Numerische Methoden für Differentialgleichungen

Lars Grüne
Lehrstuhl für Angewandte Mathematik
Mathematisches Institut
Universität Bayreuth
95440 Bayreuth
lars.gruene@uni-bayreuth.de
num.math.uni-bayreuth.de

Vorlesungsskript
7. Auflage
Wintersemester 2019/2020

Vorwort

Dieses Skript ist im Rahmen einer gleichnamigen Vorlesung entstanden, die ich im Sommersemester 2019/2020 an der Universität Bayreuth gehalten habe. Es ist die siebte Auflage eines Skriptes, das zuerst im Sommersemester 2003 erstellt wurde. Ich möchte mich an dieser Stelle wie stets bei all den Studentinnen und Studenten bedanken, die mir Fehler mitgeteilt und dadurch zur Verbesserung dieser Auflage beigetragen haben. Neben der Verbesserungen von Fehlern wurde gegenüber der sechsten Auflage die Numerik für partielle Differentialgleichungen neu aufgenommen, also die Kapitel 10–15. Hierzu möchte ich mich besonders bei meinem Kollegen Anton Schiela bedanken, dessen Vorlesungsskript Vorlage für einige der neuen Kapitel und daher eine wichtige Quelle beim Erstellen dieses Skripts war.

Darüberhinaus wurde das Skript auf Basis verschiedener Lehrbücher, Skripte und Monographien erstellt. Dabei wurden bei den gewöhnlichen Differentialgleichungen insbesondere die Bücher von Deuflhard und Bornemann [3] sowie von Hairer, Lubich und Wanner [7] verwendet, allerdings wurden sowohl in Aufbau und Notation als auch bei einer Reihe von Beweisen Änderungen vorgenommen. Bei den partiellen Differentialgleichungen wurden neben dem oben genannten Skript insbesondere die Bücher von Braess [2] sowie von Deuflhard und Weiser [4] und die Skripte von Ohlberger [10] und Rannacher [12] verwendet.

Eine elektronische Version dieses Skripts findet sich unter num.math.uni-bayreuth.de → Lars Grüne → Skripte/Lecture Notes.

Bayreuth, Februar 2020 Lars Grüne

Inhaltsverzeichnis

Kapitel 1

Gewöhnliche Differentialgleichungen

Im Rahmen unserer numerischen Betrachtungen werden wir die benötigten theoretischen Resultate dort einführen, wo wir sie verwenden. Bevor wir mit der Numerik beginnen können, benötigen wir aber zumindest ein theoretisches Grundgerüst mit einigen Basisdefinitionen und Resultaten zu den gewöhnlichen Differentialgleichungen, das der nun folgende Abschnitt bereit stellt.

In diesem Abschnitt werden wir die grundlegenden Gleichungen definieren, mit denen wir uns im ersten Teil dieser Vorlesung beschäftigen wollen und einige ihrer Eigenschaften betrachten. Zudem werden wir zwei verschiedene grafische Darstellungsmöglichkeiten für die Lösungen kennen lernen. Für weitergehende Informationen über gewöhnliche Differentialgleichungen können z.B. die einführenden Lehrbücher [1] oder [6] empfohlen werden.

1.1 Definition

Eine gewöhnliche Differentialgleichung setzt die Ableitung einer Funktion $x : \mathbb{R} \to \mathbb{R}^n$ nach ihrem (eindimensionalen) Argument mit der Funktion selbst in Beziehung. Formal beschreibt dies die folgende Definition.

Definition 1.1 Eine *gewöhnliche Differentialgleichung* (DGL) im $\mathbb{R}^n$, $n \in \mathbb{N}$, ist gegeben durch die Gleichung

$$\frac{d}{dt}x(t) = f(t, x(t)), \tag{1.1}$$

wobei $f : D \to \mathbb{R}^n$ eine stetige Funktion ist und *Vektorfeld* genannt wird, deren Definitionsbereich D eine offene Teilmenge von $\mathbb{R} \times \mathbb{R}^n$ ist.

Eine *Lösung* von (1.1) ist eine stetig differenzierbare Funktion $x : \mathbb{R} \to \mathbb{R}^n$, die (1.1) erfüllt. □

Einige Anmerkungen zur Notation bzw. Sprechweise:

- Die unabhängige Variable t werden wir üblicherweise als Zeit interpretieren, obwohl (abhängig vom modellierten Sachverhalt) gelegentlich auch andere Interpretationen möglich sind.
- Statt $\frac{d}{dt}x(t)$ schreiben wir oft kurz $\dot{x}(t)$.
- Die Lösungsfunktion $x(t)$ nennen wir auch *Lösungskurve* oder *(Lösungs-)Trajektorie*.
- Falls das Vektorfeld f nicht von t abhängt, also $\dot{x}(t) = f(x(t))$ ist, nennen wir die Differentialgleichung *autonom*.

1.2 Anfangswertprobleme

Eine gewöhnliche Differentialgleichung besitzt im Allgemeinen unendlich viele Lösungen. Als Beispiel betrachte die (sehr einfache) eindimensionale DGL mit $f(x,t) = x$, also

$$\dot{x}(t) = x(t)$$

mit $x(t) \in \mathbb{R}$. Betrachte die Funktion $x(t) = Ce^t$ mit beliebigem $C \in \mathbb{R}$. Dann gilt

$$\dot{x}(t) = \frac{d}{dt}Ce^t = Ce^t = x(t).$$

Für jedes feste C löst Ce^t die obige DGL, es gibt also unendlich viele Lösungen.

Um *eindeutige* Lösungen zu erhalten, müssen wir eine weitere Bedingung festlegen. Dies geschieht in der folgenden Definition.

Definition 1.2 Ein *Anfangswertproblem* für die gewöhnliche Differentialgleichung (1.1) besteht darin, zu gegebenem $t_0 \in \mathbb{R}$ und $x_0 \in \mathbb{R}^n$ eine Lösungsfunktion $x(t)$ zu finden, die (1.1) erfüllt und für die darüberhinaus die Gleichung

$$x(t_0) = x_0 \tag{1.2}$$

gilt. □

Notation und Sprechweisen:

- Für die Lösung $x(t)$, die (1.1) und (1.2) erfüllt, schreiben wir $x(t; t_0, x_0)$. Im Spezialfall $t_0 = 0$ werden wir oft kurz $x(t; x_0)$ schreiben.
- Die Zeit $t_0 \in \mathbb{R}$ bezeichnen wir als *Anfangszeit*, den Wert $x_0 \in \mathbb{R}^n$ als *Anfangswert*. Das Paar (t_0, x_0) bezeichnen wir als *Anfangsbedingung*, ebenso nennen wir die Gleichung (1.2) *Anfangsbedingung*.

Bemerkung 1.3 Eine stetig differenzierbare Funktion $x : I \to \mathbb{R}^n$ löst das Anfangswertproblem (1.1), (1.2) für ein $t_0 \in I$ und ein $x_0 \in \mathbb{R}^n$ genau dann, wenn sie für alle $t \in I$ die *Integralgleichung*

$$x(t) = x_0 + \int_{t_0}^{t} f(\tau, x(\tau))d\tau \tag{1.3}$$

erfüllt. Dies folgt sofort durch Integrieren von (1.1) bzgl. t bzw. durch Differenzieren von (1.3) nach t unter Verwendung des Hauptsatzes der Differential- und Integralrechnung. Beachte dabei, dass eine stetige Funktion x, die (1.3) erfüllt, „automatisch“ stetig differenzierbar ist, da aus der Stetigkeit von x sofort die stetige Differenzierbarkeit der rechten Seite in (1.3) und damit wegen der Gleichheit auch für x selbst folgt. □

1.3 Ein Existenz- und Eindeutigkeitssatz

Unter geeigneten Bedingungen an f können wir einen Existenz- und Eindeutigkeitssatz für Anfangswertprobleme der Form (1.1), (1.2) erhalten.

Satz 1.4 Betrachte die gewöhnliche Differentialgleichung (1.1) für ein $f : D \to \mathbb{R}^n$ mit $D \subseteq \mathbb{R} \times \mathbb{R}^n$ offen. Das Vektorfeld f sei stetig, darüberhinaus sei f Lipschitz-stetig im zweiten Argument im folgenden Sinne: Für jede kompakte Teilmenge $K \subset D$ existiere eine Konstante $L > 0$, so dass die Ungleichung

$$\|f(t,x) - f(t,y)\| \leq L\|x - y\|$$

gilt für alle $t \in \mathbb{R}$ und $x, y \in \mathbb{R}^n$ mit (t,x), $(t,y) \in K$.

Dann gibt es für jede Anfangsbedingung $(t_0, x_0) \in D$ genau eine Lösung $x(t; t_0, x_0)$ des Anfangswertproblems (1.1), (1.2). Diese ist definiert für alle t aus einem offenen *maximalen Existenzintervall* $I_{t_0,x_0} \subseteq \mathbb{R}$ mit $t_0 \in I_{t_0,x_0}$.

Beweis: Teil 1: Wir zeigen zunächst, dass es für jede Anfangsbedingung $(t_0, x_0) \in D$ ein abgeschlossenes Intervall J um t_0 gibt, auf dem die Lösung existiert und eindeutig ist.

Dazu wählen wir ein beschränktes abgeschlossenes Intervall I um t_0 und ein $\varepsilon > 0$, so dass die kompakte Umgebung $U = I \times \overline{B}_\varepsilon(x_0)$ von (t_0, x_0) in D liegt (dies ist möglich, da D eine offene Menge ist). Da f stetig ist und U kompakt ist, existiert eine Konstante M, so dass $\|f(t,x)\| \leq M$ für alle $(t,x) \in U$ gilt. Wir wählen nun $J = [t_0 - \delta, t_0 + \delta]$ wobei $\delta > 0$ so gewählt ist, dass $J \subseteq I$ gilt und $L\delta < 1$ sowie $M\delta < \varepsilon$ erfüllt ist, wobei L die Lipschitz-Konstante von f für $K = U$ ist. Alle somit konstruierten Mengen sind in Abbildung 1.1 dargestellt.

Nun verwenden wir zum Beweis der Existenz und Eindeutigkeit der Lösung auf J den Banachschen Fixpunktsatz auf dem Banachraum $C(J, \mathbb{R}^n)$ mit der Norm

$$\|x\|_\infty := \sup_{t \in J} \|x(t)\|.$$

Auf $C(J, \mathbb{R}^d)$ definieren wir die Abbildung

$$T : C(J, \mathbb{R}^d) \to C(J, \mathbb{R}^d), \quad T(x)(t) := x_0 + \int_{t_0}^{t} f(\tau, x(\tau)) d\tau.$$

Beachte, dass für jedes $t \in J$ und jedes $x \in B := C(J, \overline{B}_\varepsilon(x_0))$ die Ungleichung

$$\begin{aligned} \|T(x)(t) - x_0\| &= \left\| \int_{t_0}^{t} f(\tau, x(\tau)) d\tau \right\| \leq \left| \int_{t_0}^{t} \underbrace{\|f(\tau, x(\tau))\|}_{\leq M, \text{ weil } (\tau, x(\tau)) \in \overline{U}} d\tau \right| \\ &\leq \delta M \leq \varepsilon \end{aligned}$$

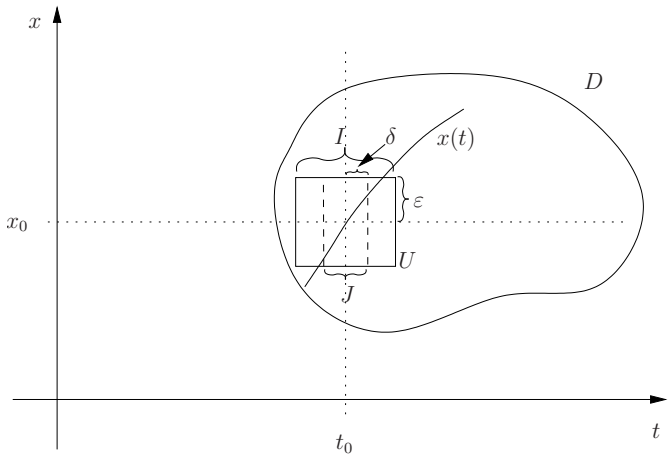

Abbildung 1.1: Mengen im Beweis von Teil 1

gilt, weswegen T die Menge B in sich selbst abbildet.

Um den Banachschen Fixpunktsatz auf dieser Menge anzuwenden, müssen wir zeigen, dass $T : B \to B$ eine Kontraktion ist, also dass

$$\|T(x) - T(y)\|_\infty \leq k\|x - y\|_\infty$$

gilt für alle $x, y \in B$ und ein $k < 1$. Diese Eigenschaft folgt für $k = L\delta < 1$ aus

$$\begin{aligned}
\|T(x) - T(y)\|_\infty &= \sup_{t\in J} \left\| \int_{t_0}^{t} f(\tau, x(\tau))d\tau - \int_{t_0}^{t} f(\tau, y(\tau))d\tau \right\| \\
&\leq \sup_{t\in J} \left| \int_{t_0}^{t} \underbrace{\|f(\tau, x(\tau)) - f(\tau, y(\tau))\|}_{\leq L\|x(\tau)-y(\tau)\| \leq L\|x-y\|_\infty} d\tau \right| \\
&\leq \sup_{t\in J} |t - t_0| L\|x - y\|_\infty = \delta L\|x - y\|_\infty.
\end{aligned}$$

Also sind die Voraussetzungen des Banachschen Fixpunktsatzes erfüllt, weswegen T einen eindeutigen Fixpunkt $x \in B$, also eine „Fixpunktfunktion“, besitzt. Da diese Fixpunktfunktion x nach Konstruktion von T die Integralgleichung (1.3) erfüllt, ist sie nach Bemerkung 1.3 eine stetig differenzierbare Lösung des Anfangswertproblems.

Es bleibt zu zeigen, dass diese eindeutig ist, dass also kein weiterer Fixpunkt $y \in C(J, \mathbb{R}^d)$ existiert. Aus dem Banachschen Fixpunktsatz folgt bereits, dass in $B = C(J, \overline{B}_\varepsilon(x_0))$ kein weiterer Fixpunkt von T liegt. Zum Beweis der Eindeutigkeit reicht es also zu zeigen, dass außerhalb von B kein Fixpunkt y liegen kann. Wir beweisen dies per Widerspruch: Angenommen, es existiert eine Fixpunktfunktion $y \notin B$ von T, d.h. es gilt $\|y(t) - x_0\| > \varepsilon$ für ein $t \in J$, für das wir o.B.d.A. $t > t_0$ annehmen. Dann existiert aus Stetigkeitsgründen ein $t^* \in J$ mit $\|y(t^*) - x_0\| = \varepsilon$ und $y(s) \in \overline{B}_\varepsilon(x_0)$ für $s \in [t_0, t^*]$. Damit folgt

$$\varepsilon = \|y(t^*) - x_0\| = \left\| \int_{t_0}^{t^*} f(s, y(s))ds \right\| \leq \int_{t_0}^{t^*} \|f(s, y(s))\|ds$$

$$\leq \quad (t^* - t_0)M \;<\; \delta M,$$

was wegen $\delta M \leq \varepsilon$ ein Widerspruch ist. Daher liegt jeder mögliche Fixpunkt $y \in C(J, \mathbb{R}^d)$ von T bereits in B, womit die Eindeutigkeit folgt.

Zusammenfassend liefert uns Teil 1 des Beweises also, dass *lokal* - also auf einem kleinen Intervall J um t_0 - eine eindeutige Lösung $x(t) = x(t; t_0, x_0)$ existiert. Dies ist die Aussage des *Satzes von Picard-Lindelöf*[1], der in vielen Büchern als eigenständiger Satz formuliert ist.

Teil 2: Wir zeigen als nächstes die Eindeutigkeit der Lösung auf beliebig großen Intervallen I. Seien dazu x und y zwei auf einem Intervall I definierte Lösungen des Anfangswertproblems. Wir beweisen $x(t) = y(t)$ für alle $t \in I$ per Widerspruch und nehmen dazu an, dass ein $t \in I$ existiert, in dem die beiden Lösungen nicht übereinstimmen, also $x(t) \neq y(t)$. O.b.d.A. sei $t > t_0$. Da beide Lösungen nach Teil 1 auf J übereinstimmen und stetig sind, existieren $t_2 > t_1 > t_0$, so dass

$$x(t_1) = y(t_1) \quad \text{und} \quad x(t) \neq y(t) \text{ für alle } t \in (t_1, t_2) \tag{1.4}$$

gilt. Offenbar lösen beide Funktionen das Anfangswertproblem mit Anfangsbedingung $(t_1, x(t_1)) \in D$. Aus Teil 1 des Beweises folgt die Eindeutigkeit der Lösungen dieses Problems auf einem Intervall $\widetilde{J}$ um t_1, also

$$x(t) = y(t) \text{ für alle } t \in \widetilde{J}.$$

Da $\widetilde{J}$ als Intervall um t_1 einen Punkt t mit $t_1 < t < t_2$ enthält, widerspricht dies (1.4), weswegen x und y für alle $t \in I$ übereinstimmen müssen.

Teil 3: Schließlich zeigen wir die Existenz des maximalen Existenzintervalls. Für J aus Teil 1 definieren wir dazu

$$t^+ := \sup\{s > t_0 \,|\, \text{es existiert eine Lösung auf } J \cup [t_0, s)\}$$

sowie

$$t^- := \inf\{s < t_0 \,|\, \text{es existiert eine Lösung auf } J \cup (s, t_0]\}$$

und setzen $I_{t_0,x_0} = (t^-, t^+)$. Sowohl t^- als auch t^+ existieren, da die Mengen, über die das Supremum bzw. Infimum genommen wird, nichtleer sind, da sie alle $s \in J$ enthalten. Per Definition von t^+ bzw. t^- kann es keine Lösung auf einem größeren Intervall $I \supset I_{t_0,x_0}$ geben, also ist dies das maximale Existenzintervall.

□

Am Rand des maximalen Existenzintervalls $I_{t_0,x_0} = (t^-, t^+)$ hört die Lösung auf zu existieren. Ist das Intervall in einer Zeitrichtung beschränkt, so kann dies nur zwei verschiedene Ursachen haben: Entweder die Lösung divergiert, oder sie konvergiert gegen einen Randpunkt von D. Formal ausgedrückt:

Falls $t^+ < \infty$ ist und die Lösung $x(t; t_0, x_0)$ für $t \nearrow t^+$ gegen ein $x^+ \in \mathbb{R}^d$ konvergiert, so muss $(t^+, x^+) \notin D$ gelten. Analog gilt die Aussage für $t \searrow t^-$. Hierbei steht $t \nearrow t^+$ kurz für $t \to t^+$ und $t < t^+$ und $t \searrow t^-$ für $t \to t^-$ und $t > t^-$.

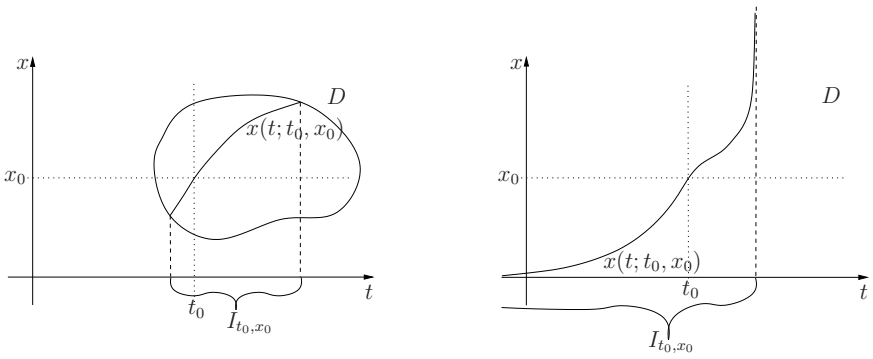

Abbildung 1.2: Lösungsverhalten am Rand des Existenzintervalls für eine beschränkte (links) und eine unbeschränkte Definitionsmenge D (rechts)

Anschaulich sind die zwei Möglichkeiten in Abbildung 1.2 dargestellt.

Die Begründung für dieses Verhalten ist wie folgt:

Wenn $x(t;t_0,x_0)$ für $t \nearrow t^+$, gegen $x^+ \in \mathbb{R}^d$ mit $(t^+,x^+) \in D$ konvergiert, so existiert eine Lösung $x(t;t^+,x^+)$ auf einem offenen Intervall I_{t^+,x^+} um t^+. Dann ist die zusammengesetzte Lösung

$$y(t) = \begin{cases} x(t;t_0,x_0), & t \in I_{t_0,x_0} \\ x(t;t^+,x^+), & t \in I_{t^+,x^+} \setminus I_{t_0,x_0} \end{cases}$$

stetig und erfüllt für alle $t \in I_{t_0,x_0} \cup I_{t^+,x^+}$ die Integralgleichung (1.3), damit nach Bemerkung 1.3 auch das Anfangswertproblem und ist folglich eine Lösung, die über t^+ hinaus definiert ist: ein Widerspruch zur Definition von t^+.

