



Institut für Numerische Simulation der Universität Bonn
Prof. Dr. Mario Bebendorf

Praktikum im Sommersemester 2012

Programmierpraktikum numerische Algorithmen (P2E1)
(Numerische Lösung der Wärmeleitungsgleichung)
Betreuer: Christian Kuske¹

Blatt 6

1 Einführung

Mit der auf Blatt 5 eingeführten Mehrgitter-Methode und deren Modifikationen lassen sich große lineare Gleichungssysteme effizient lösen. Das Ziel ist die numerische Berechnung einer Lösung der Wärmeleitungsgleichung, die auf Blatt 1 vorgestellt wurde. Dabei wurde zunächst der stationäre Fall betrachtet. Auf diesem Blatt werden Verfahren zur Behandlung der zeitabhängigen partiellen Differentialgleichung beschrieben.

2 Zeitdiskretisierung

Die folgenden Zeitdiskretisierungen basieren auf dem Verfahren der finiten Differenzen für eine Dimension, wie sie auf Blatt 1 eingeführt wurde. Zur Berechnung einer Lösung können Zeit- und Ortsdiskretisierung separat voneinander behandelt werden. Das heißt, dass für die Zeitschrittweite h_t und einen festen Zeitpunkt $t_k := t_0 + h_t k$ das diskretisierte stationäre Problem gelöst wird und anschließend mit dem nächsten Zeitschritt t_{k+1} verbunden wird. Für die Wärmeleitungsgleichung mit diskretisierten Zeitpunkten ergibt sich

$$L_h u_h(t_*) + \frac{u_h(t_{k+1}) - u_h(t_k)}{h_t} = f_h, \quad (1)$$

wobei $L_h u_h(t_*)$ die Ortsdiskretisierung zum Zeitpunkt t_* ist. Die Randwerte g_h des Gebietes können sich zu jedem Zeitpunkt t_* ändern, also $u_h(t_k) = g_h(t_k)$ auf $\partial\Omega$. Allerdings werden sie hier als konstant angenommen. Die unterschiedlichen Zeitdiskretisierungsverfahren werden durch die Wahl der Zeitpunkte t_* bestimmt. Außerdem entspricht die Gleichung (1) einer gewöhnlichen Differentialgleichung bezüglich der Zeit. Daher werden solche Verfahren angewendet, die dafür geeignet sind.

¹We6 4.002; Tel.: 0228/73-60454; Email: kuske@ins.uni-bonn.de

Explizites Euler-Verfahren

Dieses Verfahren entsteht durch die Verwendung von t_k anstelle von t_* in Gleichung (1). Dadurch lässt sich die Gleichung zu

$$u_h(t_{k+1}) = (I - h_t L_h)u_h(t_k) + h_t f_h \quad (2)$$

umformen. Beim expliziten Euler-Verfahren kann durch Matrix-Vektor-Multiplikation die Lösung des neuen Zeitschrittes bestimmt werden. Daher lässt sich dieses Verfahren ohne Aufwand implementieren und die Berechnung eines Zeitschrittes schnell bewerkstelligen. Der dadurch entstehende Nachteil ist, dass beim Verfahren Instabilitäten bei zu groß gewählter Zeitschrittweite h_t auftreten können.

Eine notwendige, aber nicht hinreichende Bedingung für die Stabilität einer partiellen Differentialgleichung ist die CFL-Bedingung (COURANT-FRIEDRICH-LEWY). Diese Bedingung stellt den Zusammenhang zwischen Orts- und Zeitdiskretisierung dar. Hierbei wird gefordert, dass die Diskretisierung des Orts stets feiner als die der Zeit ist.

