Institut für Numerische und Angewandte Mathematik 14.04.2011
FB Mathematik und Informatik der Universität Münster
Prof. Dr. M. Ohlberger, Martin Drohmann, Michael Schaefer

Arbeitsblatt zum Praktikum

Numerik Partieller Differentialgleichungen I

SS 2011 — Blatt 0

Abgabe: 21.04.2011 per Email

Anwesenheitsaufgabe 0.1 (Uni-Dateisystem)
Wechseln Sie in das Verzeichnis `/bin` und lassen Sie sich den Inhalt anzeigen. Dort befindet sich ein Programm `mail`. Öffnen Sie die Anleitung zu diesem Programm mit `man mail` und schreiben Sie anschließend hiermit eine Email an die Adresse `mdrohmann@uni-muenster.de` mit ihrem Namen, ihrer Fachsemesteranzahl, ihrer Matrikelnummer und einer Selbsteinschätzung ihrer Programmierkenntnisse.

Anwesenheitsaufgabe 0.2 (Anpassen von Umgebungsvariablen)

(a) Schauen Sie sich den Inhalt der Variablen `$PATH` an

(b) Setzen Sie die Umgebungsvariable `PRAKTHP` auf den Wert `http://wwwmath.uni-muenster.de/num/Vorlesungen/Praktikum_SS11/Dateien` und schreiben Sie diese Definition in die Datei `~/.bashrc`.

(c) Ändern Sie die Umgebungsvariable `PS1`, so dass ihnen in der Bash immer ihr aktuelles Verzeichnis angezeigt wird. Die Funktionsweise der Variablen wird ausführlich in `man bash` erklärt.

Aufgabe 0.3* (Weitergehende Bash-Funktionalität)
Lesen Sie Man-Pages der in den “Gegenbeispielen” verwendeten Befehl, und versuchen Sie die Funktionsweise der Befehle nachzuvollziehen oder zu verändern.

Anwesenheitsaufgabe 1 (Verwendung von “mygrid”)
Erstellen Sie die Dokumentation zu der Bibliothek `mygrid` und machen Sie sich damit vertraut, was bereitgestellt wird, wie ein Gitter erstellt werden kann, und wie sich einzelne Gitterzellen verfeinern lassen.
Versuchen Sie anschließend die Lücken in der Datei `gridtest_template.cc` auszufüllen, und das Ergebnis mit `gnuplot` zu visualisieren.

Aufgabe 2 (Finite Differenzen Verfahren für Erhaltungsgleichungen)
Wir betrachten eine skalare Erhaltungsgleichung in einer Raum-Dimension

$$\begin{aligned}
\partial_t u(x,t) + \partial_x f(u(x,t)) &= 0 \quad \text{in } \Omega \times [0,T] \\
u(x,0) &= u_0(x) \text{ in } \Omega \\
u(x,t) &= u_{dir}(x,t) \text{ in } \Gamma_{dir}.
\end{aligned}$$