Im Fall $D = \mathbb{R} \times \mathbb{R}^d$ gilt daher für $t^+ < \infty$ bzw. $t^- > -\infty$ insbesondere, dass die Lösung $x(t;t_0,x_0)$ für $t \nearrow t^+$ bzw. $t \searrow t^-$ divergieren muss, da eine Konvergenz gegen $(t^+,x^+) \notin D$ bzw. $(t^-,x^-) \notin D$ nicht möglich ist. Beachte, dass dieser Fall tatsächlich auftreten kann: eine unbeschränkte Definitionsmenge D von f bedeutet nicht, dass auch die Lösungen auf einem unbeschränkten Intervall $I_{t_0,x_0} = \mathbb{R}$ existieren. Ein Beispiel dafür ist die Differentialgleichung $\dot{x}(t) = x(t)^2$ mit $x(t) \in \mathbb{R}$. Diese besitzt für Anfangsbedingung $x(0) = 1$ die Lösung $x(t) = 1/(1-t)$, die für $t \to 1$ gegen unendlich strebt. Es gilt also $t^+ = 1$, obwohl $D = \mathbb{R} \times \mathbb{R}$ unbeschränkt ist.

Zu beachten ist weiterhin, dass die Divergenz nicht wie im rechten Bild in Abbildung 1.2 skizziert bedeuten muss, dass die Lösung gegen unendlich (oder minus unendlich) strebt. Ein Beispiel dafür ist $\dot{x}(t) = -\cos(1/t)/t^2$ mit $D = \mathbb{R} \setminus \{0\} \times \mathbb{R}$. Für die Anfangsbedingung $x(-1) = \sin(-1)$ erhält man hier die Lösung $x(t) = \sin(1/t)$. Für $t \to t^+ = 0$, $t < 0$ konvergiert diese Lösung nicht, weil sie immer schneller zwischen -1 und -1 oszilliert; sie ist aber für alle $t < 0$ nach oben und unten beschränkt.

Wir werden im Folgenden immer annehmen, dass die Annahmen von Satz 1.4 erfüllt sind, auch ohne dies explizit zu erwähnen. Auch werden wir oft Mengen der Form $[t_1,t_2] \times K$

[1] Charles Picard, französischer Mathematiker, 1856–1941
Ernst Lindelöf, finnischer Mathematiker, 1870–1946

mit $K \subset \mathbb{R}^n$ betrachten, bei denen wir — ebenfalls ohne dies immer explizit zu erwähnen — annehmen, dass alle Lösungen $x(t; t_0, x_0)$ mit $x_0 \in K$ für alle $t_0, t \in [t_1, t_2]$ existieren.

Eine einfache Konsequenz aus Satz 1.4 ist die sogenannte *Kozykluseigenschaft* der Lösungen, die für $(t_0, x_0) \in D$ und zwei Zeiten $t_1, t \in \mathbb{R}$ gegeben ist durch

$$x(t; t_0, x_0) = x(t; t_1, x(t_1; t_0, x_0)), \tag{1.5}$$

vorausgesetzt natürlich, dass alle hier auftretenden Lösungen zu den angegebenen Zeiten auch existieren. Zum Beweis rechnet man nach, dass der linke Ausdruck in (1.5) das Anfangswertproblem (1.1), (1.2) zur Anfangsbedingung $(t_1, x(t_1; t_0, x_0))$ löst. Da der rechte dies ebenfalls tut, müssen beide übereinstimmen.

Unter den Voraussetzungen von Satz 1.4 ist die Lösungsabbildung $x(t; t_0, x_0)$ zudem stetig in all ihren Variablen, also in t, t_0 und x_0.

1.4 Grafische Darstellung der Lösungen

Zur grafischen Darstellung von Lösungen verwenden wir zwei verschiedene Methoden, die wir hier an der zweidimensionalen DGL

$$\dot{x}(t) = \begin{pmatrix} -0.1 & 1 \\ -1 & -0.1 \end{pmatrix} x(t)$$

mit $x(t) = (x_1(t), x_2(t))^T$ und Anfangsbedingung $x(0) = (1, 1)^T$ illustrieren wollen. Da jede Lösung einer Differentialgleichung eine Funktion von $\mathbb{R}$ nach $\mathbb{R}^n$ darstellt, kann man die Graphen der einzelnen Komponenten $x_i(t)$ der Lösung in Abhängigkeit von t darstellen. Für die obige DGL ist dies in Abbildung 1.3 dargestellt. Die durchgezogene Linie zeigt $x_1(t)$ während die gestrichelte Linie $x_2(t)$ darstellt.

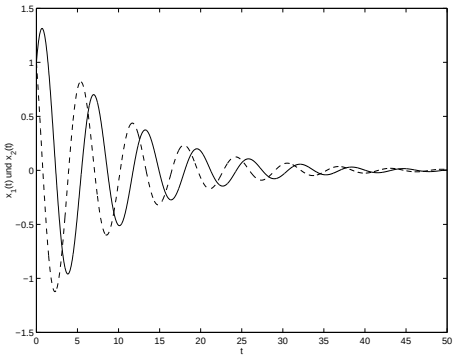

Abbildung 1.3: Darstellung von $x(t)$ mittels Graphen ($x_1(t)$ durchgezogen, $x_2(t)$ gestrichelt)

Eine alternative Darstellung, die speziell für zwei- und dreidimensionale Differentialgleichungen geeignet ist, ergibt sich, wenn man statt der Funktionsgraphen der Komponenten x_i die Kurve $\{x(t) \,|\, t \in [0, T]\} \subset \mathbb{R}^n$ darstellt. Hier geht in der Grafik die Information über die Zeit (sowohl über die Anfangszeit t_0 als auch über die laufende Zeit t) verloren.

Letzteres kann zumindest teilweise durch das Anbringen von Pfeilen, die die Zeitrichtung symbolisieren, ausgeglichen werden. Ein Beispiel für diese Darstellung zeigt Abbildung 1.4.

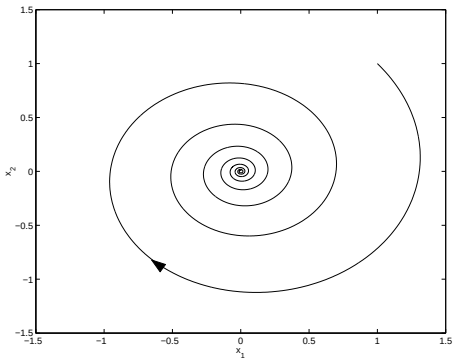

Abbildung 1.4: Darstellung von $x(t)$ als Kurve

Am Computer kann man die Darstellung als Kurve mit einer Animation verbinden, so dass man die Information über den zeitlichen Ablauf der Lösung über die Animation wieder zurück erhält.

Für autonome Differentialgleichungen ist der Verlust der Anfangszeit in der Grafik nicht weiter schlimm, da die Lösungen nicht wirklich von der Anfangszeit abhängen: man rechnet leicht nach, dass hier für die Anfangszeiten t_0 und $t_0 + t_1$ die Beziehung

$$x(t; t_0 + t_1, x_0) = x(t - t_1; t_0, x_0) \tag{1.6}$$

gilt. Die Lösung verschiebt sich also auf der t-Achse, verändert sich aber ansonsten nicht. Insbesondere ist die in Abbildung 1.4 dargestellte Kurve für autonome DGL für alle Anfangszeiten gleich.

Kapitel 2

Allgemeine Theorie der Einschrittverfahren

In diesem Kapitel werden wir eine wichtige Klasse von Verfahren zur Lösung gewöhnlicher Differentialgleichungen einführen und analysieren, die *Einschrittverfahren*.

2.1 Diskrete Approximationen

In der Numerik gewöhnlicher Differentialgleichungen wollen wir eine Approximation an die Lösungsfunktion $x(t; t_0, x_0)$ für $t \in [t_0, T]$ berechnen (wir nehmen hier immer an, dass die Lösungen auf den angegebenen Intervallen existieren). In der folgenden Definition definieren wir die Art von Approximationen, die wir betrachten wollen und einen Begriff der Konvergenzordnung.

Definition 2.1 (i) Eine Menge $\mathcal{T} = \{t_0, t_1, \ldots, t_N\}$ von Zeiten mit $t_0 < t_1 < \ldots < t_N = T$ heißt *Gitter* auf dem Intervall $[t_0, T]$. Die Werte

$$h_i = t_{i+1} - t_i$$

heißen *Schrittweiten*, der Wert

$$\bar{h} = \max_{i=0,\ldots,N-1} h_i$$

heißt *maximale Schrittweite*. Im Fall *äquidistanter Schrittweiten* $h_0 = h_1 = \ldots = h_{N-1}$ schreiben wir zumeist h statt h_i.

(ii) Eine Funktion $\tilde{x} : \mathcal{T} \to \mathbb{R}^n$ heißt *Gitterfunktion*.

(iii) Es seien $\tilde{x}_\mathcal{T}$ Gitterfunktionen zu Gittern $\mathcal{T}$ auf dem Intervall $[t_0, T] \subset I_{t_0,x_0}$ mit maximalen Schrittweiten $\bar{h}_\mathcal{T}$. Die Gitterfunktionen $\tilde{x}_\mathcal{T}$ bilden eine *(diskrete) Approximation* der Lösung $x(t; t_0, x_0)$ von (1.1), falls für jede kompakte Menge $K \subset D$ mit $[t_0, T] \subset I_{t_0,x_0}$ für alle $(t_0, x_0) \in K$ eine Funktion $\rho(h)$ mit $\rho(h) \to 0$ für $h \to 0$ existiert mit

$$\max_{t_i \in \mathcal{T}} \|\tilde{x}_\mathcal{T}(t_i) - x(t_i; t_0, x_0)\| \leq \rho(\bar{h}_\mathcal{T}).$$

Die diskrete Approximation hat die *Konvergenzordnung* $p > 0$, falls für jede kompakte Menge $K \subset D$ und alle $T > 0$ mit $[t_0, T] \subset I_{t_0,x_0}$ für alle $(t_0, x_0) \in K$ ein $C > 0$ existiert, so dass

$$\rho(h) = Ch^p$$

gewählt werden kann. In diesem Fall schreiben wir kurz $\tilde{x}(t_i; t_0, x_0) = x(t_i; t_0, x_0) + O(\bar{h}^p)$. □

Bemerkung 2.2 Wir haben in der Einführung in die Numerik verschiedene Methoden kennen gelernt, mit denen man Funktionen numerisch darstellen kann, z.B. Polynom- oder Splineinterpolation. Jede Gitterfunktion gemäß Definition 2.1 kann natürlich mit diesen Methoden zu einer "echten" Funktion erweitert werden. □

Ein Einschrittverfahren ist nun gegeben durch eine numerisch auswertbare Funktion Φ, mittels derer wir eine Gitterfunktion zu einem gegebenen Gitter berechnen können. Formal ist dies wie folgt definiert.

Definition 2.3 Ein *Einschrittverfahren* ist gegeben durch eine stetige Abbildung

$$\Phi : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n,$$

mit der zu jedem Gitter $\mathcal{T}$ und jedem Anfangswert x_0 mittels

$$\tilde{x}(t_0) = x_0, \;\; \tilde{x}(t_{i+1}) = \Phi(t_i, \tilde{x}(t_i), h_i) \text{ für } i = 0, 1, \ldots, N-1$$

rekursiv eine Gitterfunktion definiert werden kann.

Wenn die so erzeugten Gitterfunktionen die Bedingung aus Definition 2.1 (iii) erfüllen, so nennen wir das Einschrittverfahren *konvergent* bzw. *konvergent mit Konvergenzordnung p*. □

Der Name *Einschrittverfahren* ergibt sich dabei aus der Tatsache, dass der Wert $\tilde{x}(t_{i+1})$ nur aus dem direkten Vorgängerwert $\tilde{x}(t_i)$ berechnet wird. Wir werden später auch *Mehrschrittverfahren* kennen lernen, bei denen $\tilde{x}(t_{i+1})$ aus $\tilde{x}(t_{i-k}), \tilde{x}(t_{i-k+1}), \ldots, \tilde{x}(t_i)$ berechnet wird.

2.2 Erste einfache Einschrittverfahren

Bevor wir in die Konvergenztheorie einsteigen und mathematisch untersuchen, welche Bedingungen Φ erfüllen muss, damit die erzeugte Gitterfunktion eine Approximation darstellt, wollen wir in diesem Abschnitt zwei Einschrittverfahren heuristisch betrachten.

Die Idee der Verfahren erschließt sich am einfachsten über die Integralgleichung (1.3). Die exakte Lösung erfüllt ja gerade

$$x(t_{i+1}) = x(t_i) + \int_{t_i}^{t_{i+1}} f(\tau, x(\tau)) d\tau.$$

Die Idee ist nun, das Integral durch einen Ausdruck zu ersetzen, der numerisch berechenbar ist, wenn wir $x(\tau)$ für $\tau > t_i$ nicht kennen. Die einfachste Approximation ist die Rechteck-Regel (oder Newton-Cotes Formel mit $n = 0$, die wir in der Einführung in die Numerik wegen ihrer Einfachheit gar nicht betrachtet haben)

$$\int_{t_i}^{t_{i+1}} f(\tau, x(\tau))d\tau \approx (t_{i+1} - t_i)f(t_i, x(t_i)) = h_i f(t_i, x(t_i)). \tag{2.1}$$

Setzen wir also

$$\Phi(t, x, h) = x + hf(t, x), \tag{2.2}$$

so gilt

$$\tilde{x}(t_{i+1}) = \Phi(t_i, \tilde{x}(t_i), h_i) = \tilde{x}(t_i) + h_i f(t_i, \tilde{x}(t_i))$$

und wenn wir $\tilde{x}(t_i) \approx x(t_i)$ annehmen, so können wir fortfahren

$$\ldots \approx x(t_i) + h_i f(t_i, x(t_i)) \approx x(t_i) + \int_{t_i}^{t_{i+1}} f(\tau, x(\tau))d\tau.$$

Da $\tilde{x}(t_0) = x_0 = x(t_0)$ ist, kann man damit rekursiv zeigen, dass $\tilde{x}(t_{i+1})$ eine Approximation von $x(t_{i+1})$ ist. Wir werden dies im nächsten Abschnitt mathematisch präzisieren.

Das durch (2.2) gegebene Verfahren ist das einfachste Einschrittverfahren und heißt *Euler'sche Polygonzugmethode* oder einfach *Euler-Verfahren*. Es hat eine einfache geometrische Interpretation: In jedem Punkt $\tilde{x}(t_i)$ berechnen wir die Steigung der exakten Lösung durch diesen Punkt (das ist gerade $f(t_i, \tilde{x}(t_i))$) und folgen der dadurch definierten Geraden bis zum nächsten Zeitschritt. Das Prinzip ist in Abbildung 2.1 grafisch dargestellt.

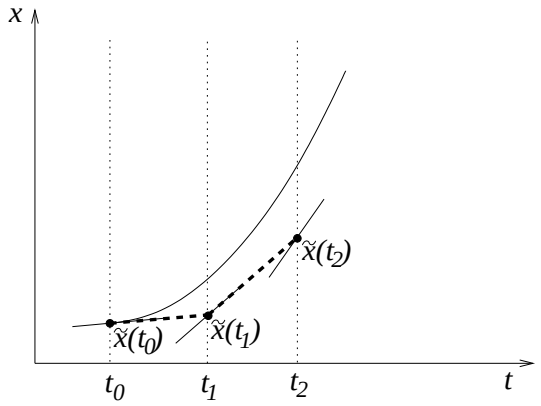

Abbildung 2.1: Grafische Veranschaulichung des Euler-Verfahrens

Das Euler-Verfahren liefert nur eine recht grobe Approximation der Lösung. Bessere Verfahren kann man erhalten, wenn man statt (2.1) eine genauere Approximation verwendet. Eine bessere Möglichkeit ist z.B.

$$\int_{t_i}^{t_{i+1}} f(\tau, x(\tau))d\tau \approx \frac{h_i}{2}\left(f(t_i, x(t_i)) + f\Big(t_{i+1}, x(t_i) + h_i f(t_i, x(t_i))\Big)\right). \tag{2.3}$$

Dies ist nichts anderes als die Trapez-Regel (oder Newton-Cotes Formel mit $n = 1$), bei der wir den unbekannten Wert $x(t_{i+1})$ durch die Euler-Approximation $x(t_{i+1}) \approx x(t_i) + h_i f(t_i, x(t_i))$ ersetzen. Das daraus resultierende Verfahren ist gegeben durch

$$\Phi(t, x, h) = x + \frac{h}{2}\left(f(t, x) + f\Big(t + h, x + hf(t, x)\Big)\right)$$

und heißt *Heun-Verfahren*. Es ist tatsächlich schon deutlich besser als das Euler-Verfahren.

Man kann sich leicht vorstellen, dass weitere bessere Verfahren sehr komplizierte Formeln benötigen. Wir werden deshalb später einen Formalismus kennen lernen, mit dem man auch sehr komplizierte Verfahren einfach aufschreiben und implementieren kann.

Ein Grundalgorithmus zur Approximation einer Lösung $x(t; t_0, x_0)$ auf $[t_0, T]$ mittels eines Einschrittverfahrens Φ lässt sich nun leicht angeben. Wir beschränken uns hierbei zunächst auf Gitter mit konstanter Schrittweite, also $h_i = h$ für alle $i = 0, 1, 2, \ldots, N$, wobei wir N als Parameter vorgeben.

Algorithmus 2.4 (Lösung eines Anfangswertproblems mit Einschrittverfahren)

Eingabe: Anfangsbedingung (t_0, x_0), Endzeit T, Schrittzahl N, Einschrittverfahren Φ

(1) Setze $h := (T - t_0)/N$, $\tilde{x}_0 = x_0$

(2) Berechne $t_{i+1} = t_i + h$, $\tilde{x}_{i+1} := \Phi(t_i, \tilde{x}_i, h)$ für $i = 0, \ldots, N - 1$.

Ausgabe: Werte der Gitterfunktion $\tilde{x}(t_i) = \tilde{x}_i$ in $t_0, \ldots, t_N$ □

2.3 Konvergenztheorie

Die Grundidee der Konvergenztheorie für numerische Methoden für Differentialgleichungen liegt in einem geschickten Trick, mit dem verschiedene Fehlerquellen separiert werden können. Wir schreiben hier kurz $x(t) = x(t; t_0, x_0)$. Um nun den Fehler

$$\|\tilde{x}(t_i) - x(t_i)\| = \|\Phi(t_{i-1}, \tilde{x}(t_{i-1}), h_{i-1}) - x(t_i)\|$$

abzuschätzen, schieben wir mittels der Dreiecksungleichung die Hilfsgröße

$$\Phi(t_{i-1}, x(t_{i-1}), h_{i-1})$$

ein. Wir erhalten so mit (1.5) die Abschätzung

$$\begin{aligned}
\|\tilde{x}(t_i) - x(t_i)\| \;\leq\; & \|\Phi(t_{i-1}, \tilde{x}(t_{i-1}), h_{i-1}) - \Phi(t_{i-1}, x(t_{i-1}), h_{i-1})\| \\
& + \|\Phi(t_{i-1}, x(t_{i-1}), h_{i-1}) - x(t_i)\| \\
= \; & \|\Phi(t_{i-1}, \tilde{x}(t_{i-1}), h_{i-1}) - \Phi(t_{i-1}, x(t_{i-1}), h_{i-1})\| \\
& + \|\Phi(t_{i-1}, x(t_{i-1}), h_{i-1}) - x(t_i; t_{i-1}, x_{i-1})\|
\end{aligned}$$

Statt also direkt den Fehler zur Zeit t_i abzuschätzen, betrachten wir getrennt die zwei Terme

(a) $\|\Phi(t_{i-1}, \tilde{x}(t_{i-1}), h_{i-1}) - \Phi(t_{i-1}, x(h_{i-1}), h_{i-1})\|$, also die Auswirkung des Fehlers bis zur Zeit t_{i-1} in Φ

(b) $\|\Phi(t_{i-1}, x(t_{i-1}), h_{i-1}) - x(t_i; t_{i-1}, x_{i-1})\|$, also den lokalen Fehler beim Schritt von $x(t_{i-1})$ nach $x(t_i)$

Die folgende Definition gibt die benötigten Eigenschaften an Φ an, mit denen diese Fehler abgeschätzt werden können.

Definition 2.5 (i) Ein Einschrittverfahren erfüllt die *Lipschitzbedingung* (oder *Stabilitätsbedingung*), falls für jede kompakte Menge $K \subset D$ des Definitionsbereiches der Differentialgleichung ein $L > 0$ existiert, so dass für alle Paare $(t_0, x_1), (t_0, x_2) \in K$ und alle hinreichend kleinen $h > 0$ die Abschätzung

$$\|\Phi(t_0, x_1, h) - \Phi(t_0, x_2, h)\| \leq (1 + Lh)\|x_1 - x_2\| \tag{2.4}$$

gilt.

(ii) Ein Einschrittverfahren Φ heißt *konsistent*, falls für jede kompakte Menge $K \subset D$ des Definitionsbereiches der Differentialgleichung eine Funktion $\varepsilon(h)$ mit $\lim_{h\to 0} \varepsilon(h) = 0$ existiert, so dass für alle $(t_0, x_0) \in K$ und alle hinreichend kleinen $h > 0$ die Ungleichung

$$\|\Phi(t_0, x_0, h) - x(t_0 + h; t_0, x_0)\| \leq h\varepsilon(h) \tag{2.5}$$

gilt. O.B.d.A. nehmen wir dabei an, dass $\varepsilon(h)$ monoton ist, ansonsten können wir $\varepsilon(h)$ durch $\sup_{h\in[0,h]} \varepsilon(h)$ ersetzen.
Das Verfahren hat die *Konsistenzordnung* $p > 0$, falls für jede kompakte Menge $K \subset D$ ein $E > 0$ existiert, so dass $\varepsilon(h) = Eh^p$ gewählt werden kann. In diesem Fall schreiben wir auch $\Phi(t_0, x_0, h) = x(t_0 + h; t_0, x_0) + O(h^{p+1})$. □

Offenbar garantiert (2.4), dass der Fehlerterm (a) nicht zu groß wird, während (2.5) dazu dient, den Term (b) abzuschätzen. Der formale Beweis folgt in Satz 2.7. Bevor wir diesen formulieren, wollen wir uns noch überlegen, ob die im vorherigen Abschnitt definierten Verfahren diese Bedingungen erfüllen.

