel bewegt: ein langer Zickzackweg führt zum tiefsten Punkt. Man vergleiche dazu die Abbildung 3.1. Für die in (4.4) definierte Funktion f läßt sich leicht nachrechnen, daß mit dem eben eingeführten Fehler $\boldsymbol{e}$ gilt:

$$f(\boldsymbol{x}) = \frac{1}{2}\,\boldsymbol{e}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{e} + \text{const mit const} = \frac{1}{2}\,\hat{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{b}.$$

Da die Konstante nichts zur Minimierung beiträgt, hätten wir statt f auch

$$\tilde{f}(\boldsymbol{x}) := \frac{1}{2}\,\boldsymbol{e}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{e} \text{ mit } \boldsymbol{e} \text{ nach (4.6)} \tag{4.7}$$

in (4.4) benutzen können. Diese Interpretation werden wir später noch einmal verwenden.

Lemma 4.3. Sei $\boldsymbol{A}$ positiv definit. Dann hat bei beliebigem Vektor $\boldsymbol{h} \neq \boldsymbol{0}$ das eindimensionale Minimierungsproblem $\min_{\alpha\in\mathbb{R}} f(\boldsymbol{x}+\alpha\boldsymbol{h})$ mit $f(\boldsymbol{x}) = 0.5\,\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{x}$ die Lösung

$$\begin{aligned}
\alpha &= \frac{\boldsymbol{h}^{\mathrm{T}}\boldsymbol{r}}{\boldsymbol{h}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}}, \text{ und es gilt} && (4.8)\\
\boldsymbol{h}^{\mathrm{T}}\{\boldsymbol{A}(\boldsymbol{x}+\alpha\boldsymbol{h})-\boldsymbol{b}\} &= 0, && (4.9)\\
\min_{\alpha\in\mathbb{R}} f(\boldsymbol{x}+\alpha\boldsymbol{h}) &= f(\boldsymbol{x}) - \frac{1}{2}\frac{\{\boldsymbol{h}^{\mathrm{T}}\boldsymbol{r}\}^2}{\boldsymbol{h}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}}. && (4.10)
\end{aligned}$$

Beweis: Die Formel (4.5)

$$f(\boldsymbol{x}+\alpha\boldsymbol{h}) = f(\boldsymbol{x}) + \alpha\boldsymbol{h}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}) + \frac{1}{2}\alpha^2\boldsymbol{h}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}$$

hatten wir bereits im Beweis zu Satz 4.1 kennengelernt. Als Funktion von α erhält man daraus das Minimum durch Nullsetzen der Ableitung nach α. Daraus ergeben sich die angegebenen Formeln. □

Nach diesen Vorbereitungen stellt sich die Frage, wie man die Richtungen zur Minimierung von f tatsächlich wählt. Wir setzen

$$H_{j+1} = \langle\boldsymbol{h}_0,\ldots,\boldsymbol{h}_j\rangle,\ R_{j+1} = \langle\boldsymbol{r}_0,\ldots,\boldsymbol{r}_j\rangle,\ j = 0,1,\ldots,n-1, \tag{4.11}$$

für die von den Richtungen $\boldsymbol{h}_0,\boldsymbol{h}_1,\ldots,\boldsymbol{h}_j$ bzw. von den Residuen $\boldsymbol{r}_0,\boldsymbol{r}_1,\ldots,\boldsymbol{r}_j$ aufgespannten Vektorräume. Nach (4.2) ist $\boldsymbol{x}_{j+1}-\boldsymbol{x}_0 \in H_{j+1}$. Wir wollen untersuchen, ob wir die Bestimmung von $\boldsymbol{x}_{j+1}$ zurückführen können auf eine Minimierung in dem kleineren Teilraum H_j und auf eine eindimensionale Minimierung bezüglich $\langle\boldsymbol{h}_j\rangle$. Sei zu diesen Zweck $\boldsymbol{y} \in H_{j+1}$. Dann hat $\boldsymbol{y}$ die Darstellung

$$\boldsymbol{y} = \boldsymbol{x} + \alpha\boldsymbol{h}_j,\ \boldsymbol{x} \in H_j, \alpha \in \mathbb{R}.$$

Wenden wir Formel (4.5) an, so erhalten wir

$$f(\boldsymbol{y}) = f(\boldsymbol{x}) + \alpha\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} + \frac{1}{2}\alpha^2\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_j - \alpha\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{b}.$$

Der zweite Summand hängt sowohl von $\boldsymbol{x}$ als auch von α ab. Können wir diesen Term zum Verschwinden bringen, ist das Problem $\min_{\boldsymbol{y}\in H_{j+1}} f(\boldsymbol{y})$ in zwei getrennte Probleme, nämlich $\min_{\boldsymbol{x}\in H_j} f(\boldsymbol{x})$ und in $\min_{\alpha\in\mathbb{R}}(0.5\alpha^2\boldsymbol{h}_j\boldsymbol{A}\boldsymbol{h}_j - \alpha\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{b})$ aufspaltbar. Die Forderung

$\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} = 0$ bedeutet aber gerade, daß $\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_\ell = 0$ für alle $\ell = 0, 1, \ldots, j-1$ wegen $\boldsymbol{x} \in H_j$. Das führt auf folgende Definition.

