Vorlesung Numerische Mathematik
Wintersemester 2019/20
Prof. Dr. Bastian von Harrach
M. Sc. Tim Jahn

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

**8. Übungsblatt (erschienen am 03.12.2019)**

**Aufgabe 8.1 (Votieraufgabe)**
Betrachten Sie das lineare Ausgleichsproblem

$$\text{minimiere } \|Ax - b\|_2 \tag{1}$$

mit einer beliebigen Matrix $A \in \mathbb{C}^{m\times n}$ und einem Vektor $b \in \mathbb{C}^m$. Für ein $\rho_k > 0$ sei $x_k$ die Lösung der regularisierten Normalengleichung

$$(A^*A + \rho_k I)x = A^*b.$$

Zeigen Sie, dass die Matrix $A^*A + \rho_k I$ für jedes $\rho_k > 0$ invertierbar ist und dass $x_k \to A^+b$ für $\rho_k \to 0^+$.

**Aufgabe 8.2 (schriftliche Aufgabe)[6 Punkte]**
Seien $QR = A = \tilde{Q}\tilde{R}$ zwei $QR$-Zerlegungen einer injektiven Matrix $A \in \mathbb{C}^{m\times n}$, also $Q, \tilde{Q} \in \mathbb{C}^{m\times m}$ seien unitär und $R, \tilde{R} \in \mathbb{C}^{m\times n}$ obere Dreiecksmatrizen mit Diagonalelementen ungleich 0. Wir schreiben $Q = \begin{pmatrix} Q_1 & Q_2 \end{pmatrix}$ und $\tilde{Q} = \begin{pmatrix} \tilde{Q}_1 & \tilde{Q}_2 \end{pmatrix}$ mit $Q_1, \tilde{Q}_1 \in \mathbb{C}^{m\times n}$ sowie $R = \begin{pmatrix} R_1 \\ 0 \end{pmatrix}$, $\tilde{R} = \begin{pmatrix} \tilde{R}_2 \\ 0 \end{pmatrix}$ mit $R_1, \tilde{R}_1 \in \mathbb{C}^{n\times n}$. Zeigen Sie, dass eine unitäre Diagonalmatrix $D \in \mathbb{C}^{n\times n}$ (d.h. $|d_{ii}| = 1$) existiert, mit $\tilde{Q}_1 = Q_1 D^*$ und $\tilde{R}_1 = DR_1$.

**Aufgabe 8.3 (Programmieraufgabe)[6 Punkte]**
Betrachtet wird die in Abbildung 1 dargestellte Anordnung von Federn und Kugeln.

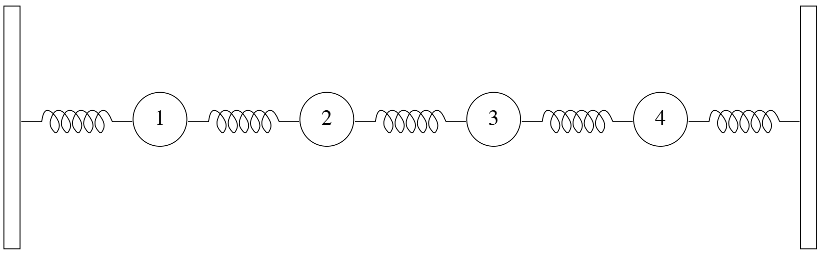


Abbildung 1: Anordnung von Federn und Kugeln.

Bezeichne $x_i$, $i = 1, \ldots, 4$, die Auslenkung der $i$-ten Kugel aus der Ruhelage nach rechts und $v_i$ ihre Geschwindigkeit, wobei $v_i > 0$ eine Bewegung nach rechts bedeutet. Durch die Federkräfte ändert sich im Zeitraum $\Delta t$ beispielsweise die Geschwindigkeit der zweiten Kugel gemäß

$$v_2^{\text{neu}} = v_2^{\text{alt}} + \Delta v_2 \quad \text{mit} \quad \Delta v_2 = k(x_1 - x_2)\Delta t + k(x_3 - x_2)\Delta t,$$

wobei $k$ von der Stärke der Feder und der Masse der Kugel abhängt und hier als konstant angenommen wird. Die Position der zweiten Kugel ändert sich gemäß