Man rechnet leicht nach, dass das Euler- und das Heun-Verfahren die Lipschitzbedingung erfüllen. Die Konsistenzbedingung (2.5) ist allerdings nicht so leicht nachzuprüfen, da sie mit Hilfe der (unbekannten) Lösungen $x(t; t_0, x_0)$ formuliert ist. Das folgende Lemma stellt eine alternative und leichter nachprüfbare Formulierung der Bedingung vor.

Lemma 2.6 Gegeben sei ein Einschrittverfahren Φ der Form

$$\Phi(t, x, h) = x + h\varphi(t, x, h)$$

mit einer stetigen Funktion $\varphi : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$. Dann ist das Verfahren genau dann konsistent, falls für alle $(t, x) \in D$ die Bedingung

$$\varphi(t, x, 0) = f(t, x) \tag{2.6}$$

gilt.

Beweis: Wir schreiben wieder kurz $x(t) = x(t; t_0, x_0)$. Es gilt

$$\begin{aligned}
&\frac{\Phi(t_0, x_0, h) - x(t_0 + h)}{h} \\
&= \frac{1}{h}\left(\Phi(t_0, x_0, h) - x_0 - \int_{t_0}^{t_0+h} f(\tau, x(\tau))d\tau\right) \\
&= \frac{1}{h}\left(\Phi(t_0, x_0, h) - x_0 - \int_{t_0}^{t_0+h} f(t_0, x_0)d\tau + \int_{t_0}^{t_0+h} f(t_0, x_0)d\tau - \int_{t_0}^{t_0+h} f(\tau, x(\tau))d\tau\right) \\
&= \frac{1}{h}\left(h\varphi(t_0, x_0, h) - \int_{t_0}^{t_0+h} f(t_0, x_0)d\tau\right) + \frac{1}{h}\left(\int_{t_0}^{t_0+h} f(t_0, x_0) - f(\tau, x(\tau))d\tau\right) \\
&= \varphi(t_0, x_0, h) - f(t_0, x_0) + \frac{1}{h}\left(\int_{t_0}^{t_0+h} f(t_0, x_0) - f(\tau, x(\tau))d\tau\right)
\end{aligned}$$

Sei nun $K \subset D$ gegeben. Die Funktion $f(t_0 + s, x(t_0 + s; t_0, x_0))$ ist stetig in s, t_0 und x_0, also gleichmäßig stetig für $(s, t_0, x_0) \in [0, h] \times K$ für hinreichend kleines $h > 0$ (so klein, dass die Lösungen $x(t_0 + s; t_0, x_0))$ für $s \in [0, h]$ existieren), da diese Menge kompakt ist. Also existiert eine Funktion $\varepsilon_1(h) \to 0$ mit

$$\|f(\tau, x(\tau)) - f(t_0, x(t_0))\| \le \varepsilon_1(h)$$

für $\tau = t_0 + s \in [t_0, t_0 + h]$ und damit

$$\frac{1}{h}\left\|\int_{t_0}^{t_0+h} f(t_0, x_0) - f(\tau, x(\tau))d\tau\right\| \le \frac{1}{h}\int_{t_0}^{t_0+h} \|f(t_0, x_0) - f(\tau, x(\tau))\|d\tau \le \varepsilon_1(h). \quad (2.7)$$

Wir nehmen nun an, dass (2.6) gilt. Ebenfalls wegen gleichmäßiger Stetigkeit und wegen (2.6) existiert eine Funktion $\varepsilon_2(h) \to 0$ mit

$$\|\varphi(t_0, x_0, h) - f(t_0, x_0)\| \le \varepsilon_2(h).$$

Damit folgt

$$\frac{\|\Phi(t_0, x_0, h) - x(t_0 + h)\|}{h} \le \varepsilon_2(h) + \varepsilon_1(h),$$

also (2.5) mit $\varepsilon(h) = \varepsilon_1(h) + \varepsilon_2(h)$.

Gelte umgekehrt (2.5). Sei $(t, x) \in D$ gegeben und sei $K = \{(t, x)\} \subset D$. Mit (2.5) und (2.7), angewendet mit $(t_0, x_0) = (t, x)$, folgt aus der Gleichung vom Anfang des Beweises

$$\|\varphi(t, x, h) - f(t, x)\| \le \varepsilon(h) + \varepsilon_1(h),$$

also

$$\lim_{h\to 0} \|\varphi(t, x, h) - f(t, x)\| = 0$$

und damit (2.6) wegen der Stetigkeit von φ. □

Mit Hilfe der Bedingung (2.6) prüft man leicht nach, dass das Euler- und das Heun-Verfahren konsistent sind. Die Konsistenzordnung kann man aus (2.6) allerdings nicht ableiten, da die Abschätzung von $\varepsilon(h)$ mittels $\varepsilon_1(h)$ und $\varepsilon_2(h)$ dafür zu grob ist, denn falls $f \not\equiv 0$ ist, gilt $\varepsilon_1(h) \ge O(h)$, so dass man maximal die Konsistenzordnung $p = 1$ nachweisen könnte. Wir werden später sehen, wie man die Konsistenzordnung berechnen kann.

Wir kommen nun zu unserem ersten wichtigen Satz, der besagt, dass Lipschitzbedingung und Konsistenz tatsächlich ausreichend für die Konvergenz sind.

Satz 2.7 Betrachte ein Einschrittverfahren Φ, das die Lipschitzbedingung erfüllt und konsistent ist. Dann ist das Verfahren konvergent. Falls das Verfahren dabei die Konsistenzordnung p besitzt, so besitzt es auch die Konvergenzordnung p.

Beweis: Wir müssen die Eigenschaft aus Definition 2.1(iii) nachprüfen. Sei dazu eine kompakte Menge $K \subset D$ und ein $T > 0$ mit $[t_0, T] \subset I_{t_0,x_0}$ für alle $(t_0, x_0) \in K$ gegeben. Die Menge

$$K_1 := \{(t, x(t; t_0, x_0)) \,|\, (t_0, x_0) \in K, t \in [t_0, T]\}$$

ist dann ebenfalls kompakt, da x stetig in allen Variablen ist und Bilder kompakter Mengen unter stetigen Funktionen wieder kompakt sind. Wir wählen ein $\delta > 0$ und betrachten die kompakte Menge

$$K_2 := \bigcup_{(t,x)\in K_1} \{t\} \times \overline{B}_\delta(x).$$

Die Menge K_2 ist also genau die Menge aller Punkte (t, x), deren x-Komponente einen Abstand $\leq \delta$ von einer Lösung $x(t; t_0, x_0)$ mit $x_0 \in K$ hat. Für hinreichend kleines $\delta > 0$ ist K_2 Teilmenge des Definitionsbereiches D von f, da D offen ist und $K_1 \subset D$ gilt. Das betrachtete Einschrittverfahren ist deswegen konsistent auf K_2 mit einer Funktion $\varepsilon(h)$, wobei $\varepsilon(h) = Eh^p$ im Falle der Konsistenzordnung p ist. Ebenfalls erfüllt Φ auf K_2 die Lipschitzbedingung mit einer Konstanten $L > 0$.

Wir beweisen die Konvergenz nun zunächst unter der folgenden Annahme, deren Gültigkeit wir später beweisen werden:

> Für alle hinreichend feinen Gitter $\mathcal{T}$ und alle Anfangsbedingungen $(t_0, x_0) \in K$ gilt für die gemäß Definition 2.3 erzeugte Gitterfunktion $\tilde{x}$ die Beziehung $(t_i, \tilde{x}(t_i)) \in K_2$ für alle $t_i \in \mathcal{T}$. (2.8)

Zum Beweis der Konvergenz wählen wir eine Anfangsbedingung $(t_0, x_0) \in K$ und schreiben wieder kurz $x(t) = x(t; t_0, x_0)$. Mit $\tilde{x}$ bezeichnen wir die zugehörige numerisch approximierende Gitterfunktion und mit

$$e(t_i) := \|\tilde{x}(t_i) - x(t_i)\|$$

bezeichnen wir den Fehler zur Zeit $t_i \in \mathcal{T}$. Dann gilt nach den Vorüberlegungen am Anfang dieses Abschnitts

$$\begin{aligned}
e(t_i) \;=\; \|\tilde{x}(t_i) - x(t_i)\| \;&\leq\; \|\Phi(t_{i-1}, \tilde{x}(t_{i-1}), \tau_{i-1}) - \Phi(t_{i-1}, x(t_{i-1}), \tau_{i-1})\| \\
&\quad +\; \|\Phi(t_{i-1}, x(t_{i-1}), h_{i-1}) - x(t_i)\| \\
&=\; \|\Phi(t_{i-1}, \tilde{x}(t_{i-1}), h_{i-1}) - \Phi(t_{i-1}, x(t_{i-1}), h_{i-1})\| \\
&\quad +\; \|\Phi(t_{i-1}, x(t_{i-1}), h_{i-1}) - x(t_i; t_{i-1}, x(t_{i-1}))\| \\
&\leq\; (1 + Lh_{i-1})\|\tilde{x}(t_{i-1}) - x(t_{i-1})\| \;+\; h_{i-1}\varepsilon(h_{i-1}) \\
&=\; (1 + Lh_{i-1})e(t_{i-1}) + h_{i-1}\varepsilon(h_{i-1})
\end{aligned}$$

wobei wir im vorletzten Schritt die Lipschitzbedingung und die Konsistenz sowie die Tatsache, dass $(t_{i-1}, \tilde{x}(t_{i-1})) \in K_2$ liegt, ausgenutzt haben. Wir erhalten also für den Fehler $e(t_i)$ die rekursive Gleichung

$$e(t_i) \leq (1 + Lh_{i-1})e(t_{i-1}) + h_{i-1}\varepsilon(h_{i-1})$$

gemeinsam mit der "Anfangsbedingung" $e(t_0) = 0$, da $\tilde{x}(t_0) = x_0 = x(t_0)$ ist.

Mittels Induktion zeigen wir nun, dass daraus die Abschätzung

$$e(t_i) \leq \varepsilon(\bar{h})\frac{1}{L}(\exp(L(t_i - t_0)) - 1)$$

folgt. Für $i = 0$ ist die Abschätzung klar. Für $i - 1 \to i$ verwenden wir

$$\exp(Lh_i) = 1 + Lh_i + \frac{L^2h_i^2}{2} + \ldots \geq 1 + Lh_i$$

und erhalten damit mit der Induktionsannahme

$$\begin{aligned}
e(t_i) &\leq (1 + Lh_{i-1})e(t_{i-1}) + h_{i-1}\varepsilon(h_{i-1}) \\
&\leq (1 + Lh_{i-1})\varepsilon(\bar{h})\frac{1}{L}(\exp(L(t_{i-1} - t_0)) - 1) + h_{i-1}\underbrace{\varepsilon(h_{i-1})}_{\leq \varepsilon(h)} \\
&= \varepsilon(\bar{h})\frac{1}{L}\Big(h_{i-1}L + (1 + Lh_{i-1})(\exp(L(t_{i-1} - t_0)) - 1)\Big) \\
&= \varepsilon(\bar{h})\frac{1}{L}\Big(h_{i-1}L + (1 + Lh_{i-1})\exp(L(t_{i-1} - t_0)) - 1 - Lh_{i-1}\Big) \\
&= \varepsilon(\bar{h})\frac{1}{L}\Big((1 + Lh_{i-1})\exp(L(t_{i-1} - t_0)) - 1\Big) \\
&\leq \varepsilon(\bar{h})\frac{1}{L}\Big(\exp(Lh_{i-1})\exp(L(t_{i-1} - t_0)) - 1\Big) \\
&= \varepsilon(\bar{h})\frac{1}{L}(\exp(L(t_i - t_0)) - 1).
\end{aligned}$$

Damit folgt die Konvergenz und im Falle von $\varepsilon(\bar{h}) \leq E\bar{h}^p$ auch die Konvergenzordnung mit $C = E(\exp(L(T - t_0)) - 1)/L$.

Es bleibt zu zeigen, dass unsere oben gemachte Annahme (2.8) tatsächlich erfüllt ist. Wir zeigen, dass (2.8) für alle Gitter $\mathcal{T}$ gilt, deren maximale Schrittweite $\bar{h}$ die Ungleichung

$$\varepsilon(\bar{h}) \leq \frac{\delta L}{\exp(L(T - t_0)) - 1}$$

erfüllt. Wir betrachten dazu eine Lösung $\tilde{x}$ mit Anfangswert $x_0 \in K$ und beweisen die Annahme per Induktion. Für $\tilde{x}(t_0)$ ist wegen $\tilde{x}(t_0) = x_0$ nichts zu zeigen. Für den Induktionsschritt $i - 1 \to i$ sei $(t_k, \tilde{x}(t_k)) \in K_2$ für $k = 0, 1, \ldots, i - 1$. Wir müssen zeigen, dass $(t_i, \tilde{x}(t_i)) \in K_2$ liegt. Beachte, dass die oben gezeigte Abschätzung

$$e(t_i) \leq \varepsilon(\bar{h})\frac{1}{L}(\exp(L(T - t_0)) - 1)$$

bereits gilt, falls $(t_k, \tilde{x}(t_k)) \in K_2$ liegt für $k = 0, 1, \ldots, i-1$. Mit der Wahl von h folgt damit $e(t_i) \leq \delta$, also

$$\|\tilde{x}(t_i) - x(t_i)\| \leq \delta.$$

Da $(t_i, x(t_i)) \in K_1$ liegt, folgt $(t_i, \tilde{x}(t_i)) \in \{t_i\} \times \overline{B}_\delta(x(t_i)) \subset K_2$, also die gewünschte Beziehung. □

Bemerkung 2.8 (i) Schematisch dargestellt besagt Satz 2.7 das Folgende:

$$\begin{array}{rcl} \text{Lipschitzbedingung + Konsistenz} & \Rightarrow & \text{Konvergenz} \\ \text{Lipschitzbedingung + Konsistenzordnung } p & \Rightarrow & \text{Konvergenzordnung } p \end{array}$$

(ii) Die Schranke für $e(T)$ wächst — sogar sehr schnell — wenn die Intervallgröße $T - t_0$ wächst. Insbesondere lassen sich mit dieser Abschätzung keinerlei Aussagen über das Langzeitverhalten numerischer Lösungen machen, z.B. über Grenzwerte $\tilde{x}(t_i)$ für $t_i \to \infty$. Tatsächlich kann es passieren, dass der "numerische Grenzwert" von $\tilde{x}(t_i)$ für $t_i \to \infty$ für beliebig feine Gitter $\mathcal{T}$ weit von dem tatsächlichen Grenzwert der exakten Lösung $x(t)$ entfernt ist. Wir werden später genauer auf dieses Problem eingehen.

(iii) Der Konsistenzfehler $\varepsilon(h)h$ wird auch als *lokaler Fehler* bezeichnet, während der im Beweis abgeschätzte Fehler $e(t)$ als *globaler Fehler* bezeichnet wird. Im Falle der Konsistenzordnung p gilt $\varepsilon(h)h = O(h^{p+1})$ und $e(t) = O(h^p)$. Man "verliert" also eine Ordnung beim Übergang vom lokalen zum globalen Fehler. Dies lässt sich anschaulich wie folgt erklären: Bis zur Zeit t muss man (bei äquidistantem Gitter) gerade ca. $N(t) = (t - t_0)/h$ Schritte machen, weswegen sich $N(t)$ lokale Fehler aufsummieren, was zu dem globalen Fehler $O(h^{p+1})N(t) = O(h^{p+1})/h = O(h^p)$ führt. □

2.4 Kondition

Wie bei allen numerischen Problemen sollte auch hier die Kondition des Problems "Berechne eine Lösung des Anfangswertproblems (1.1), (1.2)" betrachtet werden. Eine detaillierte Darstellung der hierfür nötigen Theorie würde den Rahmen dieser Vorlesung leider sprengen. Wir werden hier nur kurz (ohne Beweise) beschreiben, wie sich die Kondition bzgl. Störungen Δx_0 im Anfangswert x_0 berechnen lässt, d.h., wir wollen eine Abschätzung für den Ausdruck

$$\kappa := \max_{\Delta x_0 \in \mathbb{R}^n, \|\Delta x_0\| = 1} \left\| \frac{\partial}{\partial x_0} x(t; t_0, x_0) \Delta x_0 \right\|$$

berechnen. Dazu betrachtet man das Anfangswertproblem

$$\dot{y}(t) = f_x(t, x(t; t_0, x_0))y(t), \quad y(t_0) = \Delta x_0, \tag{2.9}$$

wobei $f_x(t, x) = \frac{\partial}{\partial x} f(t, x) \in \mathbb{R}^{n \times n}$ und $x(t; t_0, x_0)$ die Lösung von (1.1), (1.2) ist. Die Lösung von (2.9) lässt sich in der Form

$$y(t; t_0, \Delta x_0) = W(t; t_0)\Delta x_0$$

mit einer Matrix $W(t; t_0) \in \mathbb{R}^{n \times n}$ schreiben. Dieses W ist dann gerade gleich der obigen Ableitung $\frac{\partial}{\partial x_0} x(t; t_0, x_0)$, die Matrix-Norm $\|W(t; t_0)\|$ gibt also gerade die Kondition κ an.

Als Beispiel betrachte die eindimensionale DGL

$$\dot{x}(t) = \lambda x(t)$$

für $\lambda \in \mathbb{R}$. Für diese Gleichung ist $f(t,x) = \lambda x$, also $f_x(t,x) = \lambda$, weswegen (2.9) die Form

$$\dot{y}(t) = \lambda y(t)$$

hat. Die Lösungen sind durch $y(t;t_0,\Delta x_0) = e^{\lambda(t-t_0)}\Delta x_0$ gegeben, es gilt also $W(t;t_0) = e^{\lambda(t-t_0)}$. Die Matrixnorm dieser 1×1-Matrix ist gerade der Betrag, da $e^{\lambda(t-t_0)}$ positiv ist, gilt also

$$\kappa = e^{\lambda(t-t_0)}.$$

Für $t >> t_0$ und $\lambda > 0$ ist das Problem also schlecht konditioniert (κ wird sehr groß), während das Problem für $t >> t_0$ und $\lambda < 0$ sehr gut konditioniert ist, da $\kappa \approx 0$ ist.

Eine ausführliche Diskussion der Kondition für gewöhnliche Differentialgleichungen findet sich im Kapitel 3 des Buches [3].

Kapitel 3

Taylor-Verfahren

Wir werden in diesem Kapitel eine spezielle Klasse von Einschrittverfahren einführen, die in der numerischen Praxis zwar eher selten verwendet werden (wir werden später sehen, wieso), für das Verständnis der weiteren Einschrittverfahren aber sehr nützlich sind.

3.1 Definition

Die Taylor-Verfahren haben ihren Namen von der zu Grunde liegenden Taylor-Formel und gehen in direkter Weise aus diesen hervor. Allerdings wird die Taylor-Formel in zunächst etwas ungewohnt erscheinender Weise angewendet: Wir verwenden den Differentialoperator L_f^i, $i \in \mathbb{N}$, der für (hinreichend oft differenzierbare) Funktionen f, $g : D \to \mathbb{R}^n$ mit $D \subseteq \mathbb{R} \times \mathbb{R}^n$ mittels

$$L_f^0 g(t,x) := g(t,x), \quad L_f^1 g(t,x) := \frac{\partial g}{\partial t}(t,x) + \frac{\partial g}{\partial x}(t,x) f(t,x), \quad L_f^{i+1} g(t,x) = L_f^1 L_f^i g(t,x)$$

definiert ist. Beachte, dass $L_f^i g$ wieder eine Funktion von D nach $\mathbb{R}^n$ ist. Der folgende Satz stellt die hier benötigte Version der Taylor-Formel vor.

Satz 3.1 Gegeben sei eine Differentialgleichung (1.1) mit p-mal stetig differenzierbarem Vektorfeld f. Sei $x(t) = x(t; t_0, x_0)$ eine Lösung dieser Differentialgleichung. Dann gilt

$$x(t) = x_0 + \sum_{i=1}^{p} \frac{(t-t_0)^i}{i!} L_f^{i-1} f(t_0, x_0) + O((t-t_0)^{p+1}),$$

wobei das O-Symbol im Sinne von Definition 2.1(iii) verwendet wird.

