Prof. Dr. Dirk Lebiedz
M.Sc. Pascal Heiter
Dipl.-Math. oec. Klaus Stolle

Universität Ulm
Institut für Numerische Mathematik
Wintersemester 2013/14

Numerische Lineare Algebra - Matlabblatt 4

(Besprechung: Freitag, 20. Dezember 2013)

Aufgabe 8 *(Cholesky-Zerlegung)* (12 Punkte)

Wir wollen uns mit der Cholesky-Zerlegung für symmetrisch positiv definite Matrizen $A \in \mathbb{R}^{n\times n}$ befassen. Im zweiten Teil dieser Aufgabe hat die Matrix A auch noch eine spezielle Struktur, welche wir ausnutzen können.

a) Schreiben Sie eine Funktion

```
L = cholesky(A)
```

welche zu einer symmetrisch positiv definiten Matrix $A \in \mathbb{R}^{n\times n}$ die Cholesky-Zerlegung bestimmt, also eine untere Dreieckmatrix L berechnet mit $A = LL^T$ und testen Sie ihre Funktion mit Hilfe des Skriptes `test_cholesky.m`, welches auf der Homepage verfügbar ist.

Hinweis: Den Algorithmus finden Sie im Skript auf Seite 49.

b) Schreiben Sie eine Funktion

```
L = cholesky_band(A,m)
```

welche zu einer symmetrisch positiv definiten Matrix $A \in \mathbb{R}^{n\times n}$ mit einer **Bandmatrixstruktur** und Bandbreite $m \in \mathbb{N}$, $m < n$ die Cholesky-Zerlegung bestimmt, also eine untere Dreieckmatrix L berechnet mit $A = LL^T$ und testen Sie ihre Funktion mit Hilfe des Skriptes `test_choleskyBand.m`, welches auf der Homepage verfügbar ist.

Hinweis: Den Algorithmus und Erläuterungen zu Bandmatrizen finden Sie im Skript auf Seite 51ff.

c) Führen Sie das bereitgestellte Skript `test_runtime.m` aus und interpretieren Sie das Ergebnis. Was beobachten Sie hinsichtlich der Aufwandsabschätzungen aus den Sätzen 3.6.10 und 3.7.6?

Aufgabe 9 *(Thomas-Algorithmus für Tridiagonalmatrizen)* (8 Punkte)

Der Thomas-Algorithmus löst lineare Gleichungssysteme $Tx = (d_1, ..., d_n)^T$ für Tridiagonalmatrizen

$$T = \begin{pmatrix} b_1 & c_1 & 0 & \dots & & 0 \\ a_2 & b_2 & c_2 & \dots & & 0 \\ 0 & a_3 & b_3 & c_3 & & 0 \\ \vdots & & & \ddots & & \\ 0 & \dots & & a_{n-1} & b_{n-1} & c_{n-1} \\ 0 & \dots & & 0 & a_n & b_n \end{pmatrix} \in \mathbb{R}^{n\times n}$$

mit einem Aufwand von $\mathcal{O}(n)$. Dabei wird die Struktur der Matrix T ausgenutzt und das Gleichungssystem wie folgt gelöst:

1) Berechnen Sie einen Vektor $c' \in \mathbb{R}^{n-1}$ mit $c'_i = \begin{cases} \dfrac{c_i}{b_i} & i = 1 \\ \dfrac{c_i}{b_i - c'_{i-1}a_i} & i = 2, ..., n-1. \end{cases}$

2) Berechnen Sie einen Vektor $d' \in \mathbb{R}^n$ mit $d'_i = \begin{cases} \dfrac{d_i}{b_i} & i = 1 \\ \dfrac{d_i - d'_{i-1}a_i}{b_i - c'_{i-1}a_i} & i = 2, ..., n. \end{cases}$

3) Berechnen Sie die Lösung des Gleichungssystems $Ax = d$ durch

$$x_n = d'_n \qquad \text{und} \qquad x_i = d'_i - c'_i x_{i+1} \quad , i = n-1, ..., 1.$$

Schreiben Sie eine Funktion

```
x = thomasAlgo(T,d)
```

welche zu einer Tridiagonalmatrix $T \in \mathbb{R}^{n \times n}$ mit obiger Gestalt das lineare Gleichungssystem $Tx = d$ mit $d \in \mathbb{R}^n$ löst und testen Sie ihre Funktion mit Hilfe des Skriptes `test_thomasAlgo.m`, welches auf der Homepage verfügbar ist.