Albert-Ludwigs-Universität Freiburg
Abteilung für Angewandte Mathematik
Prof. Dr. Patrick Dondl
Johannes Daube
Hannes Eberlein

WS 2015/16

Theorie und Numerik partieller Differentialgleichungen

Blatt 9

Abgabe: 22. Dezember 2015

Aufgabe 33 (4 Punkte). *Approximationsfehler I*

Es sei $\theta\colon [0,T] \to S_h$ gegeben als

$$\theta(t) = u_h(t) - R_h u(t),$$

wobei u und u_h die Lösung der Wärmeleitungsgleichung bzw. ihrer (zeitkontinuierlichen) Galerkinapproximation darstellen und R_h die Ritz-Projektion auf S_h ist.

(i) Zeigen Sie, dass aus

$$\frac{1}{2}\frac{d}{\mathrm{d}t}\|\theta\|^2_{L^2(\Omega)} \leq \|\rho'\|_{L^2(\Omega)}\|\theta\|_{L^2(\Omega)}$$

folgt

$$\|\theta(t)\|_{L^2(\Omega)} \leq \|\theta(0)\|_{L^2(\Omega)} + \int_0^t \|\rho'\|_{L^2(\Omega)}\,\mathrm{d}s.$$

(ii) Es sei nun θ^n die zeitdiskrete Variante aus dem Eulerverfahren. Zeigen Sie, dass aus

$$\|\theta^n\|^2_{L^2(\Omega)} - (\theta^{n-1}, \theta^n)_{L^2(\Omega)} \leq k\|\eta^n\|\|\theta^n\|$$

folgt

$$\|\theta^n\|_{L^2(\Omega)} \leq \|\theta^0\|_{L^2(\Omega)} + k\sum_{j=1}^n \|\eta^j\|_{L^2(\Omega)}$$

(iii) Zeigen Sie, dass (nun für Crank-Nicolson) aus

$$\|\theta^n\|^2_{L^2(\Omega)} - \|\theta^{n-1}\|^2_{L^2(\Omega)} \leq k\|\eta^n\|_{L^2(\Omega)}(\|\theta^n\|_{L^2(\Omega)} - \|\theta^{n-1}\|_{L^2(\Omega)})$$

ebenso folgt dass

$$\|\theta^n\|_{L^2(\Omega)} \leq \|\theta^0\|_{L^2(\Omega)} + k\sum_{j=1}^n \|\eta^j\|_{L^2(\Omega)}$$

Aufgabe 34 (4 Punkte). *Approximationsfehler II*

Zeigen Sie, dass gilt

$$(\overline{\partial}\theta^n, w)_{L^2(\Omega)} + (\nabla\hat{\theta}^n, \nabla w)_{L^2(\Omega)} = (-\eta^n, w)_{L^2(\Omega)} \quad \text{für } w \in S_h,\ n \geq 1$$

mit

$$\eta^n = (R_h - \mathrm{Id})\overline{\partial}u(t_n) + (\overline{\partial}u(t_n) - u'(t_{n-\frac{1}{2}})) + \Delta\left(u(t_{n-\frac{1}{2}}) - \frac{1}{2}(u(t_n) + u(t_{n-1}))\right).$$

Die Notation entspricht der aus der Vorlesung im Beweis des Satzes 4.12.

Aufgabe 35 (8 Punkte). *Crank-Nicolson*

Implementieren Sie die Crank-Nicolson-Methode in `heat_1.m` (gerne auch in einer anderen Programmiersprache) und überprüfen Sie experimentell die Fehlerabschätzung. Es genügt ein Graph.