

Prof. Dr. A. Klawonn  
J. Knepper, M. Sc.  
J. Weber, M. Sc.

17. Oktober 2018

## 2. Übung zu Wissenschaftliches Rechnen I

### Aufgabe 1: (7 Punkte)

Gegeben sei eine symmetrisch positiv definite Blockmatrix der Form

$$K = \begin{pmatrix} A & B^T \\ B & C \end{pmatrix},$$

mit den Eigenwerten  $0 < \lambda_1 \leq \dots \leq \lambda_n$ . Zeigen Sie, dass für die Eigenwerte  $\mu_1 \leq \dots \leq \mu_m$  des Schurkomplements  $S = C - BA^{-1}B^T$  Folgendes gilt:

$$\mu_1 > 0 \quad \text{und} \quad \frac{\mu_m}{\mu_1} \leq \frac{\lambda_n}{\lambda_1}.$$

### Aufgabe 2: (3 + 3 = 6 Punkte)

- i) Überprüfen Sie anhand des Modellproblems aus der Programmieraufgabe (Dirichlet-Neumann), dass das Schurkomplement auf dem Interface  $\Gamma$  für das gegebene Modellproblem dicht besetzt ist.

Sei  $K$  die global assemblierte Steifigkeitsmatrix auf dem Gebiet  $\Omega$ , wobei die Partitionierung  $\Omega = I \cup \Gamma \cup D$ , also innere, Interface- und Randknoten, vorliegt. Stellen Sie das Schurkomplement auf dem Interface

$$S_{\Gamma\Gamma} = K_{\Gamma\Gamma} - K_{\Gamma I} K_{II}^{-1} K_{I\Gamma}$$

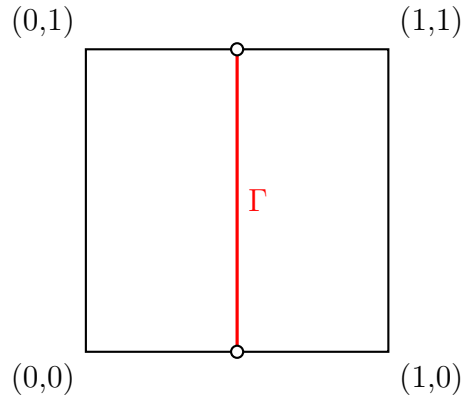
auf und plotten Sie die Dünnbesetztheitsstruktur mit dem Matlab-Befehl `spy` für  $S_{\Gamma\Gamma}$  und  $K_{II}$ .

- ii) Nutzen Sie Ihren Code, um die spektralen Konditionszahlen  $\kappa_2(K_{II})$  der global assemblierten Steifigkeitsmatrix und  $\kappa_2(S_{\Gamma\Gamma})$  des global assemblierten Schurkomplements zu berechnen.

*Hinweis:* Benutzen Sie hier nicht den Befehl `cond` in Matlab, um die spektrale Konditionszahl zu berechnen, sondern überlegen Sie, wie die Konditionszahl anders berechnet werden kann; vgl. mit Aufgabe 1.

### Programmieraufgabe (Dirichlet-Neumann): (20 Punkte)

Wir betrachten das Gebiet  $\Omega = (0, 1)^2$  und darauf das Modellproblem  $-\Delta u = 1$  mit der Dirichletrandbedingung  $u|_{\partial\Omega} = 0$ . Zur Diskretisierung nutzen wir ein strukturiertes Gitter mit linearen Dreieckselementen. Partitionieren Sie  $\Omega$  in die zwei Teilgebiete  $\Omega_1 = (0, 0.5) \times (0, 1)$  und  $\Omega_2 = (0.5, 1) \times (0, 1)$  und nutzen Sie zur Lösung des Modellproblems den Dirichlet-Neumann-Algorithmus. Nutzen Sie pro Teilgebiet  $2 \cdot 20^2$  Dreiecke.



