Universität Konstanz
Fachbereich Mathematik und Statsitik
Johannes Schropp
Jan Rohleff

Wintersemester 2024/2025

Numerik partieller Differentialgleichungen I

`https://www.math.uni-konstanz.de/~rohleff/numpde1.html`

2. Übungsblatt

Ausgabe: 19.12.2024, Abgabe: 09.01.2025

Vorbemerkung zu den Programmieraufgaben

Die Programmieraufgaben können in Zweiergruppen bearbeitet werden. Dabei besteht die Option, die Aufgaben entweder in Matlab oder Python zu lösen. Dokumentieren Sie ihre Ergebnisse in einem Report als PDF File. Die fertigen Programme mit Reports sollten anschließend per E-Mail an jan.rohleff@uni-konstanz.de gesendet werden. Es ist wichtig, in den Programmen alle Schritte angemessen zu kommentieren, um die Nachvollziehbarkeit und Verständlichkeit zu gewährleisten.

Aufgabe 2.1 (Theorie - 6 Punkte)

Gegeben sei die Aufgabe

$$\begin{aligned} -\Delta v &= \lambda v \text{ in } \Omega, \\ v|_{\partial\Omega} &= 0. \end{aligned} \tag{1}$$

Eine Lösung $v \in C^2(\Omega) \cap C^0(\bar{\Omega}), v \neq 0$ heißt Eigenfunktion zum Eigenwert λ.
a) Zeigen Sie: Für jeden Eigenwert λ von (1) gilt $\lambda > 0$.
b) Seien v_1, v_2 Eigenfunktionen zu Eigenwerten λ_1 und λ_2, $\lambda_1 \neq \lambda_2$. Dann sind v_1, v_2 orthogonal bezüglich des Skalarproduktes

$$\langle u, w \rangle = \int_\Omega u(x)w(x)dx.$$

c) Berechnen Sie die Eigenwerte von (1) im Fall $\Omega =]0,1[^2$ explizit und vergleichen Sie sie mit denen von Aufgabe 3, Blatt 7. Verwenden Sie hierzu den *Separationsansatz* $v(x_1, x_2) = w(x_1)z(x_2)$.

Aufgabe 2.2 (Theorie - 6 Punkte)

Vorgelegt sei das lineare Randwertproblem

$$\begin{cases} -u''(x) + b(x)u'(x) + c(x)u(x) = f(x) \text{ für alle } x \in (0,1), \\ u(0) = \alpha,\ u(1) = \beta \end{cases} \tag{2}$$

für auf dem abgeschlossenen Intervall $[0,1] \subset \mathbb{R}$ stetigen Funktionen b, c, f mit $c > 0$, sowie $\alpha, \beta \in \mathbb{R}$. Diskretisieren Sie das Problem (2) für eine Schrittweite $h = \frac{1}{N}$, $N \in \mathbb{N}$, und approximieren Sie $u''(x)$ und $u'(x)$ mittles *zentraler Differenzenquotienten.* Stellen Sie das zugehörige lineare Gleichungssystem auf und untersuchen Sie dieses auf Lösbarkeit. Wann ist die Koeffizientenmatrix des linearen Gleichungssystems zeilendiagonaldominant? Was ändert sich, wenn $u'(x)$ mittels sogenanntem *Aufwind-Differenzenquotient* approximiert wird:

$$u'(x) \approx \begin{cases} \frac{u(x+h)-u(x)}{h} & \text{falls } b(x) < 0, \\ \frac{u(x)-u(x-h)}{h} & \text{falls } b(x) \geq 0. \end{cases} \tag{3}$$

Hinweis: Die $n \times n$ Matrix $A = (a_{ik})_{i=1\ldots n, k=1\ldots n}$ heißt *zeilendiagonaldominant*, falls gilt

$$|a_{ii}| > \sum_{\substack{k=1 \\ k \neq i}}^{n} |a_{ik}| \qquad (i = 1, 2, ..., n),$$

(vgl. Skript Numerik I, Seite 16).

Bemerkung: Die Differenzenapproximation (3) mittles Aufwind-Differenzenquotient ist insbesondere bei singulär gestörten Problemen sehr hilfreich.

Aufgabe 2.3 (Programm - 8 Punkte)

Es sei $\Omega = (a, b)^2$ und $h = \frac{b-a}{M}$ mit $M \in \mathbb{N}$. Lösen Sie numerisch die Aufgabe

$$\begin{aligned} -\Delta u &= f(x, y) \quad \text{in} \quad \Omega, \\ u &= g(x, y) \quad \text{auf} \quad \partial\Omega, \end{aligned}$$

mit dem klassischen Differenzenverfahren für $a = 0$ und $b = 1$ und folgenden Funktionen f und g:

1. $f(x, y) = 0,$ $\qquad g(x, y) = y \cos(4\pi x)$
2. $f(x, y) = \begin{cases} 1, & \text{falls } |x - 0.5| + |y - 0.5| \leq 2, \\ 0, & \text{sonst.} \end{cases}$ $\qquad g(x, y) = 0$

Verwenden Sie für die Diskretisierung von Ω die natürliche, zeilenweise von links unten nach rechts oben verlaufende Nummerierung der Gitterpunkte. Schreiben Sie hierzu eine Matlab/Python Funktion `cfdm`

```
function [U] = cfdm(M,a,b,hf,hg)
```

die eine Lösungsmatrix `U` $\in \mathbb{R}^{(M+1)\times(M+1)}$ entsprechend der Diskretisierung (inklusive der Werte auf dem Rand!) zurückgibt. Als Übergabewerte soll die Funktion **ausschließlich und in dieser Reihenfolge** die Anzahl an diskreten Gitterabschnitten für eine Koordinatenrichtung `M` (siehe Vorlesung), die Intervallgrenzen `a` und `b`, sowie Function Handles `hf` und `hg` für die Funktionen f und g akzeptieren. Rufen Sie diese Funktion in einem Main-File `main` für gegebene Funktionen f und g sowie verschiedene M auf und visualisieren Sie die erhaltene Lösung u sowie die dazu gehörige Funktion f auf dem Gitter. Was beobachten Sie?

Anmerkung: Folgende Matlab Befehle können hilfreich sein: `ndgrid`, `spdiags`, `reshape`, `sparse`, `find`.
Die korrespondierenden Python befehle lauten `numpy.meshgrid`, `scipy.sparse.diags`, `numpy.reshape`, `scipy.sparse`, `numpy.nonzero`.