Definition 4.4. Sei $\boldsymbol{A}$ positiv definit. Zwei Vektoren $\boldsymbol{x} \neq \boldsymbol{0}, \boldsymbol{y} \neq \boldsymbol{0}$ heißen *konjugiert in bezug auf* $\boldsymbol{A}$ (oder $\boldsymbol{A}$-konjugiert), wenn

$$< \boldsymbol{x}, \boldsymbol{y} >_{\mathrm{A}} := \boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{y} = 0.$$

Lemma 4.5. Die eben eingeführte Größe $< \cdot\,, \cdot >_{\boldsymbol{A}}$ ist ein Skalarprodukt im $\mathbb{R}^n$, entsprechend ist $\|\boldsymbol{x}\|_{\boldsymbol{A}} = \sqrt{< \boldsymbol{x}, \boldsymbol{x} >_{\mathrm{A}}}$ eine Norm, die wir $\boldsymbol{A}$-Norm von $\boldsymbol{x}$ nennen.

Beweis: Als Übung, man vgl. Beweis von Satz 3.13. □

Zwei Richtungen $\boldsymbol{h}_1, \boldsymbol{h}_2$ sind also konjugiert, wenn sie bezüglich des Skalarprodukts $< \cdot, \cdot >_{\boldsymbol{A}}$ orthogonal sind. Daher wird in der Literatur (z. B. Luenberger [1969, p.291]) auch von $\boldsymbol{A}$-Orthogonalität gesprochen. Sind in irgendeinem Raum mit Skalarprodukt $h_1, h_2, \ldots$ paarweise zueinander orthogonal und von Null verschieden, so sind sie auch linear unabhängig.

Die Verwendung konjugierter Richtungen läßt sich also als Globalstrategie auffassen, aus *allen* vorhergehenden Richtungen jeweils den kleinstmöglichen Wert von f auszurechnen. Die wesentliche Konsequenz steht im nachfolgenden Satz.

Satz 4.6. Seien n paarweise konjugierte Richtungen $\boldsymbol{h}_0, \boldsymbol{h}_1, \ldots, \boldsymbol{h}_{n-1}$ in bezug auf eine positiv definite Matrix $\boldsymbol{A}$ gegeben. Wählen wir diese Richtungen nacheinander im oben beschriebenen Richtungsverfahren bei beliebigem $\boldsymbol{x}_0$, so minimiert $\boldsymbol{x}_j$ die Funktion f im Teilraum H_j und

$$\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{r}_k = 0 \text{ für alle } 0 \leq j < k \leq n \text{ und } \boldsymbol{A}\boldsymbol{x}_n = \boldsymbol{b}. \tag{4.12}$$

Beweis: Die Minimalität von f im Raum H_j an der Stelle x_j ergibt sich aus den bereits angestellten Rechnungen. Für jedes Richtungsverfahren gilt $\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \alpha_{k-1}\boldsymbol{h}_{k-1}$, $k \geq 1$, also

$$\boldsymbol{x}_k = \boldsymbol{x}_{j+1} + \sum_{\ell=j+1}^{k-1} \alpha_\ell \boldsymbol{h}_\ell,\ 0 \leq j < k \leq n. \tag{4.13}$$

Aus Formel (4.9) folgt

$$\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{r}_{j+1} = 0,\ 0 \leq j \leq n-1. \tag{4.14}$$