Beweis: Aus der Theorie der gewöhnlichen Differentialgleichungen ist bekannt, dass die Lösung $x(t)$ unter der vorausgesetzten Differenzierbarkeitsbedingung an f $p+1$-mal stetig differenzierbar nach t ist. Nach der aus der Analysis bekannten Taylor-Formel für Funktionen von $\mathbb{R}$ nach $\mathbb{R}^n$ gilt demnach

$$x(t) = x_0 + \sum_{i=1}^{p} \frac{(t-t_0)^i}{i!} \frac{d^i x}{dt^i}(t_0) + O((t-t_0)^{p+1}).$$

Zum Beweis des Satzes werden wir nun nachweisen, dass

$$\frac{d^i x}{dt^i}(t) = L_f^{i-1} f(t, x(t)) \tag{3.1}$$

ist für alle $t \in I_{t_0,x_0}$, denn dann folgt die Behauptung aus

$$\frac{d^i x}{dt^i}(t_0) = L_f^{i-1} f(t_0, x(t_0)) = L_f^{i-1} f(t_0, x_0).$$

Wir zeigen (3.1) per Induktion über i. Für $i = 1$ gilt

$$\frac{dx}{dt}(t) = f(t, x(t)) = L_f^0 f(t, x).$$

Für $i \to i+1$ beachte, dass für je zwei differenzierbare Funktionen $g : D \to \mathbb{R}^n$ und $x : \mathbb{R} \to \mathbb{R}^n$ die Gleichung

$$\frac{d}{dt} g(t, x(t)) = \frac{\partial g}{\partial t}(t, x(t)) + \frac{\partial g}{\partial x}(t, x(t)) \frac{d}{dt} x(t)$$

gilt (man nennt dies auch die *totale Ableitung* von g entlang der Funktion $x(t)$). Mit $g(t, x) = L_f^{i-1} f(t, x)$ gilt damit

$$\begin{aligned}
\frac{d^{i+1} x}{d^{i+1} t}(t) &= \frac{d}{dt}\frac{d^i x}{d^i t}(t) = \frac{d}{dt} L_f^{i-1} f(t, x(t)) = \frac{d}{dt} g(t, x(t)) \\
&= \frac{\partial g}{\partial t}(t, x(t)) + \frac{\partial g}{\partial x}(t, x(t)) \frac{d}{dt} x(t) \\
&= \frac{\partial g}{\partial t}(t, x(t)) + \frac{\partial g}{\partial x}(t, x(t)) f(t, x(t)) \\
&= L_f^1 g(t, x(t)) = L_f^1 L_f^{i-1} f(t, x(t)) = L_f^i f(t, x(t)),
\end{aligned}$$

also gerade (3.1). □

Die Idee der Taylor-Verfahren ist nun denkbar einfach: Wir verwenden die Taylor-Formel und lassen den Restterm weg.

Definition 3.2 Das *Taylor-Verfahren der Ordnung* $p \in \mathbb{N}$ ist gegeben durch

$$\Phi(t, x, h) = x + \sum_{i=1}^{p} \frac{h^i}{i!} L_f^{i-1} f(t, x).$$

□

3.2 Eigenschaften

Der folgende Satz gibt die wesentlichen Eigenschaften der Taylor-Verfahren an.

Satz 3.3 Gegeben sei eine Differentialgleichung mit p-mal stetig differenzierbarem Vektorfeld $f : D \to \mathbb{R}^n$. Dann erfüllt das Taylor-Verfahren der Ordnung p die Lipschitzbedingung und ist konsistent mit Konsistenzordnung p.

Beweis: Wir zeigen zunächst die Lipschitzbedingung. Beachte, dass in der Formulierung der Taylor-Verfahren partielle Ableitungen von f bis zur Ordnung $p-1$ auftreten. Jede der auftretenden Funktionen $L_f^{i-1} f$ ist also ein weiteres mal stetig differenzierbar, woraus (mit dem Mittelwertsatz der Differentialrechnung) folgt, dass für jede kompakte Menge $K \subset D$ Lipschitz-Konstanten $L_i > 0$ existieren, so dass $L_f^{i-1} f$ Lipschitz in x mit dieser Konstante ist. Für die Funktion Φ gilt also für alle $h \leq 1$ die Abschätzung

$$\begin{aligned} \|\Phi(t, x_1, h) - \Phi(t, x_2, h)\| &\leq \|x_1 - x_2\| + \sum_{i=1}^{p} \frac{h^i}{i!} L_i \|x_1 - x_2\| \\ &\leq \|x_1 - x_2\| + \sum_{i=1}^{p} h L_i \|x_1 - x_2\| = (1 + Lh)\|x_1 - x_2\| \end{aligned}$$

mit

$$L = \sum_{i=1}^{p} L_i.$$

Dies ist gerade die gewünschte Lipschitz-Bedingung.

Die Konsistenz sowie die behauptete Konsistenzordnung folgt direkt aus Satz 3.1. □

Bemerkung 3.4 Wenn alle auftretenden Ableitungen auf ganz D beschränkt sind, so sind auch die Konstanten in den Lipschitz- und Konsistenzabschätzungen unabhängig von K gültig, man erhält also globale Fehlerabschätzungen. □

Beachte, dass das Taylor-Verfahren der Ordnung $p = 1$ durch

$$\Phi(t, x, h) = x + h L_f^0 f(t, x) = x + h f(t, x).$$

gegeben ist, also gerade das Euler-Verfahren ist. Dies führt sofort zu dem folgenden Korollar.

Korollar 3.5 Falls f einmal stetig differenzierbar ist, so ist das Euler-Verfahren konsistent mit Konsistenzordnung $p = 1$.

Beweis: Das Taylor-Verfahren der Ordnung $p = 1$ ist gerade das Euler-Verfahren, das also nach Satz 3.3 die Konsistenzordnung $p = 1$ besitzt. □

Bemerkung 3.6 Mit einem direkten Beweis kann man die Konsistenzordnung $p = 1$ für das Euler-Verfahren auch beweisen, wenn f nur Lipschitz-stetig (in x *und* t) ist. Die Beweisidee geht wie folgt: Zunächst zeigt man, dass $\|x(t+h) - x(t)\| \leq C_1 |h|$ für ein $C_1 > 0$ und alle hinreichend kleinen h ist; dies verwendet man dann, um

$$\int_t^{t+h} \|f(\tau, x(\tau)) - f(t, x(t))\| d\tau \leq C_2 h^2$$

für ein $C_2 > 0$ zu beweisen. Damit kann man schließlich die Konsistenzordnung zeigen. □

Das Euler-Verfahren ist das einzige Taylor-Verfahren, bei dem keine Ableitungen des Vektorfeldes f auftreten. Das Auftreten der Ableitungen ist tatsächlich der Hauptgrund dafür, dass Taylor-Verfahren in der Praxis eher selten verwendet werden, da man dort Verfahren bevorzugt, die ohne explizite Verwendung der Ableitung funktionieren (auch wenn symbolische Mathematikprogramme wie z.B. MAPLE heutzutage zur automatischen Berechnung der benötigten Ableitungen verwendet werden können). Trotzdem gibt es Spezialanwendungen, in denen Taylor-Verfahren verwendet werden: Für hochgenaue Numerik, bei der Verfahren sehr hoher Ordnung ($p \geq 15$) benötigt werden, sind Taylor-Verfahren nützlich, da sie systematisch für beliebige Konsistenzordnungen hergeleitet werden können und die auftretenden Konstanten (in der Lipschitzbedingung und der Konsistenzabschätzung) durch genaue Analyse der Ableitungen und Restterme exakt abgeschätzt werden können.

Eine der Hauptanwendungen der Taylor-Verfahren bzw. der Taylor-Entwicklung aus Satz 3.1 ist die Konsistenzanalyse beliebiger Einschrittverfahren. Hier gilt der folgende Satz.

Satz 3.7 Sei $f : D \to \mathbb{R}^n$ p-mal stetig differenzierbar. Gegeben sei ein Einschrittverfahren $\Phi : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$, das $p+1$-mal stetig differenzierbar ist. Dann besitzt Φ genau dann die Konsistenzordnung $p \in \mathbb{N}$, wenn die Bedingungen

$$\Phi(t,x,0) = x \quad \text{und} \quad \frac{\partial^i \Phi}{\partial h^i}(t,x,0) = L_f^{i-1} f(t,x) \text{ für } i = 1,\dots,p \tag{3.2}$$

für alle $(t,x) \in D$ gelten.

Beweis: Es bezeichne $\Phi_{T,p}$ das Taylor-Verfahren der Ordnung p. Die Taylor-Entwicklung von Φ nach der Variablen h in $h = 0$ ist gegeben durch

$$\Phi(t,x,h) = \Phi(t,x,0) + \sum_{i=1}^{p} \frac{h^i}{i!} \frac{\partial^i \Phi}{\partial h^i}(t,x,0) + O(h^{p+1}).$$

Sei nun (3.2) erfüllt. Dann liefert der Koeffizientenvergleich mit $\Phi_{T,p}$

$$\Phi(t,x,h) = \Phi_{T,p}(t,x,h) + O(h^{p+1})$$

Aus Satz 3.3 folgt daher

$$x(t+h;t,x) = \Phi_{T,p}(t,x,h) + O(h^{p+1}) = \Phi(t,x,h) + O(h^{p+1}),$$

was die Konsistenz zeigt.

Falls (3.2) nicht erfüllt ist, so gibt es $(t,x) \in D$, so dass entweder $\Phi(t,x,0) \neq x$ gilt (in diesem Fall setzen wir $i^* = 0$) oder

$$\frac{\partial^{i^*} \Phi}{\partial h^{i^*}}(t,x,0) \neq L_g^{i^*-1} f(t,x)$$

für ein $i^* \in \{1, \ldots, p\}$ gilt. Wenn wir i^* minimal mit dieser Eigenschaft wählen, so folgt aus dem Koeffizientenvergleich mit $\Phi_{T,p}$, dass ein $C > 0$ existiert, so dass für alle hinreichend kleinen $h > 0$ die Ungleichung

$$\|\Phi(t, x, h) - \Phi_{T,p}(t, x, h)\| > Ch^{i^*}$$

gilt. Mit Satz 3.3 und der umgekehrten Dreiecksungleichung erhalten wir daher

$$\|x(t + h, t, x) - \Phi(t, x, h)\| > Ch^{i^*} - O(h^{p+1}) > \widetilde{C}h^{i^*}$$

für geeignetes $0 < \widetilde{C} < C$ und alle hinreichend kleinen $h > 0$, was der Konsistenz widerspricht. Also folgt die behauptete Äquivalenz. □

Mit diesem Satz können wir die Konsistenzordnung beliebiger Einschrittverfahren überprüfen. Beachte, dass die Aussage über die Ordnung nur stimmt, wenn das Vektorfeld f hinreichend oft differenzierbar ist. Verfahren mit hoher Konsistenzordnung verlieren diese typischerweise, wenn das Vektorfeld der zu lösendenden DGL nicht die nötige Differenzierbarkeit besitzt!

Ein wesentlicher Nachteil dieses Satzes ist, dass die Ausdrücke $L_f^i f(t, x)$ für große i sehr umfangreich und kompliziert werden. Hier können — wie bereits erwähnt — symbolische Mathematikprogramme wie MAPLE bei den Rechnungen helfen. Das folgende MAPLE Programm berechnet die Ableitungen $L_f^i f(t, x)$ für $i = 0, \ldots, p$. (Vor der Ausführung muss der Variablen `p` natürlich ein Wert zugewiesen werden.)

```
> L[0]:=f(t,x);
> for i from 1 to p do
>   L[i] := simplify(diff(L[i-1],t) + diff(L[i-1],x)*f(t,x));
> od;
```

Die Ausgabe für `p:=3` ist

$$\begin{aligned}
L_0 &:= \mathrm{f}(t, x) \\
L_1 &:= (\frac{\partial}{\partial t}\mathrm{f}(t, x)) + (\frac{\partial}{\partial x}\mathrm{f}(t, x))\,\mathrm{f}(t, x) \\
L_2 &:= (\frac{\partial^2}{\partial t^2}\mathrm{f}(t, x)) + 2\,(\frac{\partial^2}{\partial x\,\partial t}\mathrm{f}(t, x))\,\mathrm{f}(t, x) + (\frac{\partial}{\partial x}\mathrm{f}(t, x))\,(\frac{\partial}{\partial t}\mathrm{f}(t, x)) \\
&\quad + (\frac{\partial^2}{\partial x^2}\mathrm{f}(t, x))\,\mathrm{f}(t, x)^2 + \mathrm{f}(t, x)\,(\frac{\partial}{\partial x}\mathrm{f}(t, x))^2 \\
L_3 &:= (\frac{\partial^3}{\partial t^3}\mathrm{f}(t, x)) + 3\,(\frac{\partial^3}{\partial x\,\partial t^2}\mathrm{f}(t, x))\,\mathrm{f}(t, x) + 3\,(\frac{\partial^2}{\partial x\,\partial t}\mathrm{f}(t, x))\,(\frac{\partial}{\partial t}\mathrm{f}(t, x)) \\
&\quad + (\frac{\partial}{\partial x}\mathrm{f}(t, x))\,(\frac{\partial^2}{\partial t^2}\mathrm{f}(t, x)) + 3\,(\frac{\partial^3}{\partial x^2\,\partial t}\mathrm{f}(t, x))\,\mathrm{f}(t, x)^2 \\
&\quad + 3\,(\frac{\partial^2}{\partial x^2}\mathrm{f}(t, x))\,\mathrm{f}(t, x)\,(\frac{\partial}{\partial t}\mathrm{f}(t, x)) + (\frac{\partial}{\partial t}\mathrm{f}(t, x))\,(\frac{\partial}{\partial x}\mathrm{f}(t, x))^2 \\
&\quad + 5\,\mathrm{f}(t, x)\,(\frac{\partial}{\partial x}\mathrm{f}(t, x))\,(\frac{\partial^2}{\partial x\,\partial t}\mathrm{f}(t, x)) + (\frac{\partial^3}{\partial x^3}\mathrm{f}(t, x))\,\mathrm{f}(t, x)^3 \\
&\quad + 4\,(\frac{\partial^2}{\partial x^2}\mathrm{f}(t, x))\,\mathrm{f}(t, x)^2\,(\frac{\partial}{\partial x}\mathrm{f}(t, x)) + \mathrm{f}(t, x)\,(\frac{\partial}{\partial x}\mathrm{f}(t, x))^3
\end{aligned}$$

Diese Ausdrücke gelten für den skalaren Fall $x \in \mathbb{R}$, für höhere Dimensionen muss das MAPLE-Programm erweitert werden.

Bemerkung 3.8 Man sieht, dass die Ausdrücke tatsächlich sehr unübersichtlich werden; ebenso ist das natürlich bei den entsprechenden Termen der Einschrittverfahren. Eine Hilfe hierfür bietet ein Formalismus, der von dem neuseeländischen Mathematiker J.C. Butcher in den 1960er Jahren entwickelt wurde, und bei dem die auftretenden Ableitungen mittels einer grafischen Repräsentierung in einer Baumstruktur übersichtlich strukturiert werden. □

Kapitel 4

Explizite Runge-Kutta-Verfahren

In diesem Kapitel kommen wir zu einer der wichtigsten Klassen von Einschrittverfahren, zu denen z.B. das Euler- und das Heun-Verfahren gehören.

4.1 Definition

Bei der Konstruktion des Heun-Verfahrens haben wir das Euler-Verfahren verwendet, um einen Schätzwert für den unbekannten Wert $x(t_{i+1})$ zu erhalten. Es liegt nun nahe, diese Methode systematisch rekursiv anzuwenden, um zu Verfahren höherer Konsistenzordnung zu gelangen. Genau dies ist die Grundidee der Runge-Kutta-Verfahren.

Um die dabei entstehenden Verfahren übersichtlich zu schreiben, benötigen wir einen geeigneten Formalismus. Wir erläutern diesen am Beispiel des Heun-Verfahrens

$$\Phi(t,x,h) = x + \frac{h}{2}\Big(f(t,x) + f\Big(t+h, x+hf(t,x)\Big)\Big).$$

Wir schreiben dieses nun als

$$\begin{aligned} k_1 &= f(t,x) \\ k_2 &= f(t+h, x+hk_1) \\ \Phi(t,x,h) &= x + h\left(\frac{1}{2}k_1 + \frac{1}{2}k_2\right) \end{aligned}$$

Was zunächst vielleicht komplizierter als die geschlossene Formel aussieht, erweist sich als sehr günstige Schreibweise, wenn man weitere k_i-Terme hinzufügen will. Dies ist gerade die Schreibweise der expliziten Runge-Kutta-Verfahren.

Definition 4.1 Ein *s-stufiges explizites Runge-Kutta-Verfahren* ist gegeben durch

$$k_i = f\left(t + c_i h,\, x + h\sum_{j=1}^{i-1} a_{ij}k_j\right) \quad \text{für } i = 1,\ldots,s$$

$$\Phi(t,x,h) \quad = \quad x + h\sum_{i=1}^{s} b_i k_i.$$

Den Wert $k_i = k_i(t,x,h)$ bezeichnen wir dabei als i-te Stufe des Verfahrens. □

Die Koeffizienten eines Runge-Kutta-Verfahrens können wir mittels

$$b = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_s \end{pmatrix} \in \mathbb{R}^s, \quad c = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_s \end{pmatrix} \in \mathbb{R}^s, \quad \mathcal{A} = \begin{pmatrix} 0 & & & & \\ a_{21} & 0 & & & \\ a_{31} & a_{32} & 0 & & \\ \vdots & \vdots & \ddots & \ddots & \\ a_{s1} & \cdots & \cdots & a_{s,s-1} & 0 \end{pmatrix} \in \mathbb{R}^{s\times s}$$

kompakt schreiben. Konkrete Verfahren werden meist in Form des Butcher-Tableaus (oder Butcher-Schemas)

$$\begin{array}{c|cccc} c_1 & & & & \\ c_2 & a_{2\,1} & & & \\ c_3 & a_{3\,1} & a_{3\,2} & & \\ \vdots & \vdots & \vdots & \ddots & \\ c_s & a_{s\,1} & a_{s\,2} & \cdots & a_{s\,s-1} \\ \hline & b_1 & b_2 & \cdots & b_{s-1} \quad b_s \end{array}$$

geschrieben, das wiederum auf J.C. Butcher zurückgeht.

Einfache Beispiele solcher Verfahren sind das Euler-Verfahren ($s = 1$), das Heun-Verfahren ($s = 2$) und das sogenannte *klassische Runge-Kutta-Verfahren* ($s = 4$), das von C. Runge[1] und M. Kutta[2] entwickelt wurde, und dem die ganze Verfahrensklasse ihren Namen verdankt. Diese Verfahren sind (von links nach rechts) gegeben durch die Butcher-Tableaus

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array} \qquad \begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \qquad \begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ \frac{1}{2} & 0 & \frac{1}{2} & \\ 1 & 0 & 0 & 1 \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} \quad \frac{1}{6} \end{array}$$

Beachte, dass das Euler-Verfahren sowohl das einfachste Runge-Kutta-Verfahren als auch das einfachste Taylor-Verfahren ist; es ist das einzige Verfahren, das in beiden Klassen liegt, da alle Runge-Kutta-Verfahren per Definition ohne Ableitungen von f auskommen, was gegenüber den Taylor-Verfahren einen großen Vorteil darstellt.

[1] deutscher Mathematiker, 1856–1927
[2] deutscher Mathematiker und Ingenieur, 1867–1944

Es ist in diesem Zusammenhang interessant, den Aufwand des Heun-Verfahrens und des Taylor-Verfahrens der Ordnung 2 z.B. für $x \in \mathbb{R}$ zu vergleichen, die ja die gleiche Konsistenzordnung besitzen. Beim Taylor-Verfahren der Ordnung 2 müssen in jedem Schritt $L_f^0 f(t,x) = f(t,x)$ und

$$L_f^1 f(t,x) = \frac{\partial f}{\partial t}(t,x) + \frac{\partial f}{\partial x}(t,x) f(t,x)$$

ausgewertet werden, also 3 Funktionsauswertungen; beim Heun Verfahren müssen $k_1 = f(t,x)$ und $f(t+h, x+hk_1)$, also 2 Funktionen ausgewertet werden. Der Aufwand ist folglich nur 2/3 so groß. Dieser geringere Aufwand, der bei höherer Konsistenzordnung noch deutlicher ausfällt, ist typisch für Runge-Kutta-Verfahren, ein weiterer Vorteil gegenüber den Taylor-Verfahren.

Beachte, dass Runge-Kutta-Verfahren immer die Lipschitz-Bedingung erfüllen, wenn das Vektorfeld f Lipschitz-stetig im Sinne des Eindeutigkeitssatzes 1.4 ist: Mittels Induktion sieht man leicht, dass jede Stufe k_i Lipschitz-stetig ist. Damit gilt dies auch für ihre Summe, weswegen Φ die gewünschte Bedingung erfüllt.

4.2 Konsistenz

Wir wollen nun untersuchen, wie sich die Konsistenzeigenschaften der Runge-Kutta-Verfahren über ihre Koeffizienten auszudrücken lassen. Das erste wichtige Resultat ist das folgende Lemma.

Lemma 4.2 Ein explizites Runge-Kutta-Verfahren ist genau dann konsistent, wenn die Bedingung

$$\sum_{i=1}^{s} b_i = 1$$

erfüllt ist.

Beweis: Beachte, dass ein Runge-Kutta-Verfahren von der Form

$$\Phi(t,x,h) = x + h\varphi(t,x,h)$$

mit

$$\varphi(t,x,h) = \sum_{i=1}^{s} b_i k_i(t,x,h)$$

ist. Nach Lemma 2.6 ist das Verfahren also genau dann konsistent, wenn

$$\varphi(t,x,0) = \sum_{i=1}^{s} b_i k_i(t,x,0) = f(t,x)$$

ist. Aus Definition 4.1 folgt sofort, dass $k_i(t,x,0) = f(t,x)$, also ist das Verfahren genau dann konsistent, falls $\sum_{i=1}^{s} b_i f(t,x) = f(t,x)$, was für beliebige f dann und nur dann der Fall ist, wenn $\sum_{i=1}^{s} b_i = 1$ ist. □

Etwas schwieriger wird die Sache, wenn wir Aussagen über die Konsistenzordnung machen wollen. Zunächst wollen wir eine obere Schranke für die Konsistenz beweisen.