Setzen Sie  $u_\Gamma^0 := 0$  und  $\theta = 0.25$  und lösen Sie abwechselnd das Dirichlet- und das Neumannproblem:

- Dirichletproblem auf  $\Omega_1$

$$K_{II}^{(1)} u_I^{(1),n} + K_{I\Gamma}^{(1)} u_\Gamma^{(1),n} = f_I^{(1)}, \quad u_\Gamma^{(1),n} := u_\Gamma^{n-1}$$

- $\lambda_\Gamma^{(1),n} := K_{\Gamma I}^{(1)} u_I^{(1),n} + K_{\Gamma\Gamma}^{(1)} u_\Gamma^{(1),n} - f_\Gamma^{(1)}$
- Neumannproblem auf  $\Omega_2$

$$\begin{pmatrix} K_{II}^{(2)} & K_{I\Gamma}^{(2)} \\ K_{\Gamma I}^{(2)} & K_{\Gamma\Gamma}^{(2)} \end{pmatrix} \cdot \begin{pmatrix} u_I^{(2),n} \\ u_\Gamma^{(2),n} \end{pmatrix} = \begin{pmatrix} f_I^{(2)} \\ f_\Gamma^{(2)} - \lambda_\Gamma^{(1),n} \end{pmatrix}$$

- $u_\Gamma^n := \theta u_\Gamma^{(2),n} + (1 - \theta) u_\Gamma^{(1),n}$
- Residuum  $r^n := u_\Gamma^{(2),n} - u_\Gamma^{(1),n}$

Iterieren Sie bis das relative Residuum kleiner als  $10^{-8}$  ist und geben Sie die benötigte Anzahl an Iterationen an. Plotten Sie in jedem Schritt die Teilgebietslösungen  $u^{(i),n}$  bis das relative Residuum kleiner als  $10^{-2}$  ist. Drucken Sie nur den Plot für  $n = 1$  aus. Sie können folgenden Code zum Plotten verwenden:

```
trisurf(tri__sd{1},x__sd{1}(:,1),x__sd{1}(:,2),u1)
hold on
trisurf(tri__sd{2},x__sd{2}(:,1),x__sd{2}(:,2),u2)
hold off
```

Hierbei sind im Cell-Array `tri__sd` die Elementlisten der Teilgebiete und in `x__sd` die zugehörigen Punkte gespeichert sowie in `u1` und `u2` die Teilgebietslösungen.

### Hinweise:

- Schreiben Sie Ihre Gitterpartitionierung für  $N$  Teilgebiete und halten Sie Ihren Code allgemein, um möglichst viel des Programms in Zukunft wiederverwenden zu können.
- Nutzen Sie `tripplot` und `scatter`, um die Korrektheit lokaler Gitter sowie von Randknoten etc. zu überprüfen.

- Zur Partitionierung des Gitters erstellen Sie für jedes Teilgebiet die folgenden Listen: Punkteliste, Elementliste und eine Indexabbildung. Steht bspw. der Punkt  $(1, 1)$  in der globalen Punkteliste an  $n$ -ter Stelle und ist im  $i$ -ten Teilgebiet mit der Nummer  $s$  versehen, so gilt