Weiter ist nach (4.13) $\boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b} = \boldsymbol{A}(\boldsymbol{x}_{j+1} + \sum_{\ell=j+1}^{k-1} \alpha_\ell \boldsymbol{h}_\ell) - \boldsymbol{b} = \boldsymbol{A}\boldsymbol{x}_{j+1} - \boldsymbol{b} + \sum_{\ell=j+1}^{k-1} \alpha_\ell \boldsymbol{A}\boldsymbol{h}_\ell$, $0 \leq j < k \leq n$. Multiplizieren wir diese Gleichung mit $\boldsymbol{h}_j^{\mathrm{T}}$, so ist

$$\boldsymbol{h}_j^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b}) = \boldsymbol{h}_j^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{x}_{j+1} - \boldsymbol{b}) + \sum_{\ell=j+1}^{k-1} \alpha_\ell \boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_\ell = 0,$$

denn der erste Summand verschwindet wegen (4.14), der zweite wegen $\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_\ell = 0$ für $j \neq \ell$. Für $k = n$ folgt $\boldsymbol{h}_j^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{x}_n - \boldsymbol{b}) = 0$ für alle $0 \leq j \leq n-1$, also ist notwendig $\boldsymbol{A}\boldsymbol{x}_n = \boldsymbol{b}$, denn die $\boldsymbol{h}_j$ sind linear unabhängig, wie wir bereits hinter Lemma 4.5 behandelt haben. □

Dieser Satz besagt gerade, daß das Residuum $\boldsymbol{r}_k$ orthogonal ist zu allen Elementen von H_j, in Zeichen

$$\boldsymbol{r}_k \perp H_j \text{ für alle } 0 \leq j \leq k. \tag{4.15}$$

Damit wir wirklich nach diesem Richtungsverfahren rechnen können, benötigen wir noch eine Methode, konjugierte Richtungen möglichst effektiv herzustellen. Wir modifizieren zur Bestimmung der konjugierten Richtungen die in der Definition 4.2, S. 32 eingeführten Residuen in der Form $\boldsymbol{h}_{j+1} = \boldsymbol{r}_{j+1} + \beta_j \boldsymbol{h}_j$ und versuchen die β_j so zu bestimmen, daß *alle* Richtungen paarweise konjugiert sind. Aus $\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_{j+1} = \boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\{\boldsymbol{r}_{j+1} + \beta_j \boldsymbol{h}_j\} = 0$ folgt notwendig $\beta_j = -\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{r}_{j+1}/\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_j$. Daß diese Wahl tatsächlich *alle* Richtungen zueinander konjugiert, ist Inhalt des nächsten Satzes.

Satz 4.7. Sei $\boldsymbol{A}$ eine SPD-Matrix, $\boldsymbol{x}_0 \in \mathbb{R}^n$ beliebig und $\boldsymbol{h}_0 = \boldsymbol{r}_0$. Wir definieren

$$\begin{aligned}
\alpha_j &= \boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{h}_j/\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_j, && (4.16)\\
\boldsymbol{x}_{j+1} &= \boldsymbol{x}_j + \alpha_j \boldsymbol{h}_j, && (4.17)\\
\beta_j &= -\boldsymbol{r}_{j+1}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_j/\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_j, && (4.18)\\
\boldsymbol{h}_{j+1} &= \boldsymbol{r}_{j+1} + \beta_j \boldsymbol{h}_j,\ j = 0, 1, \ldots, n-1. \quad \text{Dann gilt} && (4.19)\\
\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_k &= 0,\ \boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{r}_k = 0,\ j \neq k,\ \boldsymbol{A}\boldsymbol{x}_n = \boldsymbol{b}. && (4.20)
\end{aligned}$$

Beweis: Nach Formel (4.19) mit $k = j+1$ ist

$$\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_k = \boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{r}_k + \beta_{k-1}\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_{k-1}.$$

Für $j = k-1$ heben sich nach der Definition (4.18) von β_j die beiden Summanden weg. Seien $\boldsymbol{h}_0, \boldsymbol{h}_1, \ldots, \boldsymbol{h}_{k-1}$ bereits zueinander konjugiert. Dann ist $\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{r}_k = 0$ für $j = 0, 1, \ldots, k-1$ nach Satz 4.6 Formel (4.12). Mit (4.19) folgt $\boldsymbol{r}_k^{\mathrm{T}}\boldsymbol{r}_j = \boldsymbol{r}_k^{\mathrm{T}}(\boldsymbol{h}_j - \beta_{j-1}\boldsymbol{h}_{j-1}) = 0$ für $j = 1, 2, \ldots, k-1$. Für $j = k$ folgt aus dieser Formel $\boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{r}_j = \boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{h}_j$, somit $\alpha_j > 0$ falls $\boldsymbol{r}_j \neq \boldsymbol{0}$. Wegen $\boldsymbol{h}_0 = \boldsymbol{r}_0$ ist auch $\boldsymbol{r}_k^{\mathrm{T}}\boldsymbol{r}_0 = 0$. Weiter ist nach (4.17) $\boldsymbol{r}_{j+1} = \boldsymbol{r}_j - \alpha_j \boldsymbol{A}\boldsymbol{h}_j$. Für $\boldsymbol{r}_j \neq \boldsymbol{0}$ ist also $\boldsymbol{A}\boldsymbol{h}_j = (\boldsymbol{r}_j - \boldsymbol{r}_{j+1})/\alpha_j$ und somit $\boldsymbol{h}_k^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_j = \boldsymbol{h}_k^{\mathrm{T}}(\boldsymbol{r}_j - \boldsymbol{r}_{j+1})/\alpha_j = 0$ für $j = 0, 1, \ldots, k-2$. Damit haben wir gezeigt, daß auch $\boldsymbol{h}_0, \boldsymbol{h}_1, \ldots, \boldsymbol{h}_k$ konjugiert sind. Daraus folgt auch nach den durchgeführten Rechnungen die mittlere Formel in (4.20). Daß $\boldsymbol{x}_n$ das Gleichungssystem löst, folgt aus Satz 4.6. □

