Programmierblatt 4.

Bearbeiten bis: **Sonntag, 11.12.2022**

Nachdem wir uns bisher mit dem stationären, elliptischen Poisson-Problem befasst haben, möchten wir uns auf dem letzten Programmierblatt der Lösung von zeitabhängigen Problemen widmen. Exemplarisch betrachten wir dazu ein parabolisches Problem, die *Wärmeleitungsgleichung*.

Wärmeleitungsgleichung

Auf einem Gebiet $\Omega \subset \mathbb{R}^2$ und einem Intervall $[0, T]$ mit $T < \infty$ betrachten wir die Gleichung

$$\frac{\partial}{\partial t} u(\mathbf{x}, t) - \Delta u(\mathbf{x}, t) = f(\mathbf{x}, t), \qquad (\mathbf{x}, t) \in \Omega \times (0, T]. \tag{1}$$

Dabei entspricht $f(\mathbf{x}, t): \ \Omega \times (0, T] \longrightarrow \mathbb{R}$ einer Wärmequelle. Die Lösung $u(\mathbf{x}, t)$ der Gleichung beschreibt dann die Temperaturverteilung zu jedem Zeitpunkt $t \in (0, T]$ innerhalb von Ω. Zur eindeutigen Lösbarkeit der Gleichung müssen entsprechende Anfangs- und Randbedingungen vorgegeben werden, wobei wir uns der Einfachheit halber auf homogene Dirichlet-Randbedingungen beschränken.[1] Damit lautet die vollständige Formulierung

$$\begin{aligned} \frac{\partial}{\partial t} u(\mathbf{x}, t) - \Delta u(\mathbf{x}, t) &= f(\mathbf{x}, t) && \text{für } \mathbf{x} \in \Omega, t \in (0, T], \\ u(\mathbf{x}, 0) &= u_0(\mathbf{x}) && \text{für } t = 0, \\ u(\mathbf{x}, t) &= 0 && \text{auf } \Gamma \times (0, T]. \end{aligned} \tag{2}$$

Ortsdiskretisierung.

Für einen vorgegebenen Finite-Elemente-Raum V_h ergibt sich durch die Diskretisierung der Gleichung im Ort die gewöhnliche Differentialgleichung

$$\mathbf{M} \frac{\partial}{\partial t} \mathbf{u}(t) + \mathbf{A}\mathbf{u}(t) = \mathbf{f}(t).$$

Hierbei ist $\mathbf{M} = [\langle \phi_i, \phi_j \rangle_{L^2(\Omega)}]_{i,j}$ die Massenmatrix und $\mathbf{A} = [\langle \nabla\phi_i, \nabla\phi_j \rangle_{L^2(\Omega)}]_{i,j}$ die Steifgkeitsmatrix sowie $\mathbf{f}(t) = [\langle f(t), \phi_i \rangle_{L^2(\Omega)}]_i$ die zeitabhängige rechte Seite.

Zeitdiskretisierung

Die Zeitableitung wird nun entweder mithilfe einer Vorwärts- oder Rückwärtsdifferenz realisiert. Hierzu sei $\{t_0, t_1 \dots t_N\}$ mit $t_0 = 0$, $t_N = T$ und $t_i = ki$ mit $k = T/N$ ein regelmässiges Gitter für das Zeitintervall $[0, T]$. Bei der Vorwärtsdifferenz ergibt sich das explizite Euler-Verfahren

$$\mathbf{M}\mathbf{u}_{i+1}^{\text{expl}} = (\mathbf{M} - k\mathbf{A})\mathbf{u}_i^{\text{expl}} + k\mathbf{f}(t_i), \tag{3}$$

die Rückwärtsdifferenz führt analog auf das implizite Euler-Verfahren

$$(\mathbf{M} + k\mathbf{A})\mathbf{u}_{i+1}^{\text{impl}} = \mathbf{M}\mathbf{u}_i^{\text{impl}} + k\mathbf{f}(t_{i+1}). \tag{4}$$

[1] Die homogenen Dirichlet-Daten entsprechen dem Vorschreiben der konstanten Temperatur null am Gebietsrand.

In beiden Fällen ergibt sich der Startvektor $\mathbf{u}_0$ für nodale finite Elemente aus der Interpolation von u_0 in den Gitterpunkten.

Durch bilden einer Konvexkombination dieser beiden Approximationen (3) und (4) ergibt sich das θ-Schema

$$\mathbf{u}_{i+1}^{\theta} = \theta \mathbf{u}_{i+1}^{\text{impl}} + (1-\theta)\mathbf{u}_{i+1}^{\text{expl}},$$

wessen Iterierte sich direkt gemäss

$$(\mathbf{M} + k\theta\mathbf{A})\mathbf{u}_{i+1}^{\theta} = \big(\mathbf{M} - k(1-\theta)\mathbf{A}\big)\mathbf{u}_{i}^{\theta} + k(1-\theta)\mathbf{f}(t_i) + k\theta\mathbf{f}(t_{i+1}). \tag{5}$$

berechnen lasen. Das θ-Schema ist für $\theta = 1/2$ konsistent von der Ordnung 2.

Aufgabe 1. Schreiben Sie eine Funktion

```
function [us, ts] = solve_heat_equation(mesh, f, u0, T, k, theta),
```

welche die Wärmeleitungsgleichung mithilfe des θ-Schemas gemäss (5) löst. Zurückgegeben werden soll dabei eine Matrix `us`, welche in der j-ten Spalte die Koeffizienten der Funktion $u(\cdot, t_j)$ enthält, sowie ein Zeilenvektor `ts`. Die Funktionen u_0 und f sollen dabei als `function handle` übergeben werden. Zur Lösung der linearen Gleichungssysteme in jedem Zeitschritt soll das diagonal vorkonditionierte CG-Verfahren mit $\varepsilon = 10^{-8}$ verwendet werden.

Aufgabe 2. Schreiben Sie eine Funktion Testen Sie ihre Implementierung auf dem siebenmal verfeinerten Einheitsquadrat. Wählen Sie dazu $T = 0.5$, $k = 10^{-2}$, $u_0 \equiv 0$ und

$$f(\mathbf{x}, t) = \begin{cases} 100, & \text{falls } (x_1 - 0.5)^2 + (x_2 - 0.5)^2 < 0.1, \\ 0, & \text{sonst.} \end{cases}$$

Ergänzen Sie die Funktion `visfunc` und stellen Sie für $\theta \in \{0.4, 0.5, 0.6\}$ die Lösung in einer Animation dar.

Aufgabe 3. Schreiben Sie eine Funktion

```
function [us, ts] = solve_heat_equation_mg(mgmesh, f, u0, T, ...
                                           k, theta, K, R, alpha),
```

welche die Wärmeleitungsgleichung mithilfe des θ-Schemas löst. Zur Lösung der linearen Gleichungssysteme in jedem Zeitschritt soll hier ein Mehrgitterverfahren mit R V-Zyklen verwendet werden. Testen Sie Ihre Implementierung auf dem siebenmal verfeinerten Einheitsquadrat mit einem Glätter Ihrer Wahl und stellen Sie die Lösung in einer Animation dar.