$$\Delta x_2 = x_2^{\text{neu}} - x_2^{\text{alt}} = v_2 \Delta t.$$

Stellen Sie für die anderen Kugeln entsprechende Gleichungen für $\Delta x_i$ und $\Delta v_i$ auf und fassen Sie diese in einem linearen Gleichungssystem der Form

$$\Delta\eta = \begin{pmatrix} \Delta x_1 \\ \Delta v_1 \\ \vdots \\ \Delta x_4 \\ \Delta v_4 \end{pmatrix} = \Delta t A \begin{pmatrix} x_1 \\ v_1 \\ \vdots \\ x_4 \\ v_4 \end{pmatrix} = \Delta t A \eta, \quad A \in \mathbb{R}^{8\times 8}, \tag{2}$$

zusammen. Zur Berechnung von

$$\eta^{\text{neu}} = \eta^{\text{alt}} + \Delta\eta$$

kann in der rechten Seite von (1) entweder $\eta^{\text{alt}}$ oder $\eta^{\text{neu}}$ verwendet werden. Im zweiten Fall muss die Gleichung dann noch nach $\eta^{\text{neu}}$ aufgelöst werden. Dazu sollte einmalig eine LR-Zerlegung der zu invertierenden Matrix mit dem `MATLAB`-Befehl `lu` berechnet werden, und die Lösungen sukzessiv wie in der Vorlesung in Bemerkung 2.3 beschrieben berechnet werden.
Implementieren und testen Sie beide Möglichkeiten für jeweils 300 Zeitschritte mit $\Delta t = 0.05$ und Anfangsvorgabe $\eta = (0.3,\ 0,\ -0.3,\ 0,\ 0.3,\ 0,\ -0.3,\ 0)^T$ und $k = 1$. Plotten Sie in einem Plot jeweils $x_1$ und $x_2$ abhängig von der Zeit.

Zusätzlich können Sie die zeitliche Entwicklung mit der unten angegebenen Funktion `Zeichne(eta)` visualisieren, indem Sie diese in jedem Zeitschritt aufrufen (hierzu ist nichts abzugeben).

```
function Zeichne(eta)
figure(1); set(gcf,'Color','w'); clf; axis([-0.5 10.5 -2 2]);
axis equal; hold on; axis tight; axis off;
[x01,y01]=MaleFeder(0,2+eta(1));
[x12,y12]=MaleFeder(2+eta(1),4+eta(3));
[x23,y23]=MaleFeder(4+eta(3),6+eta(5));
[x34,y34]=MaleFeder(6+eta(5),8+eta(7));
[x45,y45]=MaleFeder(8+eta(7),10);
plot([x01, x12, x23, x34, x45],[y01, y12, y23, y34, y45]);
patch([-0.2 0 0 -0.2],[-1 -1 1 1],'r');
patch([10.2 10 10 10.2],[-1 -1 1 1],'r');
plot([2,4,6,8]+eta([1,3,5,7])',zeros(1,4),'o','MarkerSize',20,...
    'MarkerEdgeColor','k','MarkerFaceColor','r');
end
```

```
function [x,y]=MaleFeder(a,b)
x=[a linspace(a+0.35,b-0.35,10) b];
y=[0 0 1 -1 1 -1 1 -1 1 -1 0 0]/5;
end
```

# Hinweise zur Übungsblattbearbeitung:

- Zu **schriftlichen Aufgaben*** soll eine Ausarbeitung/Lösung angefertigt werden, die bis zum 10.12.2019 um 10:00 Uhr in den Kästen ihres Übungleiters im 3. Stock der Robert-Mayer-Str. 6-8 abzugeben ist. Die jeweilige Fachnummer entnehmen Sie der Homepage.

***Die Abgabe und Bearbeitung darf in Zweiergruppen erfolgen.**

- Zu **Programmieraufgaben*** ist bis zum 10.12.2019 um 10:00 Uhr ein **kommentierter** `MATLAB`-Quellcode zu schreiben, welcher zusammen mit den (gegebenenfalls) damit erstellten Plots ausgedruckt und in den Kasten des Übungsleiters eingeworfen werden soll. Der Code ist nicht mehr per Mail einzureichen.

- Zu **Votieraufgaben** wird keine schriftliche Abgabe verlangt. Die Lösung wird in der Übung besprochen.