Andreas Dedner
dedner@mathematik.uni-freiburg.de
Assistent: Julien Diaz

Übung zur Vorlesung

Numerik partieller Differentialgleichungen

WS 2006/2007 — Blatt 5 (01.12.2006, Abgabe 08.12.2006)

Auf der Internetseite zur Vorlesung befindet sich eine Datei (maske1.zip), welche einige Matlab Funktionen zur Verwaltung eines Dreiecksgitters beinhaltet. Das Programm kann mit dem Kommando `unzip maske1.zip` ausgepackt werden und legt ein Verzeichnis `maske1` an, welche einige .m Files enthält.
Machen Sie sich mit der Funktionsweise der Funktion `LagrangeInterpolation` vertraut - befindet sich in der Datei `LagrangeInterpolation.m`, welche zu Beginn eine kurze Dokumentation enthält. Lesen Sie auch die Datei `README`.

Aufgabe 1 (Lagrange Interpolation)
In der Zeile 60 wird `uh` mit Nullen belegt; konstruieren Sie stattdessen die Lagrangeinterpolation einer gegebene Funktion, d.h. für alle Knoten $c(i) = (c(i,1), c(i,2))$ des Gitters setze $uh(i) = \text{lsg\_square}(c_i)$, wobei die Funktion `lsg_square` in der Datei `lsg_square.m` definiert wird und das `square` Problem mit dem ersten Parameter der Funktion `LagrangeInterpolation` vom Benutzer festgelegt werden kann.
Interpolieren Sie nun eine weitere Funktion in `lsg_C0.m`, welche stetig aber nicht mehr C^1 in $[-1,1]^2$ sein sollte und eine unstetige Funktion (etwa die charakteristische Funktion ξ_{B_r}, wobei B_r der Kreis mit Radius R um den Nullpunkt sei — Sie können aber auch gern etwas anderes verwenden...
Als vierte Funktion implementieren Sie das Problem mit der einspringenden Ecke mit einem Winkel von 270 Grad (siehe Blatt 2)— `lsg_corner.m` sollte dabei die exakte Lösung implementieren und eine Zerlegung des Gebiets in Dreiecke konstruieren Sie in den Files `nodes_corner.dat`, `elements_corner.dat` und `dirichlet_corner.dat`.
Das Programm sollte bei richtiger Anwendung — etwa `LagrangeInterpolation('square',4,0,4)` ein Fenster mit 12 einzelnen Plots öffnen; für Details schauen Sie in das Programm `show.m`. Geben Sie diese für die vier oben angegebene Funktion ab.

Aufgabe 2 (EOC für der Lagrangeinterpolation)
Berechnen Sie den L^2 Fehler zwischen den in der letzten Aufgabe beschriebenen vier Funktionen u und der Interpolation u_h. Füllen Sie dazu den Vektor `error` mit den elementweise quadrierten L^2:

$$\texttt{error}(i) = \int_{T_i} (u(x) - u_h(x))^2 \, dx$$

für $i = 1, \dots, \texttt{NT}$. Dabei benutzen Sie folgende Quadratur auf jedem Element:

$$\int_{T_i} g(x) \, dx \approx |T_i| g(\omega(i));$$

dabei ist $\omega(i) := \sum_{k=1}^{3} c(e(i,k))/3$, d.h. das Mittel der Koordinaten der Knoten des Dreiecks. Die diskrete Funktion u_h an dieser Stelle wird ebenfalls durch die Mittelung der Werte an den drei Knoten berechnet — machen Sie sich unbedingt den Strukturen `grid.coordinate` und `grid.elements3` vertraut!
Wenn alles richtig gemacht wurde, sollte die Ausgabe für das `square` Problem nun einen EOC von ungefähr zwei aufweisen!

Aufgabe 3 (Dünnbesetzte Matrizen (engl. sparse))
Solche speichersparenden Matrizen sind sehr wichtig zur Lösung partieller Differenzialgleichungen. Machen Sie sich mit der Funktionsweise von `sparse` in Matlab vertraut — etwa (`help sparse`) oder (`http://www.math.mtu.edu/~msgocken/intro/node18.html`).
Schauen Sie die Datei `edge.m` an. Dort werden zum einen die Kanten (`grid.primedge`) berechnet aber auch weitere Strukturen wie etwa (`grid.dualedge`), welche benutzt werden kann, um etwa die beiden anliegenden Elemente an einer Kante zwischen Knoten `i` und `j` zu bestimmen.
`grid.dualedge` wird mittels einer Sparsematrix berechnet (Zeile 11). Schreiben sie eine For-Schleife, welche die Matrix äquivalent zu dieser Zeile anlegt.

Aufgabe 4

(a) Zeigen Sie, dass eine auf einem Dreieckgitter stückweise quadratische Funktion und global stetige Funktion, eindeutig durch Wert auf den Knoten und auf den Kantenmittelpunkten festgelegt wird. Stimmt das auch, wenn wir nur zwei Werte pro Kante festlegen?

(b) Wir bezeichnen mit $\hat{T}$ das Referenzdreieck. Zeigen Sie, dass die folgenden Quadraturen auf angegebenen polynomialen Funktionenrumen exakt sind (vgl. Blatt 4):

   a) $\hat{w}_1 = \frac{1}{2}$ und $\hat{\mathbf{x}}_1 := (\frac{1}{3}, \frac{1}{3})^T$ auf $P_1(\hat{T})$

   b) $(\hat{w}_1, \hat{w}_2, \hat{w}_3) := (\frac{1}{6}, \frac{1}{6}, \frac{1}{6})$ und $(\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \hat{\mathbf{x}}_3) := \left((\frac{1}{2}, 0)^T, (\frac{1}{2}, \frac{1}{2})^T, (0, \frac{1}{2})^T\right)$ auf $P_2(\hat{T})$

(c) Wir betrachten das Referenzquadrat $\hat{T} := [0,1]^2 \subset \mathbb{R}^2$ für Vierecksgitter, dessen Ecken mit $\hat{\mathbf{p}}_0 = (0,0)^T, \hat{\mathbf{p}}_1 = (1,0)^T, \hat{\mathbf{p}}_2 = (0,1)^T$ und $\hat{\mathbf{p}}_3 = (1,1)^T$ bezeichnet sind. Es sei $Q_s(\mathbb{R}^n) := \operatorname{span}\left\{\prod_{i=1}^{n} x_i^{d_i} | d_i \in \mathbb{N}_0, d_i \leq s, i = 1, \ldots, n\right\}$ und $Q_s(\hat{T})$ die Menge dieser Funktionen eingeschränkt auf $\hat{T}$.

   a) Geben Sie eine *nodale Basis* $\{\hat{\varphi}_i\}_{i=0}^{3}$ für $Q_1(\hat{T})$ an, d.h. $\hat{\varphi}_i(\hat{\mathbf{p}}_j) = \delta_{ij}$ für $i, j = 0, \ldots, 3$.

   b) Als Referenzabbildung auf ein beliebiges Viereck T mit Ecken $\mathbf{p}_i \in \mathbb{R}^2, i = 0, \ldots, 3$ reichen affine Abbildungen nicht aus, sondern es wird $F_T \in Q_1(\hat{T})^2$ gewählt. Zeigen Sie, dass F_T in diesem Raum eindeutig ist.