Das in Satz 4.7 beschriebene Verfahren ist das eigentliche *Verfahren der konjugierten Richtungen*, auch *cg-Verfahren* genannt, von [engl.] *conjugate gradients*. Aus rechentechnischen Gründen sind noch einige Vereinfachungen vorzunehmen. Aus dem eben geführten Beweis übernehmen wir $\boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{h}_j = \boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{r}_j$. Wir ersetzen daher (4.16) durch $\alpha_j = \boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{r}_j/\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_j$. Danach ist dann $\boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{r}_j = \alpha_j\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_j$. Nun ist $\boldsymbol{r}_{j+1} = \boldsymbol{r}_j - \alpha_j\boldsymbol{A}\boldsymbol{h}_j$ und $\boldsymbol{r}_{j+1}^{\mathrm{T}}\boldsymbol{r}_j = 0$ nach (4.20). Also ist $\boldsymbol{r}_{j+1}^{\mathrm{T}}\boldsymbol{r}_{j+1} = -\alpha_j\boldsymbol{r}_{j+1}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_j$. Das führt auf $\beta_j = \boldsymbol{r}_{j+1}^{\mathrm{T}}\boldsymbol{r}_{j+1}/\boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{r}_j$.

Damit besteht das endgültige Verfahren der konjugierten Gradienten aus den Schritten:

$$\begin{aligned}
\boldsymbol{h}_0 &= \boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0 \text{ (zweckmäßig } \boldsymbol{x}_0 = 0), && (4.21)\\
\alpha_j &= \boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{r}_j/\boldsymbol{h}_j^{\mathrm{T}}\boldsymbol{A}\boldsymbol{h}_j, && (4.22)\\
\boldsymbol{x}_{j+1} &= \boldsymbol{x}_j + \alpha_j\boldsymbol{h}_j, && (4.23)\\
\boldsymbol{r}_{j+1} &= \boldsymbol{r}_j - \alpha_j\boldsymbol{A}\boldsymbol{h}_j, && (4.24)\\
\beta_j &= \boldsymbol{r}_{j+1}^{\mathrm{T}}\boldsymbol{r}_{j+1}/\boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{r}_j, && (4.25)\\
\boldsymbol{h}_{j+1} &= \boldsymbol{r}_{j+1} + \beta_j\boldsymbol{h}_j,\ j = 0, 1, \ldots, n-1. && (4.26)
\end{aligned}$$

Bei der Umsetzung in ein Programm ist natürlich vor (4.25) noch eine Prüfung auf $\boldsymbol{r}_j = \boldsymbol{0}$ vorzunehmen. Fällt diese Prüfung positiv aus, ist $\boldsymbol{x}_j$ bereits Lösung und das Programm abzubrechen. Pro Schritt ist die Anzahl der Operationen also Matrix mal Vektor, 2 Skalarprodukte, 3 (Vektor plus Vektor, Skalar mal Vektor), 2 Divisionen. Das sind zusammen (pro Schritt)

$$\text{Aufwand}_{\text{konjugierte Gradienten}} = n^2 + 5n \text{ Mult/Div und } n^2 + 4n \text{ Add.}$$

