Einführung in die Grundlagen der Numerik

Winter semester 2019/2020
Prof. Dr. Marc Alexander Schweitzer
Denis Düsseldorf

Übungsblatt 5.

Abgabe: 19.11.2019 vor der Vorlesung

Aufgabe 11. (Gleichungssysteme und QR Verfahren)

In dieser Aufgabe benutzen wir die QR Zerlegung der regulären Matrix A um das Gleichungssystem $Ax = b$ zu lösen.

a) Machen Sie sich klar, dass die Lösung x des Gleichungsystems $Ax = b$ berechnet werden kann durch lösen von $Rx = c$ mit $c = Q^T x$. Diese Variante hat den Vorteil, dass $Rx = c$ wegen der oberen Dreiecksform von R sehr einfach durch Rückwärtssubstitution zu lösen ist.

b) Lösen Sie das Gleichungssystem $Ax = b$ mit

$$A = \begin{bmatrix} 1 & 2 \\ -1 & 2 \end{bmatrix}, \qquad b = \begin{bmatrix} 4 \\ 6 \end{bmatrix}$$

mittels QR-Zerlegung (in der euklidischen Norm). Berechnen Sie die QR Zerlegung per Hand!

(3 Punkte)

Aufgabe 12. (Gleichungssysteme und CG-Verfahren)

Berechnen Sie die Lösung des linearen Gleichungssystems

$$\begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} x = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

indem Sie das CG-Verfahren per Hand anwenden. Benutzen Sie den Startvektor $x_0 = \begin{bmatrix} 0 & 0 \end{bmatrix}$.

(2 Punkte)

Aufgabe 13. (Klassische Iterationsverfahren)

Wir rufen uns die klassischen Iterationsverfahren für lineare Gleichungssysteme $Ax = b$ in Erinnerung. Man nutzt eine Zerlegung der regulären Systemmatrix $A = M + (A - M)$ und erhält über

$$Ax = b \quad \Leftrightarrow \quad \Big[M + (A - M)\Big] x = b \quad \Leftrightarrow \quad Mx = b - (A - M)x \quad \Leftrightarrow \quad x = M^{-1}\Big[b - (A - M)x\Big]$$

die Iterationsvorschrift

$$x_{k+1} = M^{-1}\Big[b - (A - M)x_k\Big].$$

M sollte also einfach invertierbar sein.

Die Zerlegung

$$A = \underbrace{\begin{bmatrix} 0 & 0 \\ \star & 0 \end{bmatrix}}_{=:L} + \underbrace{\begin{bmatrix} \star & 0 \\ 0 & \star \end{bmatrix}}_{=:D} + \underbrace{\begin{bmatrix} 0 & \star \\ 0 & 0 \end{bmatrix}}_{=:U}$$

liefert mit

- $M = D$ das Jacobi Verfahren
- $M = L + D$ das Gauss-Seidel Verfahren.

Das Richardson Verfahren erhält man mit $M = I$.

a) Zeigen Sie, dass die Iterierten des Richardson-Verfahrens in einem Krylovraum liegen, d.h.
$$x_k \in b + K_k(A, c) = b + \text{span} c, Ac, \ldots, A^{k-1}c\}$$
für passende Vektoren b und c.

b) Zeigen Sie, dass die Iterierten jedes Verfahrens der Form
$$x_{k+1} = M^{-1}\Big[b - (A - M)x_k\Big]$$
basierend auf einer Zerlegung $A = M + (A - M)$ in einem Krylovraum liegen.

(3 Punkte)

Aufgabe 14. (Cholesky-Zerlegung)

Es sei $A \in \mathbb{R}^{n\times n}$ eine symmetrische, positiv-definite Matrix. Zeigen Sie: Es existiert eine eindeutige untere Dreiecksmatrix $G \in \mathbb{R}^{n\times n}$ mit positiven Diagonaleinträgen, so dass
$$A = GG^T$$
gilt. Diese Zerlegung nennt man Cholesky-Zerlegung der Matrix A.

(6 Punkte)

Programmieraufgabe 4. (CG Verfahren)

a) Implementieren Sie das CG-Verfahren zum lösen von $Ax = b$ mit $A \in \mathbb{R}^{n\times n}, b \in \ltimes$. Benutzen Sie die Signatur

```
# CG Verfahren
def solve_system_cg(A,b,x_0):
    # Input:
    #   A           System-Matrix
    #   b           rechte Seite
    #   x_0         Startvektor
    # Output:
    #   x           Approximation an die Loesung von Ax_*=b
    #   r_k_norms   Liste der Residuen aller Iterationen
```

Da numerisch nicht in exakter Arithmetik gerechnet wird, können wir nicht erwarten nach n Iterationen die exakte Lösung x_* berechnet zu haben. Falls die Norm des Residuums klein genug ist (nehmen Sie hier 10^{-8}) so kann das Verfahren abgebrochen werden. Falls mehr als k_max Iterationen gebraucht werden, so bricht man das Verfahren ebenfalls ab und gibt eine Fehlermeldung aus. Wählen Sie $k_max = 100$.

b) Schreiben Sie Ihr CG Verfahren so um, dass die norm des Residuums für jede Iteration verfügbar ist. Wenden Sie das CG-Verfahren auf das folgende Gleichungssystem an und plotten Sie die Konvergenz, d.h. die normen der Residuen aus jeder Iteration:
$$A \cdot \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{199} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 199 \end{bmatrix},$$
wobei A die Matrix `laplacian_200.txt` ist, die Sie auf der Webseite als Textdatei finden. Diese können Sie einlesen mittels

```python
A = np.loadtxt("laplacian_200.txt",delimiter=',')
```

(5 Punkte)

Abgabe per Mail an die Tutoren.