Lemma 4.3 Für ein s-stufiges explizites Runge-Kutta-Verfahren Φ mit Konsistenzordnung p gilt die Ungleichung $p \leq s$, d.h. die Konsistenzordnung ist maximal so groß wir die Stufenzahl.

Beweis: Wir wenden das Verfahren auf das Anfangswertproblem

$$\dot{x}(t) = x(t), \;\; x(0) = 1$$

an. Für die exakte Lösung gilt hier

$$x(h;0,1) = e^h = 1 + h + \frac{h^2}{2!} + \cdots + \frac{h^s}{s!} + \frac{h^{s+1}}{(s+1)!} + O(h^{s+2}).$$

Andererseits sieht man durch Induktion über i, dass $k_i(0,1,\cdot) \in \mathcal{P}_{i-1}$ ist, also ein Polynom vom Grad $\leq i-1$ in h ist. Also ist $\Phi(0,1,\cdot) \in \mathcal{P}_s$, weswegen in $\Phi(0,1,h)$ kein Term der Form ah^{s+1} auftreten kann. Daher gilt für jede Konstante $C > 0$ und hinreichend kleines $h > 0$ die Abschätzung

$$\|x(h;0,1) - \Phi(0,1,h)\| \geq \frac{h^{s+1}}{(s+1)!} - O(h^{s+2}) \geq \left(\frac{1}{h(s+1)!} - \widetilde{C}\right) h^{s+2} \geq Ch^{s+2},$$

weswegen die Konsistenzordnung maximal s sein kann, also $p \leq s$ gilt. □

Um nun genauere Aussagen über die Konsistenzordnung zu machen, empfiehlt es sich, die zu betrachtenden Differentialgleichungen etwas zu vereinfachen: Wir wollen uns auf autonome DGL einschränken. Damit wir trotzdem Aussagen für allgemeine Probleme erhalten können, überlegen wir uns zuerst, dass dies keine echte Einschränkung ist. Tatsächlich kann man aus jeder Differentialgleichung

$$\dot{x}(t) = f(t, x(t)), \quad x(t_0) = x_0 \tag{4.1}$$

mittels

$$y = \begin{pmatrix} x \\ s \end{pmatrix}, \quad \hat{f}(y) = \begin{pmatrix} f(s,x) \\ 1 \end{pmatrix}$$

(mit $s \in \mathbb{R}$) eine *autonome* Differentialgleichung

$$\dot{y}(t) = \hat{f}(y(t)), \quad y(t_0) = y_0 = \begin{pmatrix} x_0 \\ t_0 \end{pmatrix} \tag{4.2}$$

machen, für deren Lösungen die Beziehung

$$y(t;t_0,y_0) = \begin{pmatrix} x(t;t_0,x_0) \\ t \end{pmatrix} \tag{4.3}$$

gilt. Die ursprüngliche Lösung $x(t;t_0,x_0)$ von (4.1) findet sich also gerade in den ersten n Komponenten der $n+1$-dimensionalen Lösung $y(t;t_0,y_0)$ der autonomen Gleichung (4.2) wieder. Mit anderen Worten kann jede DGL im $\mathbb{R}^n$ in eine autonome DGL im $\mathbb{R}^{n+1}$ umgewandelt werden, dieses Verfahren nennt man *Autonomisierung*. Beachte, dass die neue DGL die Bedingungen des Eindeutigkeitssatzes nur dann erfüllt, wenn f Lipschitz-stetig bezüglich x *und* t ist, was eine stärkere Forderung als die Lipschitz-Stetigkeit bzgl. x ist. Da

wir diese Bedingung für unsere numerischen Aussagen aber sowieso immer benötigen (meist nehmen wir ja sogar Differenzierbarkeit von f bzgl. x und t an), stellt diese Annahme für unsere numerischen Untersuchungen keine Einschränkung dar.

Wir betrachten nun die von einem Runge-Kutta-Verfahren Φ erzeugten approximativen Lösungen $\tilde{x}(t_i)$ und $\tilde{y}(t_i)$ der Gleichungen (4.1) und (4.2). Unser Ziel ist es, uns bei der folgenden Konsistenzordnungsanalyse auf autonome Gleichungen einzuschränken. Damit wir dabei trotzdem Resultate für allgemeine nichtautonome Gleichungen erhalten können, also die für (4.2) gültigen Resultate auf (4.1) übertragen können, muss hier die zu (4.3) analoge Beziehung

$$\tilde{y}(t_i) = \begin{pmatrix} \tilde{x}(t_i) \\ t_i \end{pmatrix} \tag{4.4}$$

gelten. Ein Runge-Kutta-Verfahren, das (4.4) erfüllt, wird *invariant unter Autonomisierung* genannt. Nicht jedes Runge-Kutta-Verfahren ist aber invariant unter Autonomisierung. Das folgende Lemma gibt die dafür notwendige und hinreichende Bedingung an.

Lemma 4.4 Ein explizites Runge-Kutta-Verfahren ist genau dann invarant unter Autonomisierung, wenn es konsistent ist und die Bedingung

$$c_i = \sum_{j=1}^{i-1} a_{ij}$$

für $i = 1, \ldots, s$ erfüllt ist.

Beweis: Wir bezeichnen das Verfahren für (4.1) mit Φ und das Verfahren für (4.2) mit $\widehat{\Phi}$, die zugehörigen Stufen bezeichnen wir mit k_i und $\hat{K}_i = (\hat{k}_i, \theta_i)^T$. Das Verfahren ist genau dann invariant unter Autonomisierung, wenn

$$\widehat{\Phi}(t, x, h) = \begin{pmatrix} \Phi(t, x, h) \\ t + h \end{pmatrix} \tag{4.5}$$

gilt, da sich (4.4) dann mittels Induktion über i ergibt. Wegen

$$\widehat{\Phi}(t, x, h) = \begin{pmatrix} x + h \sum_{i=1}^{s} b_i \hat{k}_i \\ t + h \sum_{i=1}^{s} b_i \theta_i \end{pmatrix} \quad \text{und} \quad \Phi(t, x, h) = x + h \sum_{i=1}^{s} b_i k_i$$

gilt (4.5) genau dann, wenn

$$\hat{k}_i = k_i \quad \text{und} \quad t + h \sum_{i=1}^{s} b_i \theta_i = t + h \tag{4.6}$$

erfüllt ist. Für $\hat{k}_i$ und θ_i gilt gerade

$$\begin{pmatrix} \hat{k}_i \\ \theta_i \end{pmatrix} = \begin{pmatrix} f\left(t + h \sum_{j=1}^{i-1} a_{ij} \theta_j,\ x + h \sum_{j=1}^{i-1} a_{ij} \hat{k}_j\right) \\ 1 \end{pmatrix}.$$

Wegen $k_i = f\left(t + c_i h,\ x + h \sum_{j=1}^{i-1} a_{ij} k_j\right)$ und $\theta_j = 1$ ergibt sich, dass die erste Gleichung in (4.6) genau dann gilt, wenn $c_i = \sum_{j=1}^{i-1} a_{ij}$ gilt. Wegen $\theta_i = 1$ gilt die zweite Gleichung

in (4.6) genau dann, wenn $t + h\sum_{i=1}^{s} b_i = t + h$ erfüllt ist, also wenn $\sum_{i=1}^{s} b_i = 1$ ist, was gerade äquivalent zur Konsistenz ist. □

Auf Basis dieses Lemmas können wir uns also im Folgenden auf autonome DGL einschränken, wenn wir Verfahren betrachten, die die Bedingung von Lemma 4.4 erfüllen. Dies hat den Vorteil, dass sich der Differentialoperator L_f^1 zu

$$L_f^1 g(x) := \left(\frac{d}{dx} g(x)\right) f(x)$$

vereinfacht, was die Taylorentwicklung deutlich übersichtlicher macht. Dies wird im folgenden Satz ausgenutzt.

Satz 4.5 Betrachte ein Runge-Kutta-Verfahren, das die Bedingung aus Lemma 4.4 erfüllt. Dann gilt für alle Vektorfelder $f \in C^p(D, \mathbb{R}^n)$:

(i) Das Verfahren besitzt genau dann die Konsistenzordnung $p = 1$, wenn die Gleichung

$$\sum_i b_i = 1$$

gilt.

(ii) Es besitzt genau dann die Konsistenzordnung $p = 2$, wenn zusätzlich zu (i) die Gleichung

$$\sum_i b_i c_i = 1/2$$

gilt.

(iii) Es besitzt genau dann die Konsistenzordnung $p = 3$, wenn zusätzlich zu (i), (ii) die Gleichungen

$$\sum_i b_i c_i^2 = 1/3, \quad \sum_{ij} b_i a_{ij} c_j = 1/6$$

gelten.

(iv) Es besitzt genau dann die Konsistenzordnung $p = 4$, wenn zusätzlich zu (i)–(iii) die Gleichungen

$$\sum_i b_i c_i^3 = 1/4, \qquad \sum_{ij} b_i a_{ij} c_i c_j = 1/8$$

$$\sum_{ij} b_i a_{ij} c_j^2 = 1/12, \qquad \sum_{ijk} b_i a_{ij} a_{jk} c_k = 1/24$$

gelten.

Hierbei laufen die Summations-Indizes in den Grenzen $i = 1, \ldots, s$, $j = 1, \ldots, i-1$ und $k = 1, \ldots, j-1$.

Beweis: Wir beweisen das Folgende: Die Gleichungen ergeben sich aus der Bedingung (3.2) in Satz 3.7, wobei die für $p \in \mathbb{N}$ angegebenen Gleichungen gerade äquivalent zu der Bedingung

$$\frac{\partial^p \Phi}{\partial h^p}(x, 0) = L_f^{p-1} f(x) \tag{4.7}$$

aus (3.2) sind.

Für $p = 1$ beweist man dies mit gleichen Rechnungen wie im Beweis von Lemma 4.2. Wir zeigen die Behauptung hier exemplarisch für $p = 2$, die höheren Ordnungen folgen mit der gleichen Beweistechnik, allerdings mit aufwändigeren Rechnungen.

Wir zeigen also, dass die in (ii) angegebene Gleichung äquivalent zu (4.7) für $p = 2$ ist. Die zweite Ableitung von $\Phi = x + h\varphi$ nach h ist gerade

$$\begin{aligned}\frac{\partial^2\Phi}{\partial h^2} &= \frac{\partial}{\partial h}\frac{\partial}{\partial h}(x + h\varphi) = \frac{\partial}{\partial h}\left(\varphi + h\frac{\partial}{\partial h}\varphi\right)\\ &= \frac{\partial}{\partial h}\varphi + \frac{\partial}{\partial h}\varphi + h\frac{\partial^2}{\partial h^2}\varphi = 2\frac{\partial}{\partial h}\varphi + h\frac{\partial^2}{\partial h^2}\varphi\end{aligned}$$

In $h = 0$ ergibt sich damit

$$\frac{\partial^2\Phi}{\partial h^2}(x,0) = 2\frac{\partial}{\partial h}\varphi(x,0) = 2\sum_{i=1}^{s} b_i \sum_{j=1}^{i-1} a_{ij}\left(\frac{d}{dx}f(x)\right) f(x).$$

Andererseits ist die Ableitung $L_f^1 f(x)$ gerade durch

$$L_f^1 f(x) = \left(\frac{d}{dx}f(x)\right) f(x)$$

gegeben ist. Damit diese Ausdrücke für alle $f(x)$ übereinstimmen, muss also gerade

$$2\sum_{i=1}^{s} b_i \sum_{j=1}^{i-1} a_{ij} = 1$$

gelten, was wegen der angenommen Autonomieinvarianzbedingung

$$c_i = \sum_{j=1}^{i-1} a_{ij}$$

genau dann der Fall ist, wenn die Gleichung aus (ii) erfüllt ist. □

Diese Gleichungen an die Koeffizienten werden *Bedingungsgleichungen* genannt. Wie komplex das Problem des Aufstellens der Bedingungsgleichungen für große p wird, zeigt die folgende Tabelle, die die Anzahl der Gleichungen für gegebenes p angibt.

Konsistenzordnung p	1	2	3	4	5	6	7	8	9	10	20
Anzahl Bedingungsgl'en	1	2	4	8	17	37	85	200	486	1205	20247374

Nicht nur das Aufstellen, auch das Lösen dieser (nichtlinearen!) Gleichungssysteme wird ziemlich komplex. Hier kommt wieder das in Bemerkung 3.8 bereits erwähnte grafische Verfahren von Butcher ins Spiel. Mit diesem Verfahren können die einzelnen Terme der $L_f^i f$-Ableitungen ebenso wie die Terme der Ableitungen von Φ mittels einer Baumstruktur grafisch dargestellt werden. Dieses Verfahren erlaubt eine Einsicht in die Struktur dieser riesigen nichtlinearen Gleichungssysteme, womit es gelungen ist, die Gleichungen bis $p = 10$

(ohne Computerhilfe) zu lösen. Eine wichtige Rolle spielt dabei natürlich die Stufenzahl s der betrachtetem Verfahren. Insbesondere ist hierbei wichtig, wie viele Stufen s man zur Realisierung einer gegebenen Konsistenzordnung p benötigt. Die folgende Tabelle gibt die ebenfalls durch Butcher (in den Jahren 1964–1985) berechneten bekannten minimalen Schranken an.

Konsistenzordnung p	1	2	3	4	5	6	7	8	≥ 9
minimale Stufenzahl s	1	2	3	4	6	7	9	11	$\geq p+3$

Der Eintrag für $p \geq 9$ bedeutet nicht, dass für jedes $p \geq 9$ ein Verfahren mit $s = p + 3$ Stufen bekannt ist, sondern dass es kein Verfahren mit weniger Stufen geben kann. Für $p = 10$ wurde 1978 von E. Hairer ein Verfahren mit $s = 17$ Stufen angegeben, das sich im Guinness-Buch der Rekorde findet. Möglichst wenig Stufen zu verwenden ist allerdings nicht das einzige Qualitätsmerkmal für Runge-Kutta-Verfahren, oftmals spielen andere Kriterien eine wichtigere Rolle. Wir kommen später darauf zurück.

Kapitel 5

Implizite Runge-Kutta-Verfahren

5.1 Definition

Bisher haben wir Runge-Kutta-Verfahren betrachtet, bei denen die Koeffizientenmatrix die Form

$$A = \begin{pmatrix} 0 & & & & \\ a_{21} & 0 & & & \\ a_{31} & a_{32} & 0 & & \\ \vdots & \vdots & \ddots & \ddots & \\ a_{s1} & \cdots & \cdots & a_{s,s-1} & 0 \end{pmatrix} \in \mathbb{R}^{s\times s}$$

hatte. Es stellt sich nun die Frage, was passiert, wenn wir hier "volle" Matrizen der Form

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1s} \\ \vdots & & \vdots \\ a_{s1} & \cdots & a_{ss} \end{pmatrix} \in \mathbb{R}^{s\times s}$$

zulassen. Zunächst einmal können wir auch mit solchen Koeffizienten ganz formal durch Erweiterung von Definition 4.1 wieder Runge-Kutta-Verfahren definieren.

Definition 5.1 Ein *s-stufiges implizites Runge-Kutta-Verfahren* ist gegeben durch

$$k_i = f\left(t + c_i h,\, x + h\sum_{j=1}^{s} a_{ij}k_j\right) \quad \text{für } i = 1,\dots,s$$

$$\Phi(t,x,h) = x + h\sum_{i=1}^{s} b_i k_i.$$

Den Wert $k_i = k_i(t,x,h)$ bezeichnen wir dabei als i-te Stufe des Verfahrens. □

Der Grund für den Namen *implizites Verfahren* liegt darin, dass die Definition der k_i nun keine "Zuweisung" mehr ist, sondern ein $s\cdot n$-dimensionales nichtlineares Gleichungssystem

bildet, dessen Lösung gerade der Vektor $k^T = (k_1^T, \ldots, k_s^T) \in \mathbb{R}^{s \cdot n}$ ist. Die Werte $k_i \in \mathbb{R}^n$ sind also *implizit* definiert.

Das einfachste Verfahren dieser Klasse ist durch das Butcher-Tableau

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

gegeben. Ausgeschrieben lautet es

$$k_1 = f(t + h, x + hk_1), \quad \Phi(t, x, h) = x + hk_1,$$

die dadurch erzeugt Gitterfunktion ist rekursiv gegeben durch

$$\tilde{x}(t_{i+1}) = \tilde{x}(t_i) + h_i f(t_{i+1}, \tilde{x}(t_{i+1})).$$

Dieses Verfahren heißt *implizites Euler-Verfahren* und besitzt genau wie sein explizites Gegenstück die Konsistenzordnung $p = 1$. Beachte, dass hier tatsächlich in jedem Schritt ein nichtlineares Gleichungssystem gelöst werden muss. Implizite Runge-Kutta-Verfahren mit Konsistenzordnung $p = 2$ sind z.B. die implizite Mittelpunktregel oder die implizite Trapezregel, die durch

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array} \quad \text{bzw.} \quad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

gegeben sind.

Wir werden später sehen, dass implizite Verfahren für manche Differentialgleichungen gegenüber den expliziten Verfahren deutliche Vorteile besitzen. Zunächst wollen wir uns aber Gedanken darüber machen, wie solch ein implizites Verfahren implementiert werden kann, d.h., wie wir das nichtlineare Gleichungssystem zur Berechnung der k_i lösen können.

5.2 Lösbarkeit und Implementierung

Zunächst einmal gibt es manchmal die Möglichkeit, die entstehenden Gleichungen per Hand in explizite Form zu bringen. Betrachten wir z.B. das implizite Euler-Verfahren angewendet auf die eindimensionale DGL

$$\dot{x}(t) = \lambda x(t),$$

so erhalten wir

$$k_1 = f(t + h, x + hk_1) = \lambda(x + hk_1) = \lambda x + h\lambda k_1,$$

woraus für hinreichend kleine h die Gleichung

$$k_1 = \frac{\lambda x}{1 - h\lambda}$$

folgt.

Oft kommt man mit dieser Strategie aber nicht weiter, wir müssen das entstehende Gleichungssystem

$$k = F(k)$$

mit

$$k = \begin{pmatrix} k_1 \\ \vdots \\ k_s \end{pmatrix} \in \mathbb{R}^{s\cdot n} \quad \text{und} \quad F(k) = \begin{pmatrix} f\left(t + c_1 h,\, x + h\sum_{j=1}^{s} a_{1j}k_j\right) \\ \vdots \\ f\left(t + c_s h,\, x + h\sum_{j=1}^{s} a_{sj}k_j\right) \end{pmatrix}$$

also numerisch lösen.

Eine einfache Möglichkeit hierzu beruht auf der Tatsache, dass f nach Voraussetzung Lipschitz-stetig mit Konstante L ist. Hieraus folgt sofort, dass auch die Abbildung F Lipschitz-stetig mit Konstante hL ist. Falls $hL =: K < 1$ ist, folgt damit

$$\|F(k^1) - F(k^2)\| \leq K\|k^1 - k^2\|,$$

so dass F eine Kontraktion ist, weswegen der Vektor k mittels der aus der Einführung in die Numerik bekannten Fixpunktiteration

$$k^{(j+1)} = F(k^{(j)}) \tag{5.1}$$

berechnet werden kann. Als Startwert für diese Iteration empfiehlt es sich, im ersten Schritt $k_i^{(0)} = f(t+c_i h, x)$ und in den folgenden Schritten den Wert von k aus dem vorhergehenden Schritt zu verwenden. Ein geeignetes Abbruchkriterium ergibt sich wie in der Einführung in die Numerik diskutiert aus dem Banach'schen Fixpunktsatz: Die Iteration wird so lange durchgeführt, bis

$$\|k^{(j+1)} - k^{(j)}\| \leq \varepsilon$$

für eine vorgegebene Toleranz ε ist, damit ist dann die Genauigkeit

$$\|k^{(j+1)} - k^*\| \leq \frac{hL}{1-hL}\varepsilon$$

garantiert, wobei k^* die exakte Lösung bezeichnet. Als Letztes müssen wir uns noch überlegen, wie ε gewählt werden sollte. Damit das Verfahren

$$\Phi(t, x, h) = x + h\sum_{i=1}^{s} b_i k_i$$

den Konsistenzfehler $O(h^{p+1})$ einhält, sollte $\|k^{(j+1)} - k^*\| \leq \varepsilon_0 h^p$ für ein $\varepsilon_0 > 0$ gelten. Damit diese Schranke eingehalten wird, muss

$$\frac{hL}{1-hL}\varepsilon \leq \varepsilon_0 h^p$$

gelten, was für kleine h gerade durch die Wahl $\varepsilon \approx \varepsilon_0 h^{p-1}$ garantiert wird. Das Abbruchkriterium hängt für $p \geq 2$ also von der Schrittweite h ab.

Die Iteration (5.1) wird auch *Gesamtschrittiteration* genannt. Eine einfache Modifikation dieser Iteration ist die *Einzelschrittiteration*, die durch die Vorschrift

$$k_i^{(j+1)} = f\left(t + c_i h,\, x + h\sum_{l=1}^{i-1} a_{il} k_l^{(j+1)} + h\sum_{l=i}^{s} a_{il} k_l^{(j)}\right), \quad i = 1, \ldots, s \tag{5.2}$$

gegeben ist. Dies ist ein ähnlicher Trick, wie wir ihn in der Einführung in die Numerik beim Übergang vom Jacobi- zum Gauß-Seidel-Verfahren angewendet haben: Wir verwenden die bereits bekannten Werte $k_1^{j+1}, \ldots, k_{i-1}^{j+1}$ der $j+1$-ten Iteration bei der Berechnung von k_i^{j+1}. Im Allgemeinen konvergiert die Einzelschrittiteration (5.2) etwas schneller als die Gesamtschrittiteration (5.1).

