

## Übungsblatt 4

(Besprechung Mo. 18.11.2013)

### Aufgabe 9 (Randelement-Methode, Kollokation)

Sei  $\Omega \subset \mathbb{R}^2$  ein Gebiet mit Rand  $\Gamma := \partial\Omega$ . Wir betrachten die Laplace-Gleichung

$$\begin{aligned} \Delta u &= 0 && \text{in } \Omega \\ u &= u_D && \text{auf } \Gamma. \end{aligned} \tag{1}$$

Ausgangspunkt der Randelement-Methode ist die Darstellungsformel, mit der die Lösung  $u$  von (1) die folgende Darstellung hat:

$$u(x) = \underbrace{\int_{\Gamma} \log|x-y| \varphi(y) ds_y}_{=:(\mathcal{V}\varphi)(x)} - \underbrace{\int_{\Gamma} \frac{\partial}{\partial n} \log|x-y| u(y) ds_y}_{=:(\mathcal{K}u)(x)} \quad x \in \Omega, \tag{2}$$

wobei  $\varphi := \frac{\partial}{\partial n}u$  die Normalen-Ableitung von  $u$ ,  $\mathcal{V}$  das Einfachschichtpotential und  $\mathcal{K}$  das Doppelschichtpotential bezeichnet. Kennt man also die kompletten Cauchy-Daten  $(u, \varphi)$  auf dem Rand  $\Gamma$  so kann man die Lösung  $u$  im Gebiet  $\Omega$  berechnen. Für das Dirichlet-Problem (1) kann man folgende Rand-Integralgleichung herleiten, um die fehlenden Neumann-Daten zu bestimmen:

$$(\mathcal{V}\varphi)(x) = \left( \mathcal{K} + \frac{1}{2} \right) u(x), \quad x \in \Gamma \quad (\text{Symm's Integralgleichung}). \tag{3}$$

Um ein diskretes Problem zu erhalten, führen wir eine Triangulierung  $\mathcal{T}_h := \{T_i = \text{conv}\{x_i, x_{i+1}\}, i = 1, \dots, N\}$  des Randes  $\Gamma$  ein. Außerdem wählen wir für  $u$  als Ansatz-Funktionen die Hutfunktionen  $\psi_k$  mit  $\psi_k(x_j) = \delta_{i,j}$  und für die Lösung  $\varphi$  wählen wir stückweise konstante Funktionen  $\phi_j := \mathbb{1}_{T_j}$ , d.h.:

$$u(x) = \sum_{k=1}^N u_k \psi_k(x), \quad \varphi(x) = \sum_{k=1}^N \alpha_k \phi_k(x).$$

Testen wir (3) an den Mittelpunkten  $m_j$  der Elemente  $T_j$  ( $j = 1, \dots, N$ ) so erhalten wir das LGS

$$\mathbf{Ax} = \mathbf{b}$$

wobei  $\mathbf{x} = (\alpha_j)_{j=1, \dots, N}$ ,  $\mathbf{A} = (a_{j,k})_{i,j=1, \dots, N}$  mit

$$a_{j,k} := (V\phi_k)(m_j)$$

und  $\mathbf{b} = (b_j)_{j=1, \dots, N}$  mit

$$b_j = \sum_{k=1}^N u_k \left[ \left( K + \frac{1}{2} \right) \psi_k \right] (m_j)$$

- (i) Laden Sie sich die zip-Datei von der Homepage herunter und entpacken Sie diese. Tippen Sie im Command-Window von MATLAB

`mex inpoly.c`

ein (Dadurch wird die mex-Datei `inpoly.c` kompiliert, die wir später zum Darstellen der Lösung benötigen).

- (ii) Sei  $\Gamma := [-1, 1]$ . Werten Sie die Potentiale  $\mathcal{V}$  und  $\mathcal{K}$  für auf einem geeigneten Gitter, das  $\Gamma$  enthält, aus und plotten sie die Potentiale auf diesem Gitter. Verwenden Sie dazu die Funktionen `potV.m` und `potK.m`, die das Einfach- bzw. das Doppelschichtpotential auf einem Element  $T_k$  auswerten. Beide Funktionen werden wie folgt aufgerufen:

$$\begin{aligned} \mathbf{V} &= \text{potV}(\text{vertices}, \mathbf{x}, \mathbf{p}), \\ \mathbf{K} &= \text{potK}(\text{vertices}, \mathbf{x}, \mathbf{p}). \end{aligned}$$

Dabei sind in  $\text{vertices} \in \mathbb{R}^{2 \times 2}$  die Eckpunkte des Elements  $T_k$  (zeilenweise) gespeichert, in  $\mathbf{x} \in \mathbb{R}^{M \times 2}$  sind die  $M$  Punkte gespeichert, an denen das Potential ausgewertet werden soll und  $\mathbf{p}$  bezeichnet den Polynomgrad der Ansatzfunktionen (also bei uns für das Einfachschichtpotential  $p = 0$  und für das Doppelschichtpotential  $p = 1$ ). Zurückgegeben wird beim Einfachschichtpotential für  $p = 0$  der Vektor  $\mathbf{V} \in \mathbb{R}^M$  mit  $\mathbf{V}_j = (\mathcal{V}\phi_k)(x_j)$ . Beim Doppelschichtpotential wird für  $p = 1$  die Matrix  $\mathbf{K} \in \mathbb{R}^{M \times 2}$  zurückgegeben mit  $\mathbf{K}_{j,1} = (\mathcal{K}\psi_k|_{T_k})(x_j)$  und  $\mathbf{K}_{j,2} = (\mathcal{K}\psi_{k+1}|_{T_k})(x_j)$ .

Was fällt Ihnen an den Schaubildern auf?

- (iii) Vervollständigen Sie die Funktion `Kollokation.m`, in der die Kollokationsmethode für das Lshape-Beispiel aus dem letzten Übungsblatt programmiert werden soll. Stellen Sie die Matrix  $\mathbf{A}$  und die rechte Seite  $\mathbf{b}$  mit einer `for`-Schleife über alle Elemente auf (siehe Kommentare im Programmtext). Testen Sie Ihre Implementierung am Lshape-Beispiel.