MATHEMATISCHES INSTITUT
DER UNIVERSITÄT ZU KÖLN

Prof. Dr. A. Klawonn
J. Knepper, M. Sc.

Wintersemester 2017/2018

20. Dezember 2017

11. Übung zur Einf. in die Numerik partieller Differentialgleichungen

Hinweis: Schreiben Sie bitte auf **jedes Blatt ihren Namen**. Auf die erste Seite Ihrer Übung schreiben Sie bitte zusätzlich Ihre Matrikelnummer und die Nummer der Ihnen zugewiesenen Übungsgruppe.

Aufgabe 1: ($3 + 6 = 9$ Punkte)
Wir betrachten zum Modellproblem

$$\begin{aligned} u_t &= u_{xx}, && (x,t) \in (0,1) \times (0,\infty), \\ u(0,t) = u(1,t) &= 0, && t \geq 0, \\ u(x,0) &= \phi(x), && x \in [0,1]. \end{aligned}$$

das Verfahren

$$\frac{u^h_{i,j+1} - u^h_{i,j-1}}{2k} = \frac{u^h_{i-1,j} - 2u^h_{i,j} + u^h_{i+1,j}}{h^2}.$$

1. Leiten Sie das Verfahren her und zeigen Sie, dass in einem Schritt der Diskretisierungsfehler $\mathcal{O}(k^3) + \mathcal{O}(h^2 k)$ entsteht.
2. Zeigen Sie, dass das Verfahren nicht von-Neumann-stabil ist.

Aufgabe 2: (3 Punkte)
Zum folgenden Anfangsrandwertproblem der Wärmeleitungsgleichung,

$$\begin{aligned} \frac{\partial u}{\partial t}(x,t) &= \frac{\partial^2 u}{\partial x^2}(x,t), && (x,t) \in (a,b) \times (0,\infty), \\ u(x,0) &= \phi(x), && x \in [0,\pi], \\ u(a,t) &= \psi_a(t), && t > 0, \\ u(b,t) &= \psi_b(t), && t > 0, \end{aligned}$$

betrachten wir das implizite Differenzenverfahren

$$\frac{u^h_{i,j+1} - u^h_{i,j}}{k} = \frac{u^h_{i-1,j+1} - 2u^h_{i,j+1} + u^h_{i+1,j+1}}{h^2}.$$

Das Intervall $[a,b]$ sei durch $a = x_0 < x_1 < \cdots < x_N < x_{N+1} = b$ äquidistant diskretisiert. Für den j-ten Zeitschritt definieren wir

$$u^h_j := (u_{1,j}, u_{2,j}, \ldots, u_{N,j})^T.$$

Zeigen Sie, dass bei Anwendung des Differenzenverfahrens die numerische Lösung im $(j+1)$-ten Schritt durch Lösung eines linearen Gleichungssystems bestimmt wird, d.h., zeigen Sie, dass eine Matrix $B \in \mathbb{R}^{N \times N}$ (unabhängig von j) und ein Vektor $f_j \in \mathbb{R}^N$ existiert, sodass

$$Bu^h_{j+1} = f_j$$

gilt. Ist das System eindeutig lösbar?

Aufgabe 3: (5 Bonuspunkte)
Gegeben sei eine Zeitschrittweite $\Delta t > 0$, eine Raumschrittweite $\Delta x > 0$ und die Matrix

$$A = \frac{1}{(\Delta x)^2} \begin{pmatrix} 2 & -1 & 0 & \dots & 0 \\ -1 & 2 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & 2 & -1 \\ 0 & \dots & 0 & -1 & 2 \end{pmatrix}.$$

Zeigen Sie, dass $(I + \Delta t A)$ symmetrisch und positiv definit ist.

Programmieraufgabe: (15 Punkte)

Folgen Sie den Hinweisen aus der Vorlesung zur Codestruktur und grafischen Ausgabe. Dies wird für eine erfolgreiche Bearbeitung vorausgesetzt.

Vermeiden Sie insbesondere Copy-Paste-Codeteile. Schreiben Sie dazu die Anwendung des Differenzenverfahrens in eine eigene Funktion.