Bei n Durchläufen sind das also im wesentlichen n^3 Operationen. Bei größeren n zeigt die Erfahrung, daß man oft mit wesentlich weniger als mit n Durchläufen bereits zum Ende kommt. Das hat zuerst Reid [1971] bemerkt, der ein Gleichungssystem mit 4080 Gleichungen in 40 Schritten mit dem cg-Verfahren gelöst hat. Wir wollen uns überlegen, wie man ein derartiges Verhalten erklären kann. Nach Formel (4.2), S. 31 ist

$$\boldsymbol{x}_{j+1} \in \boldsymbol{x}_0 + H_{j+1} \text{ mit } H_{j+1} = \langle \boldsymbol{h}_0, \boldsymbol{h}_1, \ldots, \boldsymbol{h}_j \rangle,$$

und nach (4.26) gilt mit $R_{j+1} = \langle \boldsymbol{r}_0, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_j \rangle$ auch

$$H_{j+1} = R_{j+1},\ j = 0, 1, \ldots, n-1. \tag{4.27}$$

Es ist zweckmäßig, für beliebige Matrizen $\boldsymbol{B}$, Vektoren $\boldsymbol{h}$ und $k \in \mathbb{N}$ den folgenden Raum, der *Krylow-Raum* heißt, einzuführen:

$$\mathcal{K}(\boldsymbol{B}, \boldsymbol{h}, k) = \langle \boldsymbol{h}, \boldsymbol{B}\boldsymbol{h}, \boldsymbol{B}^2\boldsymbol{h}, \ldots, \boldsymbol{B}^{k-1}\boldsymbol{h} \rangle. \tag{4.28}$$

Lemma 4.8. Der von den Residuen $\boldsymbol{r}_0, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_j$ aufgespannte Raum stimmt überein mit dem von $\boldsymbol{h}_0, \boldsymbol{A}\boldsymbol{h}_0, \ldots, \boldsymbol{A}^j\boldsymbol{h}_0$ aufgespannten Krylow-Raum, oder in Formeln:

$$R_{j+1} = \mathcal{K}(\boldsymbol{A}, \boldsymbol{h}_0, j+1). \tag{4.29}$$

Beweis: Wir führen einen Induktionsbeweis. Für $j = 0$ ist die Aussage richtig. Sei die Behauptung bis $j - 1$ richtig. Nach (4.24) ist $\boldsymbol{r}_j = \boldsymbol{r}_{j-1} - \alpha_{j-1}\boldsymbol{A}\boldsymbol{h}_{j-1}$. Nach der Induktionsvoraussetzung und (4.27) gilt: $r_{j-1}, h_{j-1} \in \mathcal{K}(\boldsymbol{A}, \boldsymbol{h}_0, j)$, also $\boldsymbol{r}_j, \boldsymbol{h}_j \in \mathcal{K}(\boldsymbol{A}, \boldsymbol{h}_0, j+1)$. Da $\dim H_{j+1} = \dim R_{j+1} = j + 1$, folgt die Behauptung. □

Das Lemma besagt, daß der Abbruch des cg-Verfahrens mit $\boldsymbol{r}_m = \boldsymbol{0}$ auf $\dim \mathcal{K}(\boldsymbol{A}, \boldsymbol{h}_0, j + 1) = m$ für $j \geq m$ zurückgeführt werden kann. Die Dimension dieses Krylow-Raumes ist also entscheidend dafür, wann das cg-Verfahren spätestens (mit der Lösung) abbrechen muß.

Sei $\boldsymbol{e}_j$ der bereits in (4.6), S. 32 eingeführte Fehler von $\boldsymbol{x}_j$. Aus (4.2), S. 31 folgt die Formel $\boldsymbol{e}_{j+1} - \boldsymbol{e}_0 = \sum_{k=0}^{j} \alpha_k \boldsymbol{h}_k$, und nach dem Lemma 4.8 in Verbindung mit (4.27) gilt daher $\boldsymbol{e}_{j+1} - \boldsymbol{e}_0 \in \mathcal{K}(\boldsymbol{A}, \boldsymbol{h}_0, j+1)$. Benutzt man $\boldsymbol{h}_0 = \boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0 = \boldsymbol{A}\boldsymbol{e}_0$, so erhält man $\boldsymbol{e}_{j+1} - \boldsymbol{e}_0 \in \langle \boldsymbol{A}\boldsymbol{e}_0, \boldsymbol{A}^2\boldsymbol{e}_0, \ldots, \boldsymbol{A}^{j+1}\boldsymbol{e}_0 \rangle$. Diesen Sachverhalt kann man kurz so ausdrücken:

$$\boldsymbol{e}_j = p(\boldsymbol{A})\boldsymbol{e}_0,\ p \in \Pi_j, p(0) = 1,\ j = 0, 1, \ldots \tag{4.30}$$

Wegen $\boldsymbol{A}\boldsymbol{e}_j = \boldsymbol{r}_j$ folgt aus (4.30) die fast identische Gleichung

$$\boldsymbol{r}_j = p(\boldsymbol{A})\boldsymbol{r}_0,\ p \in \Pi_j, p(0) = 1,\ j = 0, 1, \ldots \tag{4.31}$$