Falls die Lipschitz-Konstante L des Vektorfeldes groß ist, werden bei diesen Fixpunktiterationen sehr kleine Zeitschrittweiten $h > 0$ benötigt, um die Kontraktionsbedingung $K = hL < 1$ sicher zu stellen. In diesem Falle können andere Verfahren vorteilhaft sein. So kann man das Problem $k = F(k)$ in ein geeignetes Nullstellenproblem umwandeln, z.B. mittels $0 = G(k) := k - F(k)$ (es gibt weitere, u.U. numerisch günstigere äquivalente Nullstellenprobleme, vgl. [3], Abschnitt 6.2.2). Wenn man nun die Ableitung DG ausrechnen kann, die sich aus der Ableitung $\partial/\partial x f(x)$ ergibt, so ist das Newton-Verfahren sehr gut geeignet, da man mit k aus dem vorhergehenden Schritt bzw. mit $k_i = f(t + c_i, x)$ einen guten Startwert für das (ja nur lokal konvergente) Newton-Verfahren besitzt.

Zusammenfassend führt dies auf den folgenden Algorithmus.

Algorithmus 5.2 (Lösung eines Anfangswertproblems mit implizitem Runge-Kutta-Verfahren)

Eingabe: Anfangsbedingung (t_0, x_0), Endzeit T, Schrittzahl N, Einschrittverfahren Φ

(1) Setze $h := (T - t_0)/N$, $\tilde{x}_0 = x_0$

(2) Für $i = 0, \ldots, N-1$:

(2a) Berechne $t_{i+1} = t_i + h$ und löse das nichtlineare Gleichungssystem $k = F(k)$

(2b) Berechne $\tilde{x}_{i+1} := \Phi(t_i, \tilde{x}_i, h) = \tilde{x}_i + h\sum_{j=1}^{s} b_j k_j$

Ausgabe: Werte der Gitterfunktion $\tilde{x}(t_i) = \tilde{x}_i$ in $t_0, \ldots, t_N$ □

Die Analyse impliziter Runge-Kutta-Verfahren ist im Vergleich zu den expliziten Verfahren komplizierter, da die Ableitungen von Φ (mit denen man sowohl die Konsistenz gemäß Satz 3.7 als auch die Lipschitz-Bedingung über die Ableitung nach x überprüfen kann) mit Hilfe des Satzes über implizite Funktionen berechnet werden müssen. Die Grundideen der Beweise sind aber gleich und die resultierenden Bedingungsgleichungen sind identisch zu denen in Satz 4.5, weswegen wir die technischen Details hier nicht vertiefen wollen.

Bemerkung 5.3 Für explizite Runge-Kutta-Verfahren haben wir in Lemma 4.3 gesehen, dass die Stufenanzahl s eine obere Schranke für die Konsistenzordnung p bildet, also immer $p \leq s$ gilt. Für implizite Verfahren ist die Schranke nicht ganz so strikt: Für ein s-stufiges implizites Runge-Kutta-Verfahren Φ mit Konsistenzordnung p gilt die Ungleichung $p \leq 2s$,

d.h. die Konsistenzordnung ist maximal zwei mal so groß wie die Stufenzahl. Zum Beweis dieser Aussage wenden wir das Verfahren wieder auf das Anfangswertproblem

$$\dot{x}(t) = x(t), \;\; x(0) = 1$$

mit exakter Lösung e^t an. Man kann nun zeigen, dass die numerische Lösung von der Form

$$\Phi(0, 1, h) = P(h)/Q(h)$$

für zwei Polynome $P, Q \in \mathcal{P}_s$ mit $Q \not\equiv 0$ ist (vgl. dazu Lemma ???). Falls nun $\Phi(0, 1, h) - e^h = O(h^{2s+2})$ gilt, so folgt auch $P(h) - Q(h)e^h = O(h^{2s+2})$. Mittels Induktion über s zeigt man dann, dass dies nur für $P \equiv Q \equiv 0$ gelten kann, was ein Widerspruch zu $Q \not\equiv 0$ ist. Also kann $\Phi(0, 1, h) - e^h = O(h^{2s+2})$ nicht gelten, weswegen im besten Fall $\Phi(0, 1, h) - e^h = O(h^{2s+1})$ sein kann, also $p \leq 2s$. □

Während es bei expliziten Runge-Kutta-Verfahren sehr schwierig ist, Verfahren für große p zu konstruieren, lässt sich die maximale Konsistenzordnung $p = 2s$ bei impliziten Verfahren relativ leicht realisieren. Wiederum auf Butcher geht nämlich die Familie der *Gauß-Verfahren* zurück, bei denen sich die Koeffizienten durch Nullstellen der Legendre-Polynome (ähnlich wie bei der Gauß-Quadratur) ermitteln lassen und die eine Familie von impliziten Verfahren mit $p = 2s$ bildet.

Kapitel 6

Steife Differentialgleichungen

Steife Differentialgleichungen sind eine Klasse von Differentialgleichungen, die mit expliziten Verfahren nur schwer zu lösen sind. Sie bilden die Hauptmotivation dafür, implizite Verfahren zu betrachten und zu verwenden. Leider ist es nicht ganz leicht, einer Differentialgleichung anzusehen, ob sie "steif" ist; es ist nicht einmal leicht, diese Eigenschaft formal zu definieren. Vielleicht ist die informelle Beschreibung "mit expliziten Verfahren schwer zu lösen" bereits die beste mögliche Definition. Wir wollen aber trotzdem versuchen, diese Eigenschaft etwas zu formalisieren und gewisse Kriterien herausarbeiten, an denen man erkennen kann, ob man es mit einer steifen DGL zu tun hat.

Wir wollen dazu zunächst den Begriff "schwer zu lösen" etwas genauer fassen. Aus Satz 2.7 wissen wir, dass für allgemeine Einschrittverfahren mit Konvergenzordnung $p > 0$ die Abschätzung der Form

$$\|\tilde{x}(t_i) - x(t_i)\| \leq CEh^p$$

für alle hinreichend kleinen $h > 0$ gilt, wobei $E > 0$ aus der Konsistenzbedingung stammt und

$$C = \frac{1}{L}(\exp(L(t_i - t_0)) - 1)$$

von der Konstanten L der Lipschitzbedingung sowie von der Größe des Zeitintervalls $T - t_0$ abhängt. Eine Differentialgleichung ist nun schwer zu lösen, wenn CE eine sehr große Konstante ist oder wenn die Abschätzung für $\|\tilde{x}(t_i) - x(t_i)\|$ nur für sehr kleine $h > 0$ gilt. Was "sehr groß" bzw. "sehr klein" in diesem Zusammenhang bedeutet, hängt im Wesentlichen davon ab, wieviel Zeit man in die Berechnung der Lösung investieren möchte und wie kleine Zeitschritte man noch zulassen möchte. Eine genaue Schranke kann man — ähnlich wie bei der Frage "wann ist ein Problem schlecht konditioniert?" — nicht angeben.

Sicherlich muss man damit rechnen, dass eine Differentialgleichung schwer zu lösen ist, wenn sie schlecht konditioniert ist. Steife Differentialgleichungen zeichnen sich nun dadurch aus, dass sie mit expliziten Verfahren schwer zu lösen sind, *obwohl* sie gut konditioniert sind. Dass dies tatsächlich passieren kann, wollen wir an einem bereits bekannten Beispiel illustrieren: Wir betrachten wieder die 1d DGL

$$\dot{x}(t) = \lambda x(t)$$

mit $\lambda \in \mathbb{R}$. Für diese Gleichung hatten wir gesehen, dass sie die Kondition

$$\kappa = e^{\lambda(t-t_0)}.$$

besitzt und deswegen für $t >> t_0$ und $\lambda < 0$ sehr gut konditioniert ist, da $\kappa \approx 0$ ist. Wir wollen die exakte Lösung $x(t; x_0) = e^{\lambda t} x_0$ dieser Gleichung für $\lambda << 0$ mit der numerischen Approximation durch das Euler-Verfahren vergleichen. Diese Approximation ist gegeben durch

$$\tilde{x}(t_{i+1}) = \tilde{x}(t_i) + h\lambda\tilde{x}(t_i) = (1 + h\lambda)\tilde{x}(t_i).$$

Durch Induktion sieht man leicht, dass die Euler-Lösung für $t_i = hi$ damit gerade durch

$$\tilde{x}(t_i) = (1 + h\lambda)^i x_0$$

gegeben ist. Für kleine $\lambda < 0$ konvergiert die exakte Lösung z.B. mit Anfangswert $x_0 = 1$ sehr schnell gegen 0. Damit die Euler-Lösung eine vernünftige Approximation darstellt, sollte diese also auch gegen Null streben. Damit dies passiert, muss $|1 + h\lambda| < 1$ sein, was für negative λ genau dann der Fall ist, wenn $|h\lambda| < 2$, also

$$h < 2/|\lambda|$$

ist. Z.B. für $\lambda = -10000$ müssen wir den Zeitschritt $h < 1/5000$ wählen, um überhaupt eine halbwegs sinnvolle Approximation zu erhalten und das, obwohl die Gleichung sehr gut konditioniert ist.

Zum Vergleich betrachten wir nun das implizite Euler-Verfahren, das durch

$$\tilde{x}(t_{i+1}) = \tilde{x}(t_i) + h\lambda\tilde{x}(t_{i+1}) \quad \Leftrightarrow \quad \tilde{x}(t_{i+1}) = \frac{\tilde{x}(t_i)}{1 - h\lambda}$$

gegeben ist. Die approximierte Lösung ist also

$$\tilde{x}(t_i) = \frac{1}{(1 - h\lambda)^i} x_0.$$

Hier strebt die Lösung genau dann gegen Null, wenn $|1/(1 - h\lambda)| < 1$ ist, also wenn $|1 - h\lambda| > 1$ ist. Da $\lambda < 0$ ist, ist diese Bedingung für sämtliche Zeitschritte $h > 0$ erfüllt, die Lösung konvergiert also für alle Zeitschritte gegen Null und stellt damit eine sinnvolle Approximation dar. Abbildung 6.1 zeigt die exakte Lösung sowie die numerischen Approximationen für $\lambda = -100$ für verschiedene Zeitschritte.

6.1 Stabilität

Für die 1d-Gleichung $\dot{x}(t) = \lambda x(t)$ können wir also sagen, dass sie steif ist, wenn $\lambda < 0$ und $|\lambda|$ groß ist. Wir wollen dieses Kriterium auf eine größere Klasse von Differentialgleichungen verallgemeinern.

Wir betrachten dazu die Klasse der *linearen zeitinvarianten* DGL, die gegeben ist durch

$$\dot{x}(t) = Ax(t), \tag{6.1}$$

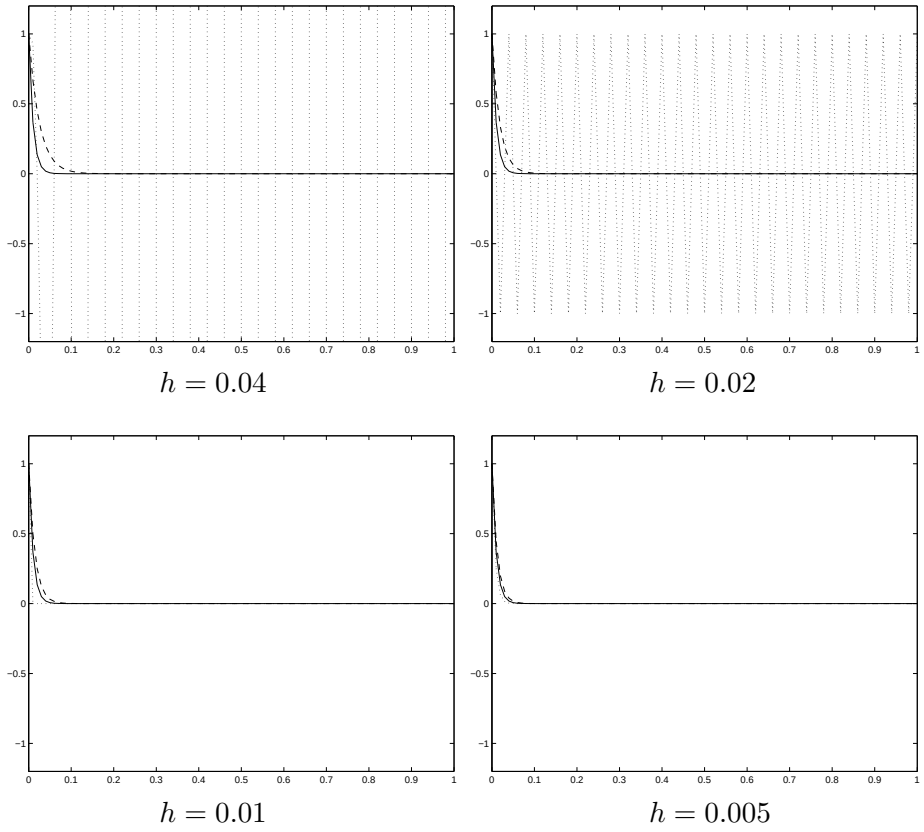

Abbildung 6.1: Exakte Lösung (durchgezogen), explizite Euler-Lösung (gepunktet) und implizite Euler-Lösung (gestrichelt) für $\dot{x}(t) = \lambda x(t)$, $x(0) = 1$, $\lambda = -100$

wobei $x(t) \in \mathbb{R}^n$ und $A \in \mathbb{R}^{n\times n}$ ist. Die Idee, diese Klasse von Differentialgleichungen zu betrachten, geht auf Germund Dahlquist[1] zurück. Für solche Gleichungen sind die durch ein Runge-Kutta-Verfahren erzeugten approximativen Lösungen stets von der Form

$$\tilde{x}(t_{i+1}) = \widetilde{A}\tilde{x}(t_i) \tag{6.2}$$

für ein $\widetilde{A} \in \mathbb{R}^{n\times n}$. Wir beschränken uns in diesem Abschnitt auf den Fall äquidistanter Zeitschritte $h_i = h$ und $t_0 = 0$, woraus sich $t_i = hi$ ergibt. Eine Gleichung der Form (6.2) wird *lineare zeitinvariante Differenzengleichung* genannt. Wir bezeichnen die Lösungen von (6.2) mit $\tilde{x}(0) = x_0$ mit $\tilde{x}(t; x_0)$, wobei $t \in \mathbb{R}$ ein Vielfaches von h ist. Offenbar gilt gerade $\tilde{x}(hi; x_0) = \widetilde{A}^i x_0$.

Für das explizite Euler-Verfahren gilt z.B. $\widetilde{A} = \mathrm{Id} + hA$, während für das implizite Euler-Verfahren $\widetilde{A} = (\mathrm{Id} - hA)^{-1}$ gilt, wobei $\mathrm{Id} \in \mathbb{R}^{n\times n}$ die Einheitsmatrix bezeichnet. Genauer beschreibt das folgende Lemma, wie A und $\widetilde{A}$ zusammenhängen.

Lemma 6.1 Für jedes s-stufige Runge-Kutta-Verfahren lässt sich die Matrix $\widetilde{A}$ in (6.2) als

$$\widetilde{A} = R(hA)$$

[1] schwedischer Mathematiker, 1925-2005

schreiben, wobei R eine von h unabhängige Funktion ist. Für explizite Runge-Kutta-Verfahren ist R ein Polynom vom Grad $\leq s$, für implizite Verfahren ist R eine rationale Funktion, d.h. eine Funktion der Form $R(z) = P(z)Q(z)^{-1}$, wobei P und Q wieder Polynome vom Grad $\leq s$ sind.

Beweis: Es seien a_{ij} und b_i die Koeffizienten des Verfahrens. Dann gilt für die Stufen k_i bei Anwendung auf (6.1) die Beziehung

$$hk_i = hAx + \sum_{j=1}^{s} a_{ij} hAhk_j$$

wobei wir beim expliziten Verfahren die Konvention $a_{ij} = 0$ für $j \geq i$ machen. Im expliziten Fall folgt per Induktion, dass jedes hk_i ein Polynom in hA vom Grad $\leq i$ ist und linear in x ist. Damit ist $\Phi(t, x, h) = x + \sum b_i hk_i$ ein Polynom vom Grad $\leq s$ in hA und linear in x, also gerade von der behaupteten Form.

Im impliziten Fall erhalten wir

$$\left(hk_i - \sum_{j=1}^{s} a_{ij} hAhk_j \right) = hAx$$

für $i = 1, \ldots, s$. Der $n \cdot s$-dimensionale Vektor $k = (k_1^T, \ldots, k_s^T)^T$ ist also gerade die Lösung eines $n \cdot s$-dimensionalen linearen Gleichungssystems, dessen Matrix affin linear von A und dessen rechte Seite linear von A und x abhängt. Durch Auflösen dieses Gleichungssystems sieht man (nach länglicher Rechnung, die wir hier nicht durchführen wollen), dass sich die k_i als

$$hk_i = \hat{P}_i(hA)Q(hA)^{-1}x$$

schreiben lassen, wobei die $\hat{P}_i$ und Q Polynome vom Grad $\leq s$ sind. Damit ist auch Φ wegen

$$\begin{aligned} \Phi(t, x, h) &= x + \sum b_i hk_i \\ &= x + \sum b_i \hat{P}_i(hA)Q(hA)^{-1}x \\ &= \left(Q(hA) + \sum b_i \hat{P}_i(hA) \right) Q(hA)^{-1}x \\ &= P(hA)Q(hA)^{-1}x \end{aligned}$$

von der behaupteten Form. □

Bemerkung 6.2 (i) Das Wichtige an der soeben bewiesenen Struktur ist, dass die Abbildung R Eigenwerte von hA auf Eigenwerte von $R(hA)$ abbildet. Mit anderen Worten ist $\lambda \in \mathbb{C}$ genau dann ein Eigenwert von hA, wenn $R(\lambda) \in \mathbb{C}$ ein Eigenwert von $R(hA)$ ist. Dies werden wir im Beweis von Lemma 6.6 beweisen.

(ii) Aus dem Gleichungssystem des obigen Beweises kann man eine explizite Formel für $R(hA)$ berechnen, die aber recht kompliziert ist. Da wir später allerdings nur betrachten werden, wie Eigenwerte unter der Abbildung R abgebildet werden, reicht es aus, die Funktion R für komplexwertige Argumente $z \in \mathbb{C}$ explizit zu kennen. Wenn wir die Koeffizienten

des Verfahrens mit $\mathcal{A} = (a_{ij})_{i,j=1,\dots,s}$ und $b = (b_1, \dots, b_s)^T$ bezeichnen, so kann man hierfür den expliziten Ausdruck

$$R(z) = 1 + zb^T(\mathrm{Id} - z\mathcal{A})^{-1}\mathbf{e}$$

mit $\mathbf{e} = (1, \dots, 1)^T \in \mathbb{R}^s$ berechnen. Für komplexe Argumente wird die Funktion $R : \mathbb{C} \to \mathbb{C}$ als *Stabilitätsfunktion* des Verfahrens bezeichnet.

Z.B. ergeben sich für das explizite Euler-Verfahren $R(z) = 1 + z$, für das implizite Euler-Verfahren $R(z) = (1 - z)^{-1}$ und für die implizite Trapezregel aus Kapitel 5 $R(z) = (1 + z/2)/(1 - z/2)$. □

Wie im obigen eindimensionalen Fall wollen wir speziell Lösungen betrachten, die gegen Null streben und untersuchen, für welche Zeitschritte die numerische Approximation dieses Verhalten widerspiegelt. Dazu verwenden wir die folgende Definition.

Definition 6.3 Eine Differentialgleichung (6.1) bzw. eine Differenzengleichung (6.2) heißt *(global) exponentiell stabil*, falls Konstanten $c, \sigma > 0$ existieren, so dass für alle Anfangswerte $x_0 \in \mathbb{R}^n$ die Ungleichung

$$\|x(t; x_0)\| \leq ce^{-\sigma t}\|x_0\| \text{ für alle } t \geq 0$$

bzw.

$$\|\tilde{x}(t; x_0)\| \leq ce^{-\sigma t}\|x_0\| \text{ für alle } t = ih \geq 0$$

gilt. □

Für die obigen Gleichungstypen (6.1) und (6.2) kann man zeigen, dass sie genau dann exponentiell stabil sind, wenn alle Lösungen gegen Null konvergieren. Die spezielle exponentielle Abschätzung ergibt sich dann aus der Linearität der Gleichungen.

In Analogie zum eindimensionalen Fall nennen wir eine exponentiell stabile Differentialgleichung der Form (6.1) steif, wenn für explizite Verfahren ein sehr kleiner Zeitschritt nötig ist, damit die durch das Verfahren erzeugte Differenzengleichung (6.2) ebenfalls exponentiell stabil ist.

Um nun zu sehen, wie man anhand der Matrix A die Steifheit erkennen kann und zu verstehen, warum implizite Verfahren hier Vorteile haben, brauchen wir ein geeignetes Kriterium für exponentielle Stabilität. Glücklicherweise muss man nicht alle Lösungen kennen, um zu entscheiden, ob exponentielle Stabilität vorliegt; man kann diese Eigenschaft anhand der Matrizen A bzw. $\widetilde{A}$ erkennen, wie der folgende Satz zeigt. Hierbei bezeichnet $\Re(z) = a$ den Realteil und $|z| = \sqrt{a^2 + b^2}$ den Betrag einer komplexen Zahl $z = a + ib \in \mathbb{C}$.