Implizites Euler-Verfahren

Dem expliziten Euler-Verfahren steht das implizite Euler-Verfahren gegenüber. Hierbei wird in der Gleichung (1) für $t_* = t_{k+1}$ gesetzt. Wird diese nach $u_h(t_{k+1})$ aufgelöst, ergibt sich

$$u_h(t_{k+1}) = (I + h_t L_h)^{-1}(u_h(t_k) + h_t f_h). \quad (3)$$

In diesem Fall muss ein lineares Gleichungssystem gelöst werden. Dies erfordert einen wesentlichen höheren Aufwand als beim expliziten Euler-Verfahren. Allerdings ergibt sich dadurch ein stabiles Verfahren, bei dem die Zeitschrittweite h_t beliebig gewählt werden kann.

Crank-Nicolson-Verfahren

Beim Crank-Nicolson-Verfahren werden beide vorherigen Verfahren miteinander kombiniert. Das heißt, dass in der Gleichung (1) der Ausdruck $L_h u_h(t_*)$ durch

$$\frac{L_h u_h(t_k) + L_h u_h(t_{k+1})}{2}$$

ersetzt wird. Das Verfahren bestimmt damit das Mittel der beiden Euler-Verfahren, und es ergibt sich

$$u_h(t_{k+1}) = (I + \frac{h_t}{2} L_h)^{-1}((I - \frac{h_t}{2} L_h)u(t_k) + h_t f_h).$$

Dieses Verfahren lässt sich als implizites Verfahren für den Zeitpunkt $\frac{t_k + t_{k+1}}{2}$ interpretieren und ist damit numerisch stabil. Die Methode benötigt lediglich einen etwas höherem Aufwand als das implizite Euler-Verfahren. Dafür kann der Fehler in der Zeitdiskretisierung quadratisch reduziert werden, wohingegen der Fehler bei beiden Euler-Verfahren linear reduziert wird.

Verfahren höherer Ordnung

Die beiden vorgestellten Euler-Verfahren sind Verfahren erster Ordnung. Das Crank-Nicolson-Verfahren ist von zweiter Ordnung. Des Weiteren ist es möglich, Verfahren höherer Ordnung zu konstruieren. Hierbei werden weitere Zeitpunkte t_k berücksichtigt und mit einander gewichtet. Die Gewichte lassen sich ähnlich den NEWTON-Codes-Formeln bestimmen und werden allgemein in den expliziten ADAMS-BASHFORTH-Verfahren beziehungsweise impliziten ADAMS-MOULTON-Verfahren verwendet.

3 Aufgaben

a) Implementiere das explizite Euler-Verfahren für Gleichung (2); verwende dazu:

```
- void EulerExpl(const unsigned int n, double * const u,  
                const double * const f, const double ht,  
                const unsigned int iter),
```

wobei h_t die Zeitschrittweite ist. Der Parameter `iter` gibt die Anzahl der Zeitschritte an. (Hinweis: Die Matrix-Vektor-Multiplikation kann mit Hilfe der Funktionen `scal` und `axpy` realisiert werden.)

b) Implementiere das implizite Euler-Verfahren für Gleichung (3) mit Hilfe der Funktionen:

```
- void SOR(const unsigned int n, double * const x,  
          const double * const b, const double omega,  
          const double ht, const unsigned int maxIt),
```

die das Gleichungssystem $Ax = b \Leftrightarrow (I + h_t L_h)u_h(t_{k+1}) = u_h(t_k) + h_t f_h$ lösen soll, sowie

```
- void EulerImpl(const unsigned int n, double * const u,  
                const double * const f, const double ht,  
                const unsigned int iter),
```

wobei die Parameter denen des expliziten Euler-Verfahrens entsprechen sollen. Das entstehende Gleichungssystem soll mit Hilfe des Mehrgitter-Verfahrens (V-Zyklus) mit einer Genauigkeit von 10^{-4} gelöst werden!

c) Erstelle eine Tabelle für $h_t \in [\frac{1}{100}, 1]$, $n = 257$ und 100 Zeitschritte, in der die benötigte Rechenzeit beider Verfahren verglichen wird!

d) Visualisiere mit Hilfe des impliziten Euler-Verfahrens für $n = 513$ und $h_t = \frac{1}{100}$ die zeitliche Entwicklung der Wärmeleitungsgleichung vom Zeitpunkt t_0 bis t_{100} !