Hierbei bezeichnet $T > 0$ die Endzeit, $\Omega = [a,b] \subset \mathbb{R}$ ein beschränktes Intervall, $f \in C^1(\mathbb{R})$ die Flussfunktion, u_0 stückweise stetige, beschränkte Anfangsdaten und u_{dir} beschränkte Dirichlet-Randdaten, die auf dem Einfluss-Rand $\Gamma_{dir}(t) := \{x \in \partial\Omega | f'(u)n < 0\}$ mit äußerer Einheitsnormalen n vorgeschrieben werden. Der übrige Rand $\partial\Omega \backslash \Gamma_{dir}$ ist Ausfluss-Rand auf dem keine Bedingungen vorgegeben werden.
Die Diskretisierung dieser Gleichung soll mittels eines explizitem Finite Differenzen Verfahren realisiert werden. Hierzu sei K die Anzahl der Zeitintervalle, $\Delta t := T/K$ die Zeitschrittweite und es bezeichnen $t^k := k\Delta t, k = 0, \ldots, K$ die diskreten Zeitpunkte. Es seien die N Gitterpunkte im Ort gegeben als $G := \{x_n\}_{n=1}^N$ mit $a = x_1 < \ldots < x_N = b$. Das Gitter G definiert den Raum der reellwertigen Gitterfunktionen $V_h := \{v_h : \{x_n\}_{n=1}^N \to \mathbb{R}\}$. Durch das Finite Differenzen Verfahren wird eine Sequenz von diskreten Funktionen $u_h^k \in V_h$ generiert, indem eine Anfangsprojektion $u_h^0 := P_h(u_0)$ durchgeführt wird und anschließend sequenziell $u_h^{k+1} := u_h^k + \Delta t L_h(u_h^k)$ mit dem Ortsdiskretisierungsoperator $L_h : V_h \to V_h$ berechnet wird. Der Projektionsoperator und Ortsdiskretisierungsoperator sind punktweise definiert durch $\left(P_h(u_0)\right)(x_n) := u_0(x_n)$ und $(L(v_h))(x_n) := -\left(\frac{1}{\Delta x_n}[g(v_n, v_{n+1}) - g(v_{n-1}, v_n)]\right)$ mit numerischem Fluss g, Ortsschrittweite Δx_n und je nach Randtyp sinnvoll definierten Werten v_0 und v_{N+1}.
In ihrem Arbeitsverzeichnis zum Praktikum befinden sich einige Dateien mit Schablonen für Klassen und Methoden, die Sie selbst implementieren müssen:

Model in der Datei `model.hh` ist eine Sammlung der Datenfunktionen der Differentialgleichung, d.h. diese Klasse erlaubt Punktauswertung der Flussfunktion, Anfangs- und Randwerte.

InitialProjection in der Datei `finite_difference.hh` ist eine Klasse, die eine Projektion der Anfangsdaten $v_h := P_h[u_0]$ durchführt.

DiscreteSpaceOperator in der Datei `finite_difference.hh` ist eine Klasse, die die Anwendung des diskreten Ortsoperators $v_h = L_h[w_h]$ realisiert. Diese sollte so implementiert werden, dass der numerische Fluss ausgetauscht werden kann.

(a) Schreiben Sie zunächst in wenigen Zeilen Pseudo-Code ein Hauptprogramm in die Datei `finite_difference.cc`, das mit Hilfe der Klassen die Finite Differenzen Simulation durchführt.

(b) Implementieren Sie die fehlenden Methoden in den bereitgestellten Dateien. Für die Modellgleichungen sollten die folgenden Funktionen verwendet werden:

$$f(u) = \frac{1}{2}u^2 \tag{1}$$

$$u_0(x) = \begin{cases} -\frac{1}{2}x + \frac{3}{2} & \text{für } x \in [1,3] \\ 0 & \text{sonst} \end{cases} \tag{2}$$

$$u_{dir}(x,t) = 0. \tag{3}$$

Zur Verifikation der Numerik lautet die exakte Lösung auf dem Gebiet $\Omega = [a,b] = [0,5]$ mit Endzeit $T = 4$

$$u(x,t) = \begin{cases} 0 & \text{für } x < 1 \\ \frac{1}{t}(x-1) & \text{für } 1 \leq x < x_s(t) \\ \frac{1}{2-t}(x-3) & \text{für } x_s(t) \leq x < 3, t \in [0,2) \\ 0 & \text{für } 3 \leq x, t \in [0,2) \\ 0 & \text{für } x_s(t) \leq x, 2 \leq t \end{cases}$$

mit der Kurve

$$x_s(t) = \begin{cases} 1+t & \text{für } 0 \leq t < 2 \\ 1+\sqrt{2t} & \text{für } 2 \leq t. \end{cases}$$

Als erste Implementierung einer Flussfunktion kann ein Lax-Friedrichs-Fluss implementiert werden:

$$g(u,v) = \frac{1}{2}\left(f(u) + f(v) + u - v\right)$$

(c) Bei der Implementierung sollten Sie die die Funktionsweise ihres Codes mit dem Tool `doxygen` dokumentieren.