Satz 6.4 (i) Die Differentialgleichung (6.1) ist genau dann exponentiell stabil, wenn für alle Eigenwerte λ_i von A die Ungleichung $\Re(\lambda_i) < 0$ gilt.

(ii) Die Differenzengleichung (6.2) ist genau dann exponentiell stabil, wenn für alle Eigenwerte $\tilde{\lambda}_i$ von $\widetilde{A}$ die Ungleichung $|\tilde{\lambda}_i| < 1$ gilt.

Beweisskizze: Wir beweisen Teil (ii) unter der Annahme, dass $\widetilde{A}$ diagonalisierbar ist (der Beweis von (ii) im nicht-diagonalisierbaren Fall funktioniert genauso, ist aber technischer; der Beweis von (i) ist ähnlich, verlangt aber weitere Kenntnisse über die Lösungsstruktur von (6.1), auf die wir hier nicht eingehen können).

Falls $\widetilde{A}$ diagonalisierbar ist, so existiert eine Koordinatentransformationsmatrix $T \in \mathbb{R}^{n\times n}$, so dass

$$T^{-1}\widetilde{A}T = \widetilde{\Lambda} = \begin{pmatrix} \tilde{\lambda}_1 & & & \\ & \tilde{\lambda}_2 & & \\ & & \ddots & \\ & & & \tilde{\lambda}_n \end{pmatrix}$$

ist, wobei die $\tilde{\lambda}_i$ gerade die Eigenwerte von $\widetilde{A}$ sind. Für die Lösung $\tilde{x}(hi; x_0)$ gilt dann gerade

$$\begin{aligned} \tilde{x}(ih; x_0) &= \widetilde{A}^i x_0 \\ &= (T\widetilde{\Lambda}T^{-1})^i x_0 \\ &= T\widetilde{\Lambda}^i T^{-1} x_0. \end{aligned}$$

Sei nun $\alpha = \max_i |\tilde{\lambda}_i| < 1$. Wenn wir $y = (y_1, \ldots, y_n)^T = T^{-1}x_0$ setzen, so folgt

$$\widetilde{\Lambda}^i y = \begin{pmatrix} \tilde{\lambda}_1^i y_1 \\ \vdots \\ \tilde{\lambda}_n^i y_n \end{pmatrix}$$

und damit $\|\widetilde{\Lambda}^i y\| \leq \alpha^i \|y\|$. Mit $\sigma = -\ln(\alpha)/h > 0$ und $t = hi$ folgt

$$\|\widetilde{\Lambda}^i y\| \leq e^{-\sigma t}\|y\|$$

und damit

$$\|\tilde{x}(t; x_0)\| \leq \|T\|e^{-\sigma t}\|T^{-1}x_0\| \leq e^{-\sigma t}\|T\|\|T^{-1}\|\|x_0\| = ce^{-\sigma t}\|x_0\|$$

mit $c = \|T\|\|T^{-1}\|$.

Sei umgekehrt $|\tilde{\lambda}_j| \geq 1$ für ein j und sei x_0 ein zugehöriger Eigenvektor. Dann gilt

$$\|\tilde{x}(t; x_0)\| = \|\widetilde{A}^i x_0\| = |\tilde{\lambda}_j^i|\|x_0\| \geq \|x_0\|$$

für alle $t = ih > 0$, weswegen (6.2) nicht exponentiell stabil ist. □

Wir bezeichnen mit

$$\Sigma(A) = \{\lambda_i \,|\, \lambda_i \text{ ist Eigenwert von } A\}$$

die Menge aller Eigenwerte, das sogenannte *Spektrum* von A.

Für die Differentialgleichung muss damit gerade

$$\Sigma(A) \subset \mathbb{C}^- := \{z \in \mathbb{C} \,|\, \Re(z) < 0\}$$

gelten, damit exponentielle Stabilität vorliegt. Ein Eigenwert $\lambda_i \in \mathbb{C}^-$ wird dabei als *stabiler Eigenwert* bezeichnet. Analog muss für die numerische Approximation (6.2)

$$\Sigma(\widetilde{A}) \subset B_1(0) := \{z \in \mathbb{C} \,|\, |z| < 1\}$$

gelten, damit exponentielle Stabilität vorliegt.

6.2 Stabilitätsgebiet und A-Stabilität

Zu klären bleibt die Frage, welche Bedingung A aus (6.1) erfüllen muss, damit (6.2) für die Matrix $\widetilde{A} = R(hA)$ exponentiell stabil ist. Sicherlich hängt dies vom verwendeten Verfahren und vom Zeitschritt ab. Hierzu verwenden wir die folgende Definition. Beachte dabei, dass

$$\Sigma(A) \subset \mathbb{C}^- \quad \Leftrightarrow \quad \Sigma(hA) \subset \mathbb{C}^- \text{ für alle } h > 0$$

gilt, da λ_i genau dann ein Eigenwert von A ist, wenn $h\lambda_i$ ein Eigenwert von hA ist.

Definition 6.5 (i) Das *Stabilitätsgebiet* $\mathcal{S} \subset \mathbb{C}$ eines Runge-Kutta-Verfahrens mit Stabilitätsfunktion R ist definiert als die maximale Teilmenge der komplexen Zahlen, für die für alle $A \in \mathbb{R}^{n\times n}$ und alle $h > 0$ die Folgerung

$$\Sigma(hA) \subset \mathcal{S} \quad \Rightarrow \quad \Sigma(R(hA)) \subset B_1(0)$$

gilt. Mit anderen Worten ist $\mathcal{S}$ gerade die Menge von Eigenwerten λ_i, die hA aus (6.1) annehmen darf, damit (6.2) mit $\widetilde{A} = R(hA)$ exponentiell stabil ist.

(ii) Ein Runge-Kutta-Verfahren heißt *A-stabil*, falls

$$\mathbb{C}^- \subseteq \mathcal{S}$$

gilt bzw., äquivalent dazu, falls die Folgerung

$$\Sigma(hA) \subset \mathbb{C}^- \quad \Rightarrow \quad \Sigma(R(hA)) \subset B_1(0)$$

gilt. □

Die Interpretation von (i) ist wie folgt: Zur korrekten numerischen Approximation einer exponentiell stabilen Gleichung muss die Schrittweite $h > 0$ so gewählt werden, dass die Eigenwerte von hA in $\mathcal{S}$ liegen. Je besser $\mathcal{S}$ die Menge $\mathbb{C}^-$ ausschöpft, desto geringer sind die Anforderungen an die Schrittweite; im Falle der A-Stabilität gibt es überhaupt keine Einschränkungen der Schrittweite, die exponentielle Stabilität von (6.1) wird für alle Zeitschritte $h > 0$ von (6.2) "geerbt".

Das folgende Lemma zeigt, wie der Stabilitätsbereich $\mathcal{S}$ berechnet werden kann.

Lemma 6.6 Gegeben sei ein Runge-Kutta-Verfahren mit Stabilitätsfunktion R aus Bemerkung 6.2. Dann ist der Stabilitätsbereich gegeben durch

$$\mathcal{S} = \{z \in \mathbb{C} \,|\, |R(z)| < 1\}.$$

Beweis: Zum Beweis der Behauptung zeigen wir zunächst, dass für alle Matrizen $B \in \mathbb{R}^{n\times n}$ gilt: $\lambda_i \in \mathbb{C}$ ist genau dann ein Eigenwert von B, wenn $R(\lambda_i)$ ein Eigenwert von $R(B)$ ist.

Sei $C \in \mathbb{R}^{n\times n}$ eine beliebige Matrix mit Eigenwerten λ_i, $i = 1, \ldots, p \leq n$. Für ein Polynom

$$P(C) = \alpha_0 \mathrm{Id} + \alpha_1 C + \ldots + \ldots \alpha_s C^s$$

sind die Eigenwerte von $P(C)$ gerade die Eigenwerte $P(\lambda_i)$ von C, was man am einfachsten sieht, indem man P auf die Jordan-Normalform J von C anwendet. Ein Jordanblock J_i zum Eigenwert λ_i wird dabei auf eine obere Dreiecksmatrix mit $P(\lambda_i)$ in der Diagonalen abgebildet, die genau den einzigen Eigenwert $P(\lambda_i)$ besitzt (lediglich die Vielfachheiten können sich u.U. ändern). Hierbei sind Eigenvektoren von C wieder Eigenvektoren von $P(C)$.

Für die Inverse C^{-1} ist sind die Eigenwerte gerade $1/\lambda_i$ und für ein Produkt zweier Matrizen mit gleichen Eigenvektoren sind die Eigenwerte gerade die Produkte der Eigenwerte.

Also folgt, dass die Eigenwerte von $R(B) = P(B)Q(B)^{-1}$ gerade die Produkte der Eigenwerte $P(\lambda_i)$ und $Q(\lambda_i)^{-1}$, also $P(\lambda_i)Q(\lambda_i)^{-1}$ sind.

Weil R also Eigenwerte von hA auf Eigenwerte von $R(hA)$ abbildet, gilt $\Sigma(R(hA)) = R(\Sigma(hA))$ und damit

$$\begin{aligned} \Sigma(R(hA)) \subset B_1(0) &\Leftrightarrow R(\Sigma(hA)) \subset B_1(0) \\ &\Leftrightarrow |R(\lambda_i)| < 1 \text{ für alle Eigenwerte } \lambda_i \text{ von } hA. \end{aligned}$$

Für alle Matrizen hA mit $\Sigma(hA) \subset \{z \in \mathbb{C} \,|\, |R(z)| < 1\}$ gilt also $\Sigma(R(hA)) \subset B_1(0)$, woraus wegen der Maximalität von $\mathcal{S}$ die Inklusion $\{z \in \mathbb{C} \,|\, |R(z)| < 1\} \subseteq \mathcal{S}$ folgt. Andererseits gilt für jedes $z \in \mathbb{C}$ mit $|R(z)| \geq 1$ und die 1×1-Matrix $A = (z)$ sowie $h = 1$, dass $\{z\} = \Sigma(hA) \not\subset \mathcal{S}$. Also folgt die behauptete Gleichheit. □

Mit Hilfe dieses Satzes können wir die Stabilitätsbereiche nun bestimmen. Für das explizite Euler-Verfahren mit $R(z) = 1 + z$ gilt

$$|R(z)| < 1 \;\Leftrightarrow\; |1 + z| < 1$$

also ist $\mathcal{S} = \{z \in \mathbb{C} || 1 + z| < 1\} = B_1(-1)$, also gerade der offene Ball mit Radius 1 um -1. Der Zeitschritt muss also so klein gewählt werden, dass für alle Eigenwerte die Bedingung $h\lambda_i \in B_1(-1)$ erfüllt ist.

Abbildung 6.2 zeigt die Stabilitätsbereiche einiger expliziter Runge-Kutta-Verfahren mit den Ordnungen $p = 1, \ldots, 4$. Man sieht, dass der Stabilitätsbereich $\mathcal{S}$ für wachsende Konsistenz größer wird, allerdings die Menge $\mathbb{C}^-$ bei weitem nicht ausschöpft. Im Falle betragsmäßig großer Eigenwerte λ_i erhält man für all diese Verfahren starke Einschränkungen bei der Wahl der Zeitschritte.

Beachte, dass bei mehrdimensionalen Problemen nicht unbedingt der *Realteil* eines Eigenwertes betragsmäßig groß werden muss, damit der Betrag des Eigenwertes groß wird. Das folgende Beispiel illustriert dies.

Betrachte die zweidimensionale lineare DGL

$$\dot{x}(t) = \begin{pmatrix} -1 & \alpha \\ -\alpha & -1 \end{pmatrix} x(t). \tag{6.3}$$

Die zugehörige Matrix besitzt die Eigenwerte $\lambda_{1/2} = -1 \pm i\alpha$. Hier haben Realteil und Imaginärteil eine geometrische Bedeutung für die Lösung: Der Realteil gibt an, wie schnell die Lösung gegen Null konvergiert (diese Größe ist hier konstant gleich -1), während der Imaginärteil angibt, wie schnell die Lösung sich dabei dreht. Abbildung (6.3) zeigt die

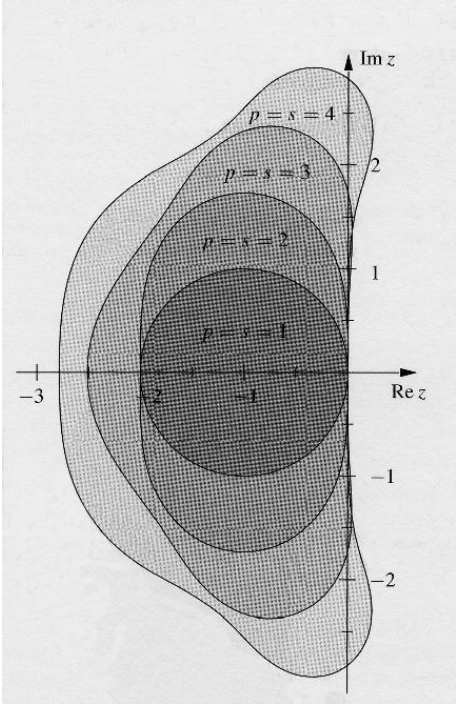

Abbildung 6.2: Stabilitätsbereiche expliziter Runge-Kutta-Verfahren, entnommen aus [3]

exakten Lösungen für $\alpha = 0,\ 1,\ 10$ sowie die zugehörigen Euler-Lösungen mit $h = 0.02$. Man sieht: Wenn der Eigenwert betragsmäßig größer wird, weil der Imaginärteil wächst, dann wird die Euler-Lösung instabil.

Wie verhalten sich nun implizite Verfahren? Für das implizite Euler-Verfahren z.B. berechnet man

$$|R(z)| < 1 \ \Leftrightarrow\ 1/|1-z| < 1 \ \Leftrightarrow\ |1-z| > 1 \ \Leftarrow\ \Re(z) < 0.$$

Folglich gilt $\mathbb{C}^- \subset \mathcal{S}$, das Verfahren ist also A-stabil.

Viele implizite Verfahren sind A-stabil, und von denjenigen, die es nicht sind, besitzen viele einen Stabilitätsbereich, der deutlich größer ist als bei expliziten Verfahren. Eine Übersicht über die Stabilitätsbereiche einiger impliziter Verfahren findet sich z.B. im Abschnitt IV.3 des Buchs [8].

Eine lineare DGL (6.1) kann auch dann steif sein, wenn sie nicht exponentiell stabil ist, aber zumindest einige stabile Eigenwerte besitzt, also solche mit negativem Realteil. Die Lösungskomponenten in den zugehörigen Eigenräumen (man nennt deren Vereinigung *stabilen Unterraum*) verhalten sich dann wie bei einer exponentiell stabilen Gleichung. Folglich treten bei betragsmäßig großen stabilen Eigenwerten exakt die gleichen Probleme auf, auch wenn die Gleichung insgesamt nicht exponentiell stabil ist. Dies führt uns auf die folgende Charakterisierung.

Bemerkung 6.7 Eine lineare zeitinvariante Differentialgleichung ist steif, falls die zugehörige Matrix A betragsmäßig große stabile Eigenwerte besitzt. □

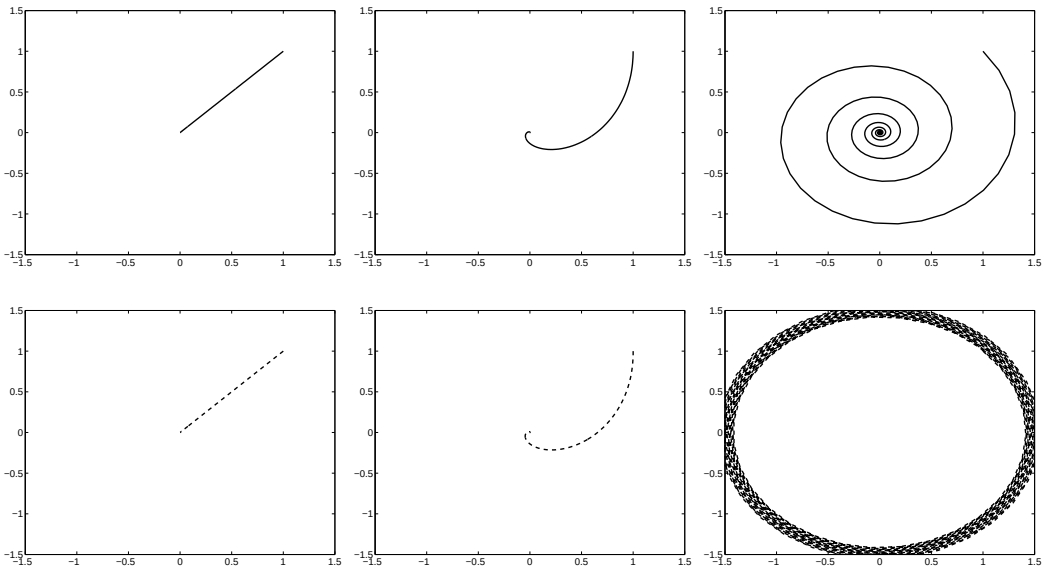

Abbildung 6.3: Exakte und Euler-Lösungen von (6.3) mit $\alpha = 0,\ 1,\ 10$, $h = 0.02$

Für nichtlineare DGL $\dot{x}(t) = f(t, x(t))$ gibt es viele weitere Phänomene, die zur Steifheit führen; meistens kann man diese nicht so einfach am Vektorfeld f ablesen. Im einfachsten Fall ist f autonom und besitzt ein Gleichgewicht x^*, in dem f stetig differenzierbar ist. In diesem Fall kann man $A = Df(x^*)$ betrachten; wenn diese Matrix betragsmäßig große stabile Eigenwerte besitzt, so wird auch die nichtlineare DGL typischerweise steif sein. Steifheit kann aber auch auftreten, wenn kein Gleichgewicht vorliegt, z.B. wenn die DGL eine exponentiell stabile periodische Lösung besitzt (also eine periodische Lösung, gegen die alle Lösungen exponentiell konvergieren, zumindest für nahe liegende Anfangswerte). In diesem Fall kann die Gleichung steif sein, wenn die anderen Lösungen sehr schnell gegen die periodische Lösung streben (dies entspricht betragsmäßig großen negativen Realteilen im linearen Fall) oder wenn sich die periodische Lösung sehr schnell bewegt (dies entspricht den großen Imaginärteilen.)

6.3 Weitere Stabilitätsbegriffe

Der Begriff der A-Stabilität wurde von G. Dahlquist in den 1960er Jahren eingeführt. A-Stabilität ist nützlich bei der Lösung steifer Differentialgleichungen, ist aber für sich genommen weder eine positive noch eine negative Eigenschaft: Zwar ist es zur numerischen Lösung steifer DGL vorteilhaft, wenn die exponentielle Stabilität von der numerischen Approximation geerbt wird. Allerdings kann es andererseits auch passieren, dass die numerische Approximation exponentiell stabil ist, obwohl die exakte Gleichung diese Eigenschaft *nicht* besitzt, was zu falschen Rückschlüssen auf das Verhalten der exakten Lösungen führen kann.

Eine stärkere Eigenschaft ist die *Erhaltung der Isometrie*, die verlangt, dass $\mathcal{S} = \mathbb{C}^-$ ist, d.h. für alle Zeitschritte $h > 0$ ist die numerische Approximation *genau dann* exponentiell

stabil, wenn die exakte Gleichung exponentiell stabil ist. Diese Eigenschaft besitzen z.B. die bereits erwähnten Gauß-Verfahren. Ein anderes Verfahren mit dieser Eigenschaft ist die implizite Mittelpunktregel, vgl. Abschnitt 5.1, für die wir dieses nachweisen wollen: Ausgeschrieben ist die zugehörige Iterationsvorschrift gegeben durch

$$\tilde{x}(t_{i+1}) = \tilde{x}(t_i) + h_i f\left(t_i + \frac{h_i}{2}, \frac{1}{2}(\tilde{x}(t_i) + \tilde{x}(t_{i+1}))\right).$$

Angewendet auf die lineare Differentialgleichung $\dot{x}(t) = Ax(t)$ ergibt sich

$$\begin{aligned}
& \tilde{x}(t_{i+1}) &&= \tilde{x}(t_i) + \frac{h_i}{2}A\tilde{x}(t_i) + \frac{h_i}{2}A\tilde{x}(t_{i+1}) \\
\Leftrightarrow\quad & \tilde{x}(t_{i+1}) - \frac{h_i}{2}A\tilde{x}(t_{i+1}) &&= \tilde{x}(t_i) + \frac{h_i}{2}A(\tilde{x}(t_i) \\
\Leftrightarrow\quad & \left(\mathrm{Id} - \frac{h_i}{2}A\right)\tilde{x}(t_{i+1}) &&= \left(\mathrm{Id} + \frac{h_i}{2}A\right)\tilde{x}(t_i) \\
\Leftrightarrow\quad & \tilde{x}(t_{i+1}) &&= \left(\mathrm{Id} - \frac{1}{2}h_iA\right)^{-1}\left(\mathrm{Id} + \frac{1}{2}h_iA\right)\tilde{x}(t_i).
\end{aligned}$$

Die Stabilitätsfunktion ist daher gegeben durch

$$R(z) = \frac{1 + z/2}{1 - z/2}.$$

Wir wollen nun nachweisen, dass dieses Verfahren die Isometrie erhält. Dazu müssen wir die Äquivalenz $\Re(z) < 0 \Leftrightarrow |R(z)| < 1$ zeigen, wozu wir alternativ auch

$$\Re(z) < 0 \Leftrightarrow |R(z)|^2 < 1$$

nachprüfen können. Für $z = a + ib$ gilt wegen $|x + iy|^2 = x^2 + y^2$ nun

$$|R(z)|^2 = \frac{|1 + z/2|^2}{|1 - z/2|^2} = \frac{(1 + a/2)^2 + (b/2)^2}{(1 - a/2)^2 + (b/2)^2}.$$

Dieser Ausdruck ist nun genau dann < 1, wenn $(1 + a/2)^2 < (1 - a/2)^2$ gilt. Dies ist aber genau dann der Fall, wenn $a < 0$ gilt, womit die Erhaltung der Isometrie folgt.