$$\text{IndexAbbildung}_i(s) = n.$$

- *Folgende Vorgehensweise ist ein Vorschlag:* Angenommen Sie haben einen logischen Vektor  $\mathbf{b}$ , welcher in der globalen Punkteliste  $\mathbf{x}$  alle Knoten markiert, welche in Teilgebiet  $i$  liegen, d.h. es gilt  $\mathbf{x}_{\text{sd}\{i\}} = \mathbf{x}(\mathbf{b}, :)$ ; dann können Sie die zugehörigen Dreiecke aus der globalen Elementliste  $\mathbf{tri}$  über  $\mathbf{tri}_{\text{sd}\{i\}} = \mathbf{tri}(\text{all}(\mathbf{b}(\mathbf{tri}), 2), :)$ ; extrahieren (Achtung: Dies funktioniert nur für das strukturierte Gitter mit der beschriebenen Partitionierung). Die Indexabbildung erhalten Sie über  $\mathbf{l2g}_{\text{sd}\{i\}} = \text{find}(\mathbf{b})$ ; Die lokale Elementliste enthält bisher noch die globalen Knotennummern. Wir müssen also die Umkehrabbildung zu  $\mathbf{l2g}_{\text{sd}\{i\}}$  aufstellen und diese auf  $\mathbf{tri}_{\text{sd}\{i\}}$  anwenden. Die Umkehrabbildung erhält man über  $\text{map} = \text{zeros}(\text{size}(\mathbf{x}, 1), 1)$ ; und  $\text{map}(\mathbf{b}) = 1:\text{nnz}(\mathbf{b})$ ;
- Sie können das Interface einfach (allgemein) bestimmen, indem Sie zählen, wie oft auf einen Knoten verwiesen wird. Nutzen Sie dazu Ihre Listen  $\mathbf{l2g}_{\text{sd}\{i\}}$ . Achten Sie bei der Bestimmung des Interfaces  $\Gamma$  darauf, dass die Endpunkte (die auf dem Globalrand liegen) nicht in  $\Gamma$  enthalten sind.
- **Achten Sie stets darauf, dass Sie die korrekte Knotensortierung verwenden.** Beispielsweise hat das Gebiet  $\Omega_2$  eine andere Knotensortierung des Interfaces als  $\Omega_1$ . Zum „Austausch“ von Werten können Sie der Einfachheit halber die Werte in einen Vektor mit globaler Knotensortierung schreiben bzw. auslesen. Beispiel:  $\text{tmp}(\mathbf{l2g}_{\text{sd}\{2\}}(\text{gamma2})) = \text{uGamma2}$  und  $\text{tmp}(\mathbf{l2g}_{\text{sd}\{1\}}(\text{gamma1}))$ .
- Das Gitter können Sie in wenigen Zeilen mit folgendem Code generieren:

```
linGitter_x = linspace(0,bx,N*n+1);
linGitter_y = linspace(0,1,n+1);
[xx,yy] = meshgrid(linGitter_x,linGitter_y);
x = [xx(:),yy(:)];
tri = delaunay(x(:,1),x(:,2));
```

Hierbei wird für  $[0, bx] \times [0, 1]$  ein Gitter mit  $2n^2$  Dreiecken für jedes der  $N$  Teilgebiete erstellt.

## Allgemeine Hinweise zum Programmiereteil

- Der Code **muss sinnvoll kommentiert** sein. Ein nicht kommentiertes Programm gilt als nicht erfolgreich bearbeitet.
- Das Programm muss ausführbar sein, ohne Änderungen am Code vornehmen zu müssen (d.h. ein Klick auf „Ausführen“ muss ausreichen). Schreiben Sie daher ein oder mehrere Skripte für die Teilaufgabe(n). Benennen Sie das Skript / die Skripte sinnvoll (z.B. `aufg1c.m`).
- Schreiben Sie bitte Funktionen in eigene Dateien und nicht in Skriptdateien (*Ausnahme*: anonyme Funktionen der Art  $\mathbf{f} = @(x) x.^2$ );
- Enthält ihr Code mehrere Funktionen, so ist jede Funktion in eine eigene Datei zu schreiben. *Ausnahme*: Die Funktion wird ausschließlich von anderen Funktionen derselben Datei aufgerufen. In diesem Fall steht an oberster Stelle der Funktionsdatei die Funktion, welche von außerhalb (z.B. von einem Skript) aufgerufen wird.

## Abgabe des Programmierteils

- Packen Sie Ihre Dateien in ein Archiv (Formate: .zip, oder .tar.gz) mit einem Dateinamen der Art:

`ueb01_nachname_vorname.zip`

- Den Quellcode schicken Sie bitte an die E-Mail-Adresse Ihrer Übungsgruppenleiter / Übungsgruppenleiterinnen, mit einem Betreff der Art:

**Betreff: Uebung1, Nachname, Vorname**

- Geben Sie bitte immer eine **ausgedruckte Version** Ihrer Programmcodes mit den schriftlichen Aufgaben ab (→ Kasten), sofern dies in der Aufgabenstellung nicht eindeutig anders vermerkt wurde.
- Sofern es zur sinnvollen Lösung der Aufgabenstellung nötig ist, drucken Sie bitte auch die Ausgabe von Matlab aus. Dies sollte nicht zwei DIN-A4-Seiten überschreiten. Gleiches gilt für Grafiken.

**Abgabe: Bis Mittwoch, 24. Oktober 2018, 16:00 Uhr, im entsprechenden Kasten in Raum 3.01 des Mathematischen Instituts.**