Schreiben Sie ein Programm in Matlab, um numerisch die Wärmeleitungsgleichung

$$u_t = u_{xx}$$

für $(x,t) \in [0,\pi] \times [0,T]$, $T = 5$, mit den Anfangswerten

$$u(x,0) := \begin{cases} 0 & , \quad \text{falls } x \leq \frac{\pi}{2}, \\ 1 + \sin\left(4 \cdot \left(x - \frac{5}{8}\pi\right)\right) & , \quad \text{sonst,} \end{cases}$$

und den Randwerten $u(0,t) = u(\pi,t) = 0$, $t \in [0,T]$, mit dem klassischen expliziten Differenzenverfahren (Zeit-Vorwärts / Raum-Zentriert)

$$\frac{u^h_{i,j+1} - u^h_{i,j}}{k} = \frac{u^h_{i+1,j} - 2u^h_{i,j} + u^h_{i-1,j}}{h^2}$$

zu lösen. Testen Sie ihre Implementierung mit $\Delta x = \frac{\pi}{100}$ sowie $\frac{\Delta t}{(\Delta x)^2} = \lambda \in \{0.1, 0.5, 0.51\}$. Geben Sie jeweils die benötigte Anzahl an Zeitschritten sowie das Maximum der numerischen Lösung im letzten Zeitschritt, $\max_i u^h(x_i, T)$, an. Sie brauchen die Grafik für $\lambda = 0.1$ nicht auszudrucken. **Erklären** Sie das Ergebnis.

Hinweise:

- Testen Sie Ihr Programm zunächst an einer kleineren Ortsschrittweite, z.B. $\Delta x = \pi/20$.
- Beachten Sie beim Plotten die Hinweise aus der Vorlesung zum Downsampling. Beschreibt `t` die Diskretisierung von $[0,T]$, so können Sie mit

  ```
  ma = 200;  % maximale Anzahl an Punkten auf der Achse
  step = ceil(length(t)/ma);
  ind_t = 1:step:length(t);
  ```

  eine Teilindizierung für die t-Achse erzeugen und mit

  `mesh(x,t(ind_t),uh(ind_t,:))` (Befehl an Ihr Programm anpassen!)

  einen Lösungsplot erzeugen.

Allgemeine Hinweise zum Programmierteil

- Der Code muss sinnvoll kommentiert sein.
- Das Programm muss ausführbar sein, ohne Änderungen am Code vornehmen zu müssen (d.h. ein Klick auf „Ausführen“ muss ausreichen). Schreiben Sie daher ein oder mehrere Skripte für die Teilaufgabe(n). Benennen Sie das Skript / die Skripte sinnvoll (z.B. `aufg1c.m`).
- Schreiben Sie bitte Funktionen in eigene Dateien und nicht in Skriptdateien (*Ausnahme:* anonyme Funktionen der Art `f = @(x) x.^2;`).
- Enthält ihr Code mehrere Funktionen, so ist jede Funktion in eine eigene Datei zu schreiben. *Ausnahme:* Die Funktion wird ausschließlich von anderen Funktionen derselben Datei aufgerufen. In diesem Fall steht an oberster Stelle der Funktionsdatei die Funktion, welche von außerhalb (z.B. von einem Skript) aufgerufen wird.

Abgabe des Programmierteils

- Packen Sie Ihre Dateien in ein Archiv (Formate: .zip, oder .tar.gz) mit einem Dateinamen der Art:

  `ueb01_nachname_vorname.zip`

- Den Code und das ausführbare Programm schicken Sie bitte an die E-Mail-Adresse Ihres Übungsgruppenleiters / Ihrer Übungsgruppenleiterin, mit einem Betreff der Art:

  `Betreff: Uebung1, Nachname, Vorname`

- Geben Sie bitte immer eine **ausgedruckte Version** Ihrer Programmcodes mit den schriftlichen Aufgaben ab (→ Kasten), sofern dies in der Aufgabenstellung nicht eindeutig anders vermerkt wurde.
- Sofern es zur sinnvollen Lösung der Aufgabenstellung nötig ist, drucken Sie bitte auch die Ausgabe von Matlab aus. Dies sollte nicht zwei DIN-A4-Seiten überschreiten. Gleiches gilt für Grafiken.

Abgabe: Bis Mittwoch, 10. Januar 2018 , 12:00 Uhr. Im entsprechenden Kasten in Raum 3.01 des Mathematischen Instituts.

Wir wünschen allen Studierenden frohe Weihnachten und einen guten Rutsch ins Jahr 2018.