Auch bei ganz anders hergeleiteten Iterationsverfahren, gelten häufig die beiden Formeln (4.30) und (4.31), (Opfer & Schober [1984]). Iterationsverfahren vom Typ (4.30) heißen auch *polynomiale Verfahren* (Fischer [1996]). Zur Erklärung des frühzeitigen Abbruchs des cg-Verfahrens ziehen wir die Formel (4.7) heran und benutzen die in Lemma 4.5 eingeführte $\boldsymbol{A}$-Norm und die Formel (4.30). Damit ist

$$\sqrt{2\tilde{f}(\boldsymbol{x}_j)} = ||\boldsymbol{e}_j||_{\boldsymbol{A}} = ||p(\boldsymbol{A})\boldsymbol{e}_0||_{\boldsymbol{A}} \leq ||p(\boldsymbol{A})||_{\boldsymbol{A}} ||\boldsymbol{e}_0||_{\boldsymbol{A}}, \tag{4.32}$$

wobei $||p(\boldsymbol{A})||_{\boldsymbol{A}}$ die zur Vektornorm $||\quad||_{\boldsymbol{A}}$ gehörende Operatornorm von $p(\boldsymbol{A})$ ist, die wir gerade im vorigen Abschnitt, Formel (3.17) ausgerechnet haben:

$$\|p(\boldsymbol{A})\|_{\boldsymbol{A}} = \max_{\lambda \in \sigma(\boldsymbol{A})} |p(\lambda)|,\ p \text{ Polynom.} \tag{4.33}$$

Da das cg-Verfahren in jedem Schritt gerade die in (4.32) angegebene $\boldsymbol{A}$-Norm minimiert, folgt daraus $||\boldsymbol{e}_j||_{\boldsymbol{A}} = 0$ und somit $\boldsymbol{e}_j = \hat{\boldsymbol{x}} - \boldsymbol{x}_j = 0$, falls es ein Polynom $p \in \Pi_j$ mit $p(0) = 1$ gibt, so daß $||p(\boldsymbol{A})||_{\boldsymbol{A}} = 0$ ausfällt.

Wir beweisen das folgende Lemma.

Lemma 4.9. Sei $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ eine SPD-Matrix mit $m \leq n$ (paarweise verschiedenen) Eigenwerten. Dann gibt es ein Polynom $p \in \Pi_m$ mit $p(0) = 1$ und $||p(\boldsymbol{A})||_{\boldsymbol{A}} = 0$.

Beweis: Seien $x_0 = 0, x_1, \ldots, x_m$ paarweise verschiedene reelle Werte, dann existiert nach bekannten Sätzen aus der Interpolationstheorie ein Polynom $p \in \Pi_m$ mit

$p(x_0) = 1, p(x_j) = 0, j = 1, 2, 3, \ldots, m$. Hat $\boldsymbol{A}$ die Eigenwerte $\lambda_1, \lambda_2, \ldots, \lambda_m$, so können wir daher ein Polynom $p \in \Pi_m$ finden mit $p(\lambda_j) = 0, j = 1, 2, \ldots, m$ und $p(0) = 1$. Für dieses Polynom ist $||p(\boldsymbol{A})||_{\boldsymbol{A}} = 0$ nach (4.33). □

Hat also die Matrix $\boldsymbol{A}$ gerade $m \leq n$ Eigenwerte, so erhält man mit dem cg-Verfahren (spätestens) nach m Schritten die Lösung. Dieser Effekt ist besonders dramatisch, wenn m im Vergleich zur Ordnung der Matrix $\boldsymbol{A}$ sehr klein ist, und das wird häufig beobachtet bei Matrizen, die sich aus der Diskretisierung von Differentialgleichungen ergeben. Weitere Informationen insbesondere auch im Zusammenhang mit Präkonditionierungstechniken geben Fischer & Freund [1992]. Wir beschließen diesen Abschnitt mit einigen Beispielen.

