TU-Chemnitz, Fakultät für Mathematik - Professur Numerische Mathematik
Vorlesung: Prof. Dr. Oliver Ernst **Übung: Dr. Roman Unger**
Homepage zur Übung: `https://www.tu-chemnitz.de/mathematik/numa/lehre/nla-2017`

8. Übung - Das CG-Verfahren

Aufgabe 1

Es seien

$$A = \mathrm{tridiag}(-12, -1) = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix} \in \mathbb{R}^{n\times n}$$

und $\boldsymbol{b} = (1, \ldots, 1)^\top \in \mathbb{R}^n$ gegeben. Nach wie vielen Schritten bricht ein Krylovverfahren zur Lösung von $A\boldsymbol{x} = \boldsymbol{b}$ mit Startvektor $\boldsymbol{x}_0 = \boldsymbol{0}$ ab? Beantworten Sie dazu die beiden folgenden Fragen.

1. Angenommen, $\boldsymbol{b}$ ist eine Linearkombination aus k verschiedenen Eigenvektoren von A. Nach wie vielen Schritten hat ein Krylovverfahren mit Startvektor $\boldsymbol{x}_0 = \boldsymbol{0}$ dann die Lösung von $A\boldsymbol{x} = \boldsymbol{b}$ gefunden?
2. Wie lässt sich das oben geannte $\boldsymbol{b}$ durch Eigenvektoren von A darstellen?

Aufgabe 2

Es sei $A \in \mathbb{R}^{n\times n}$ eine symmetrische, positiv definite Matrix, $\boldsymbol{b} \in \mathbb{R}^n$ sowie $f : \mathbb{R}^n \to \mathbb{R}$ geben durch

$$f(\boldsymbol{x}) := \frac{1}{2}\boldsymbol{x}^\top A\boldsymbol{x} - \boldsymbol{x}^\top \boldsymbol{b}.$$

1. Bestimmen Sie den Gradienten und die Hesse-Matrix von f.
2. Begründen Sie damit, dass das CG-Verfahren auch als Optimierungsverfahren für Funktionen wie f angesehen werden kann.
3. Zeigen Sie nun, dass sich in jedem Schritt des CG-Verfahrens der Funktionswert $f(\boldsymbol{x}_m)$ verringert.

Aufgabe 3

Beweisen Sie, dass für den Richtungsvektor $\boldsymbol{p}_m$ im m-ten Schritt des CG-Verfahrens die folgende Darstellung gilt

$$\boldsymbol{p}_m = \sum_{j=0}^{m-1} \frac{\|\boldsymbol{r}_{m-1}\|^2}{\|\boldsymbol{r}_j\|^2} \boldsymbol{r}_j.$$

Nutzen und zeigen Sie dazu, dass

$$\langle \boldsymbol{r}_m, \boldsymbol{p}_j \rangle = \begin{cases} 0, & 1 \leq j \leq m \\ \|\boldsymbol{r}_{j-1}\|^2, & m+1 \leq j \leq L. \end{cases}$$

Aufgabe 4

Es sei

$$T = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \in \mathbb{R}^{n \times n}$$

und

$$A = T \otimes I_n \otimes I_n + I_n \otimes T \otimes I_n + I_n \otimes I_n \otimes T.$$

Wir wollen den Vorteil des (vorkonditionierten) CG-Verfahrens gegenüber den direkten Lösern anhand des Systems $Ax = \boldsymbol{b}$ mit $\boldsymbol{b} = (1, \ldots, 1)^\top \in \mathbb{R}^{n^3}$ untersuchen.

1. Machen Sie sich mit der MATLAB-Funktion `pcg` vertraut.

2. Lassen Sie $n = 10, 20, \ldots, 70$ laufen und messen Sie die Rechenzeiten, die

   ```
   x = A\b;
   ```

   und die

   ```
   x = pcg(A,b,1e-10);
   ```

   in MATLAB benötigt und stellen Sie diese in einer doppelt-logarithmischen Grafik dar. Führen Sie dazu mehrere Durchgänge aus und bilden Sie den Mittelwert. Betrachten Sie ebenfalls den Fehler der vom CG-Verfahren berechneten Lösung bzgl. der direkten Lösung.

3. Wir verwenden nun einen ersten Vorkonditionierer für das CG-Verfahren, den Jacobi-Vorkonditionierer $M = \mathrm{diag}(A)$. Lassen Sie

   ```
   [x, flag, relres, iter] = pcg(A,b,1e-10, [], M);
   ```

   in MATLAB laufen für $n = 10, 20, \ldots, 70$ und betrachten Sie erneut die Rechenzeit, den Fehler zur direkten Lösung sowie die Anzahl der Iterationen im Vergleich zum CG-Verfahren.

4. Nun benutzen wir einen zweiten Vorkonditionierer, die unvollständige Cholesky-Zerlegung, welche in MATLAB für dünne Matrizen mittels `L = ichol(A)` berechnet werden kann. Führen Sie

   ```
   [x, flag, relres, iter] = pcg(A,b,1e-10, [], L,L');
   ```

   für $n = 10, 20, \ldots, 70$ aus und vergleichen Sie nun Rechenzeit, Fehler zur direkten Lösung und Anzahl der Iterationen mit dem normalen CG-Verfahren.