Beispiel 4.10. *Vandermonde-Matrix*
Eine Matrix, die einerseits leicht generierbar ist und andererseits jeden Gleichungslöser früher oder später zur Verzweiflung treibt, ist die Vandermondematrix $\boldsymbol{V}$. Seien $n \in \mathbb{N}, a, b$ mit $a < b$ vorgegeben. Wir lösen $\boldsymbol{Ax} = \boldsymbol{b}$ mit $\boldsymbol{A} = \boldsymbol{V}\boldsymbol{V}^{\mathrm{T}} = (v_{jk}),\ v_{jk} = \sum_\ell x_\ell^{j+k}$ und $x_j = a + j(b-a)/n,\ j, k = 0, 1, \ldots, n,\ a < b$. Die Matrix $\boldsymbol{A} \in \mathbb{R}^{n+1 \times n+1}$ hat hier die Eigenschaft, daß jeweils alle Elemente übereinstimmen für die gilt: „Zeilennummer j + Spaltennummmer k = const“. Derartige Matrizen heißen *Hankel-Matrizen*. Die rechte Seite $\boldsymbol{b} = (b_j)$ definieren wir so, daß wir die Lösung kennen, z. B. $b_j = \sum_{k=0}^n v_{jk}$. In diesem Fall ist die Lösung $\boldsymbol{x}^{\mathrm{T}} = (1, 1, \ldots, 1) \in \mathbb{R}^{n+1}$. Nehmen wir zum Beispiel $n = 127$, erhalten wir nach 18 Schritten für das Residuum $\boldsymbol{r}^{\mathrm{T}}\boldsymbol{r} = 1.26 \cdot 10^{-15}$ und beobachten eine Genauigkeit der Lösung von 10^{-4} (Rechnungen auf IBM RISC 6000/320). Selbst auf kleineren Rechnern (z. B. ATARI) konnten Probleme bis $n = 1000$ mühelos gerechnet werden.

Beispiel 4.11. *Modellproblem*
Hier dient ein physikalisches Problem und dessen mathematische Beschreibung als Hintergrund. Die Temperaturverteilung u in einer quadratischen Platte, die am Rand eine feste, vorgegebene Temperatur u_0 hat, wird beschrieben durch die sogenannte *Potentialgleichung*

$$\Delta u := u_{xx} + u_{yy} = 0, \text{ im Inneren der Platte; } u = u_0 \text{ am Rand.}$$

Als Quadrat wählen wir $Q = [0,1] \times [0,1]$. Die Funktion u hängt also von zwei Variablen x, y ab. Die Größen u_{xx}, u_{yy} bedeuten zweite Ableitungen nach x bzw. nach y. Zur Lösung zerlegen wir Q in kleine, kongruente Quadrate der Seitenlänge $h = 1/n$ und ersetzen Δu durch Differenzenquotienten an den Kreuzungspunkten der kleinen Quadrate. Die unbekannten Größen bezeichnen wir mit zwei Indizes, $x_{ij} \approx u(ih, jh), i, j = 1, 2, \ldots, n-1$. Das mit den Differenzenquotienten gebildete Ersatzproblem lautet dann

$$4x_{ij} - (x_{i,j-1} + x_{i,j+1} + x_{i-1,j} + x_{i+1,j}) = 0,\ i, j = 1, 2, \ldots, n-1,$$

$$x_{ij} = u_0(ih, jh) \text{ für } i = 0 \text{ oder } j = 0 \text{ oder } i = n \text{ oder } j = n.$$

Das ist ein lineares Gleichungssystem mit $(n-1)^2$ Unbekannten. Es ist hier nicht nötig, die $(n-1)^2 \times (n-1)^2$-Matrix explizit hinzuschreiben. Diese Matrix enthält in jeder Zeile maximal 5 nicht verschwindende Elemente, im ganzen also höchsten $5(n-1)^2$ Elemente ungleich Null, also mindestens $(n-1)^4 - 5(n-1)^2$ Nullen. Solche Matrizen nennt man *dünn besetzt*, auch [engl.] *sparse*. Dieses Modellproblem (oder leichte Verallgemeinerungen davon) dient zum Testen von Gleichungslösern verschiedener Konstruktion. Insbesondere interessiert man sich für den Aufwand, einen Fehler um einen vorgegebenen Faktor zu verkleinern. Ein Buch über iterative Verfahren zur Lösung derartiger linearer Gleichungssysteme von Hackbusch [1991] gibt weitere Informationen. . Ein Beispiel für das Modellproblem wird in der Vorlesung vorgeführt und in den Übungen behandelt.

Zum Schluß folgen eine in Matlab implementierte Version des cg-Verfahrens mit einem Beispiel für eine Funktion und eine Startdatei.

Programm 4.12. Ein cg-Matlab-Programm

```
function [lsg,k,nfc]=cg(x,amalvname,b,epsilon,kmax)
% ======================================================================
% Eine Implementation des CG-Verfahrens fuer quadratische Funktionen in
% der Form 0.5 x^t A x + b x + c.
% ----------------------------------------------------------------------
% Aurufparameter:
%  (lediglich die ersten drei Parameter muessen zwingend angegeben werden.)
% x    : aktueller Iterationspunkt. Spaltenvektor der Dimension n.
% amalvname : Name der Funktion die A * v berechnet.
% b    : Das b aus 0.5 x^T A x + b x + c.
% epsilon   : Abbruchgenauigkeit. Standard: 1E-8
% kmax : Anzahl der maximalen Iterationen. Standard: 100
% ---------------------------------------------------------------------
% Rueckgabeparameter:
% lsg : Loesung (Spaltenvektor!)
% k   : Anzahl der benoetigten Iterationen
% nfc : Anzahl der Funktionsauswertungen, muss allerdings in der
%   Unterroutine <fktname> gesetzt werden!
% ---------------------------------------------------------------------
% Programmvariablen:
% x         : aktueller Iterationspunkt x^k
% xneu      : naechster Iterationspunkt x^{k+1}
% amalx     : Produkt aus Matrix A mit Vektor x
% g         : Gradient an der Stelle x, d.h. A*x+b
% gg        : Skalarprodukt g^T * g
% gneu      : Gradient an der Stelle xneu
```

```
% gneugneu : Skalarprodukt (gneu)^T * gneu
% s        : Suchrichtung vom aktuellen x ausgehend
% amals    : Produkt aus Matrix A und Vektor s
% alpha    : Schrittlaenge
% beta     : Koeffizient fuer die neue Suchrichtung
% k        : aktuelle Iterationszahl
% kmax     : Anzahl der maximalen Iterationen
% ====================================================================
global nfc;

% Initialisierung
k = 0;
nfc = 0;
lsg = [];

if nargin < 5, kmax = 100; end;
if nargin < 4, epsilon = 1E-8; end;
% ------------------------------------------------------------
% Beginn des Algorithmus
% ------------------------------------------------------------
k = 0;
amalx = feval(amalvname,x);
[g] = amalx + b;
gg = g' * g;
s = -g;

while (sqrt(gg) > epsilon) & (k < kmax)
   % Bestimmung der Schrittlaenge
   amals = feval(amalvname,s);
   alpha = gg / (s' * amals);

   % Bestimmung des neuen Iterationspunktes
   xneu = x + alpha * s;
   amalx = amalx + alpha * amals;

   % Gradient und Norm des Gradienten fuer den neuen Iterationspunkt
   [gneu] = amalx + b;
   gneugneu = gneu' * gneu;

   % Bestimmung der konjugierten Richtungen
   beta = gneugneu / gg;
```

```
   s = -gneu + beta * s;

   % Uebertragen der neuen Werte auf die alten Variablen.
   x = xneu;
   g = gneu; gg = gneugneu
   k=k+1;
end; %while
% ----------------------------------------------------------------
% Ende des Algorithmus
% ----------------------------------------------------------------
lsg = x;
```

Programm 4.13. Eine Startdatei für das cg-Verfahren

```
function []=startcg()
% ========================================================================
% Startroutine fuer das CG-Verfahren mit einem
% einfachen Beispiel der Dimension 500.
% ------------------------------------------------------------------------

b = zeros(500,1);
amalvname='Amalv';
% Als Startvektor verwenden wir einen Vektor, der lauter Einsen enthaelt.
x = ones(500,1);

[lsg,k,nfc] = cg(x,amalvname,b);

disp('Komponenten des Loesungsvektors:')
disp(sprintf('%s%d','x(1) = ',lsg(1) ))
disp(sprintf('%s%d','x(100) = ',lsg(1) ))
disp(sprintf('%s%d','x(500) = ',lsg(1) ))
disp(sprintf('%s%d','Anzahl der Iterationen : ',k ))
disp(sprintf('%s%d','Anzahl der F-Auswertungen : ',nfc ))
```

Programm 4.14. Ein Beispiel für eine Funktion

```
function [y] = Amalv(v);
% ========================================================================
% Unterroutine zum CG-Verfahren fuer quadratische Funktionen.
% Die zu minimierende Funktion lautet:
%  f(x) = (x1)^2 + 2(x1)(x100) + 2(x100)^2 + 2(x500)
```

```
% Diese Routine berechnet das Produkt aus Matrix A und dem uebergebenen
% Vektor v. Beachte: A ist symmetrisch und positiv definit.
% -----------------------------------------------------------------------
global nfc
nfc = nfc+1;

y = zeros(500,1);
y(1,:) = 2*v(1,:) + 2*v(100,:);
y(100,:) = 2*v(1,:) + 4 * v(100,:);
y(500,:) = 2 * v(500,:);
```