Unglücklicherweise ist es aber so, dass diese Eigenschaft stets gemeinsam mit einer anderen — unerwünschten — Eigenschaft auftritt. Um diese zu illustrieren, betrachten wir die implizite Mittelpunktregel angewendet auf die Gleichung $\dot{x}(t) = \lambda x(t)$ mit $\lambda = -1000$ und Schrittweite $h = 0.01$.

Zwar ist die Lösung asymptotisch stabil, allerdings konvergiert sie nicht — wie die exakte Lösung — monoton sondern oszillierend gegen 0. Zudem konvergiert die numerische Approximation um so langsamer gegen 0, je größer $|\lambda|$ wird. Und dass, obwohl die exakte Lösung $e^{\lambda t}x_0$ ja für negative λ um so schneller gegen 0 konvergiert, je größer $|\lambda|$ ist. Dies kann man auch an der Stabilitätsfunktion sehen, die für betragsmäßig große negative λ Werte $R(\lambda) \approx 1$ und damit sehr langsame Konvergenz gegen 0 liefert. Die Approximation zeigt also das richtige Konvergenzverhalten, hat ansonsten aber nicht viel mit der exakten Lösung zu tun.

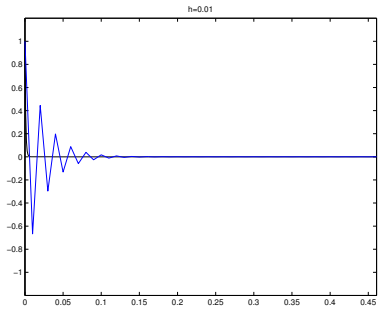

Abbildung 6.4: Oszillationen bei der impliziten Mittelpunktregel

Dies ist kein Zufall, denn jedes isometrieerhaltende Verfahren besitzt diese Eigenschaft. Der Grund liegt darin, dass für rationale Funktionen der Grenzwert

$$\lim_{k\to\infty} R(z_k)$$

— sofern er existiert — für alle komplexen Folgen $(z_k)_{k\in\mathbb{N}}$ mit $|z_k| \to \infty$ identisch ist, unabhängig von der Wahl der Folge z_k. Für $z_k = ib_k$ gilt aber nun $|R(z_k)| = 1$ (weil $|R|$ in der rechten komplexen Halbebene Werte > 1 und in der linken Halbebene Werte < 1 annimmt, muss aus Stetigkeitsgründen für rein imaginäre Zahlen $|R(z_k)| = 1$ gelten). Also gilt für $b_k \to \infty$ die Konvergenz $\lim_{k\to\infty} |R(z_k)| = \lim_{k\to\infty} |R(ib_k)| = 1$ und damit auch für alle anderen Folgen. Insbesondere gilt also $\lim_{a_k\to-\infty} |R(a_k + ib)| = 1$ für alle $b \in \mathbb{R}$ und damit $|R(a + ib)| \approx 1$ für betragsmäßig große negative a. Dies erklärt die langsame Konvergenz der isometrierhaltenden Verfahren bei sehr schnell konvergierenden Differentialgleichungen.

Was man stattdessen zur guten Approximation der Lösung haben möchte, ist die Konvergenz $R(a_k + ib) \to 0$ für $a_k \to -\infty$. Dies würde garantieren, dass mit der exakten auch die numerische Lösung immer schneller gegen 0 konvergiert.

Definition 6.8 Ein Runge-Kutta-Verfahren heißt L-stabil, wenn es A-stabil ist und zudem

$$\lim_{k\to\infty} R(z_k) = 0$$

gilt für alle komplexen Folgen $(z_k)_{k\in\mathbb{N}}$ mit $|z_k| \to \infty$. □

Wir schreiben diese Bedingung auch kurz als $R(\infty) = 0$. Beachte, dass $R(\infty)$ wohldefiniert ist, da der Grenzwert $\lim_{k\to\infty} R(z_k)$ für $|z_k| \to \infty$ nicht von der Wahl der z_k abhängt. Für diese Gleichung kann man eine hinreichende Bedingung an die Koeffizienten des Runge-Kutta-Verfahrens herleiten.

Satz 6.9 Wenn die Koeffizientenmatrix $\mathcal{A}$ eines impliziten Runge-Kutta-Verfahrens invertierbar ist und eine der beiden Bedingungen

$$a_{sj} = b_j \text{ für } j = 1, \ldots, s \quad \text{oder} \quad a_{i1} = b_1 \text{ für } i = 1, \ldots, s$$

gelten, so gilt $R(\infty) = 0$. Falls das Verfahren zusätzlich A-stabil ist, so ist es L-stabil.

Beweis: Da die Stabilitätsfunktion durch

$$R(z) = 1 + zb^T(\mathrm{Id} - z\mathcal{A})^{-1}\mathbf{e} = 1 + b^T\left(\frac{1}{z}\mathrm{Id} - \mathcal{A}\right)^{-1}\mathbf{e}$$

gegeben ist, folgt

$$R(\infty) = 1 - b^T\mathcal{A}^{-1}\mathbf{e}.$$

Im Fall der ersten Bedingung gilt $\mathcal{A}^T e_s = b$, wobei $e_s = (0, \ldots, 0, 1)^T \in \mathbb{R}^s$. Damit folgt $e_s^T\mathcal{A} = b^T$, folglich $e_s^T = b^T\mathcal{A}^{-1}$ und damit

$$R(\infty) = 1 - e_s^T\mathbf{e} = 1 - 1 = 0.$$

Im Fall der zweiten Bedingung gilt $\mathcal{A}e_1 = \mathbf{e}b_1$ und damit $\mathcal{A}^{-1}\mathbf{e} = (1/b_1, 0, \ldots, 0)^T$. Damit folgt

$$R(\infty) = 1 - b_1/b_1 = 1 - 1 = 0.$$

□

L-Stabilität spielt eine wichtige Rolle bei der Lösung sogenannter Differential-Algebraischer Gleichungen sowie bei singulär gestörten Problemen, da bei diesen Problemen typischerweise Eigenwerte mit betragsmäßig sehr großen negativen Realteilen auftreten. Ein Beispiel für ein L-stabiles implizites Runge-Kutta-Verfahren ist das Radau IIA-Verfahren (mit Ordnung $p = 5$) mit dem Butcher-Tableau

$$\begin{array}{c|ccc} \frac{4-\sqrt{6}}{10} & \frac{88-7\sqrt{6}}{360} & \frac{296-169\sqrt{6}}{1800} & \frac{-2+3\sqrt{6}}{225} \\ \frac{4+\sqrt{6}}{10} & \frac{296+169\sqrt{6}}{1800} & \frac{88+7\sqrt{6}}{360} & \frac{-2-3\sqrt{6}}{225} \\ 1 & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \\ \hline & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \end{array}$$

Offenbar ist hier gerade die erste Bedingung von Satz 6.9 erfüllt.

Das Problem mit der A-Stabilität ist, dass es viele numerische Verfahren gibt, die nicht A-stabil sind, für Eigenwerte und Schrittweiten in "sinnvollen" Bereichen aber trotzdem gute Lösungen liefern. Es ist daher zu einschränkend, prinzipiell A-Stabilität zu fordern, wenn man es mit steifen Differentialgleichungen zu tun hat. Nichtsdestotrotz ist es erstrebenswert, Stabilität für Eigenwerte mit beliebig kleinen Realteilen zu haben (also ein unbeschränktes Stabilitätsgebiet), allerdings nicht für beliebige Kombinationen von Real- und Imaginärteil.

Eine Stabilitätsbedingung, die dies mathematisch präzise definiert, ist die folgende $A(\alpha)$-Stabilität.

Definition 6.10 Ein Runge-Kutta Verfahren heißt $A(\alpha)$-stabil für ein $\alpha > 0$, falls der Sektor

$$S_\alpha := \{z \in \mathbb{C} \,|\, z \neq 0 \text{ und } |\arg(-z)| < \alpha\}$$

im Stabilitätsgebiet S enthalten ist. □

Ein Beispiel für eine Methode, die $A(\alpha)$-stabil aber nicht A-stabil ist ist die sogenannte $(0, 3)$-Padé-Approximation, mit $\alpha = 88.23°$. Für Details siehe [8, Abschnitt IV.3].

6.4 Nichtlineare A-Stabilität

Wie am Ende von Abschnitt 6.2 bereits erwähnt, gelten die bisher gemachten Stabilitätsaussagen auch für autonome nichtlineare Differentialgeichungen $\dot{x}(t) = f(x(t))$ in der Nähe von Gleichgewichten. Allerdings lässt sich mit der linearen Theorie basierend auf Jacobi-Matrizen und Eigenwerten prinzipiell keine Aussage über das Verhalten weit weg von den Gleichgewichten machen.

Um eine Eigenschaft wie die A-Stabilität nicht nur in der Nähe von Gleichgewichten nachzuweisen (also die Tatsache, dass die Approximationen für alle Schrittweiten asymptotisch stabil sind), benötigt man andere Methoden. Eine davon sind die sogenannten Lyapunov-Funktionen.

Definition 6.11 Eine stetig differenzierbare Funktion $V : \mathbb{R}^n \to \mathbb{R}$ heißt Lyapunov-Funktion für eine autonome gewöhnliche Differentialgleichung $\dot{x}(t) = f(x(t))$ an einem Gleichgewicht x^*, falls die folgenden Bedingungen gelten.

(a) $V(x^*) = 0$ und $V(x) > 0$ für alle $x \neq x^*$

(b) $V(x_n) \to \infty$ für alle Folgen $(x_n)_{n\in\mathbb{N}}$ mit $\|x_n\| \to \infty$ für $n \to \infty$

(c) $DV(x)f(x) < 0$ für alle $x \neq x^*$.

□

Satz 6.12 Betrachte eine autonome gewöhnliche Differentialgleichung $\dot{x}(t) = f(x(t))$ auf $D = \mathbb{R}^n$ mit einem Gleichgewicht x^*. Falls eine Lyapunov Funktion V existiert, so konvergieren alle Lösungen $x(t)$ der Differentialgleichung für $t \to \infty$ gegen x^*.

Beweisidee: Aus (c) folgt mit der Kettenregel die Ungleichung

$$\frac{d}{dt}V(x(t)) = DV(x(t))\dot{x}(t) = DV(x(t))f(x(t)) < 0,$$

falls $x(t) \neq x^*$. Daraus folgt, dass $t \mapsto V(x(t))$ streng monoton fällt, so lange die Lösung nicht bereits im Gleichgewicht x^* ist. Weil $V(x(t))$ wegen (a) zudem nach unten beschränkt ist, konvergiert $V(x(t))$ für $t \to \infty$ gegen einen Wert V_∞. Daraus folgt wiederum, dass die Ableitung $\frac{d}{dt}V(x(t))$ gegen 0 konvergieren muss. Dies kann wegen (c) nur passieren, wenn $x(t) \to x^*$ oder wenn $\|x(t)\| \to \infty$ konvergiert. Wegen (b) und der Beschränktheit von $V(x(t))$ ist aber nur ersteres möglich. □

Bemerkung 6.13 Tatsächlich folgt aus der Existenz einer Lyapunov Funktion mehr als nur die Konvergenz, nämlich die sogenannte *globale asymptotische Stabilität*. Neben der Konvergenz umfasst diese Eigenschaft auch die Tatsache, dass Lösungen die in der Nahe von x^* starten für alle positiven Zeiten in der Nähe von x^* bleiben. □

Die Idee einer nichtlinearen Verallgemeinerung der A-Stabilität liegt nun darin nachzuweisen, dass eine Lyapunov Funktion der Differentialgleichung für beliebige Schrittweiten auch eine Lyapunov Funktion für die numerische Approximation ist. Dazu genügt es, das Gegenstück zu Bedingung (c) aus Definition 6.11 nachzuweisen, nämlich

(c') $V(\Phi(x,h)) < V(x)$ für alle $x \neq x^*$.

Es ist nun leider zu optimistisch anzunehmen, dass es ein Verfahren gibt, mit dem (c') für alle $h > 0$, alle Lyapunov Funktionen und alle Differentialgleichungen gilt. Man kann aber beweisen, dass (c') für alle Schrittweiten $h > 0$ gilt

- für das implizite Euler-Verfahren, wenn V eine konvexe Funktion ist, wenn also für alle $x_1, x_2 \in \mathbb{R}^n$ und alle $\lambda \in [0,1]$ gilt

  $$V(\lambda x_1 + (1-\lambda)x_2) \leq \lambda V(x_1) + (1-\lambda)V(x_2),$$

  siehe [9, Theorem 4.17]

- für die implizite Mittelpunktregel, wenn V eine positiv definite quadratische Funktion ist, wenn also eine positiv definite Matrix $P \in \mathbb{R}^{n\times n}$ gibt, so dass V von der Form

  $$V(x) = x^T P x$$

  ist, siehe [13, Satz 5.15].

Da jede exponentiall stabile lineare zeitinvariante Differentialgleichung $\dot{x}(t) = Ax(t)$ eine quadratische Lyapunov Funktion besitzt und da eine solche stets konvex ist, bilden beide Aussagen eine echte Verallgemeinerung der linearen A-Stabilität.

Die Aussage für das implizite Euler-Verfahren folgt mit einigen (relativ einfachen) Argumenten aus der konvexen Analysis, die Aussage über die implizite Mittelpunktregel mit Hilfe einer geeigneten Taylor-Entwicklung unter Ausnutzung der Tatsache, dass die zweite Ableitung der quadratischen Funktion V konstant ist.

Kapitel 7

Schrittweitensteuerung

Nach den eher theoretischen Überlegungen des letzten Kapitels wollen wir uns jetzt wieder algorithmischen Aspekten widmen. Bisher sind wir davon ausgegangen, dass die Schrittweiten h_i gegeben sind, meistens haben wir sie als konstant $h_i \equiv h$ angenommen. In diesem Kapitel wollen wir uns überlegen, wie man die Schrittweiten automatisch so steuern kann, so dass dort, wo es nötig ist, kleine Schrittweiten gewählt werden, damit eine gewünschte Genauigkeit eingehalten wird und dort, wo es ohne Genauigkeitsverlust möglich ist, große Schrittweiten erlaubt werden, die eine schnellere Rechnung ermöglichen. Wir nehmen dabei durchgehend an, dass das Vektorfeld der betrachteten DGL hinreichend oft differenzierbar ist, so dass die Konsistenzordnungen der betrachteten Verfahren tatsächlich realisiert werden.

7.1 Fehlerschätzung

Zur Entscheidung darüber, ob die Schrittweite groß oder klein gewählt werden soll, ist es nötig, den Fehler zu kennen, den wir im aktuellen Schritt machen. Wir wollen uns zuerst überlegen, welcher Fehler hierfür wichtig ist. Hierbei müssen wir zunächst überlegen, wie wir die Schrittweite steuern wollen. Wie in der numerischen Praxis üblich wollen wir uns hier darauf beschränken, zur Zeit t_i eine gute Schrittweite h_i für den Schritt von t_i nach $t_{i+1} = t_i + h_i$ zu bestimmen und dabei auch einen "Schrittweitenvorschlag" h_{i+1} für den nächsten Schritt zu machen. Wir wollen aber nicht zum Zeitpunkt t_i die Schrittweiten in vorhergehenden Schritten t_j für $j < i$ nachträglich korrigieren, da die dadurch anfallenden Neuberechnungen algorithmisch sehr ineffizient wären.

Um ein gutes h_i zu bestimmen, müssen wir den Fehleranteil kennen, der durch den Schritt von t_i nach t_{i+1} hervorgerufen wird. Dieser Fehleranteil wird *lokaler Fehler* genannt. Wir haben in der Konvergenzanalyse in Abschnitt 2.3 verwendet, dass sich der Fehler zur Zeit t_{i+1} mittels

$$\begin{aligned} \|\tilde{x}(t_{i+1}) - x(t_{i+1})\| \quad &\leq \quad \|\Phi(t_i, \tilde{x}(t_i), h_i) - \Phi(t_i, x(t_i), h_i)\| \\ &+ \quad \|\Phi(t_i, x(t_i), h_i) - x(t_{i+1}; t_i, x(t_i))\| \end{aligned}$$

zerlegen lässt. Diese Zerlegung war für unsere theoretischen Überlegungen nützlich, hier ist sie nicht so günstig, da wir den in diesem Schritt hinzukommenden Fehleranteil

$$\|\Phi(t_i, x(t_i), h_i) - x(t_{i+1}; t_i, x(t_i))\|$$

nicht berechnen können, da wir $x(t_i)$ nicht kennen. Statt also in der Dreiecksungleichung den Term $\Phi(t_i, x(t_i), h_i)$ einzuschieben, schieben wir den Term $x(t_{i+1}; t_i, \tilde{x}(t_i))$ und erhalten so

$$\begin{aligned}\|\tilde{x}(t_{i+1}) - x(t_{i+1})\| \quad &\leq \quad \|\Phi(t_i, \tilde{x}(t_i), h_i) - x(t_{i+1}, t_i, \tilde{x}(t_i))\| \\ &+ \ \|x(t_{i+1}, t_i, \tilde{x}(t_i)) - x(t_{i+1}, t_i, x(t_i))\|\end{aligned}$$

Der zweite Fehlerterm hängt hierbei im Wesentlichen von dem bis zum Zeitpunkt t_i gemachten Fehler ab, den wir nur durch Änderung der Zeitschritte h_j für $j < i$ beeinflussen können, was wir gerade nicht machen wollen. Der Fehlerterm, den wir mit der Wahl von h_i wirklich beeinflussen können, ist der erste.

Die Idee der Schrittweitensteuerung (man sagt auch "adaptive Wahl der Schrittweite") liegt nun darin, h_i so groß zu wählen, dass die Fehlerbedingung

$$\|\Phi(t_i, \tilde{x}(t_i), h_i) - x(t_{i+1}, t_i, \tilde{x}(t_i))\| \leq tol$$

für eine vorgegebene Größe $tol > 0$ gerade eingehalten wird. Dies ist natürlich so nicht möglich, da wir dafür die exakte Lösung $x(t_{i+1}, \tilde{x}(t_i), t_i)$ kennen müssten. Um dieses Problem zu lösen, verwendet man einen sogenannten *Fehlerschätzer*, der wie folgt definiert ist.

Definition 7.1 Eine numerisch berechenbare Größe $\bar{\varepsilon}$ heißt *Fehlerschätzer* für den tatsächlichen Fehler ε eines numerischen Verfahrens, falls von $\bar{\varepsilon}$ und ε unabhängige Konstanten $\kappa_1, \kappa_2 > 0$ existieren, so dass die Abschätzung

$$\kappa_1 \varepsilon \leq \bar{\varepsilon} \leq \kappa_2 \varepsilon$$

gilt. □

Wie können wir nun für unsere Einschrittverfahren einen solchen Fehlerschätzer bekommen? Die Idee besteht darin, den Schritt von t nach $t_{i+1} = t_i + h_i$ mit zwei Verfahren $\widehat{\Phi}$ und Φ verschiedener Konsistenzordnung $\hat{p}$ und p zu berechnen. Für

$$\hat{\eta}_i := \widehat{\Phi}(t_i, \tilde{x}(t_i), h_i) - x(t_{i+1}, t_i, \tilde{x}(t_i)) \quad \text{und} \quad \eta_i := \Phi(t_i, \tilde{x}(t_i), h_i) - x(t_{i+1}, t_i, \tilde{x}(t_i))$$

gilt damit

$$\hat{\varepsilon}_i := \|\hat{\eta}_i\| \leq \widehat{E} h_i^{\hat{p}+1} \quad \text{und} \quad \varepsilon_i := \|\eta_i\| \leq E h_i^{p+1}. \tag{7.1}$$

Wir nehmen hierbei an, dass $p \geq \hat{p} + 1$ gilt und dass $\hat{p}$ die maximale (oder echte) Konsistenzordnung von $\widehat{\Phi}$ ist. Damit ist Φ das genauere Verfahren, weswegen für alle hinreichend kleinen $h_i > 0$ die Ungleichung $\varepsilon_i < \hat{\varepsilon}_i$ bzw.

$$\theta = \frac{\varepsilon_i}{\hat{\varepsilon}_i} < 1 \tag{7.2}$$

gilt, da $\theta \to 0$ strebt, wenn $h_i \to 0$ geht.

Wir definieren den Fehlerschätzer nun als

$$\bar{\varepsilon} := \|\bar{\eta}\| \quad \text{mit} \quad \bar{\eta} = \widehat{\Phi}(t_i, \tilde{x}(t_i), h_i) - \Phi(t_i, \tilde{x}(t_i), h_i). \tag{7.3}$$

Der folgende Satz zeigt, dass diese Größe tatsächlich ein Fehlerschätzer im Sinne von Definition 7.1 ist.

Satz 7.2 Betrachte zwei Einschrittverfahren $\widehat{\Phi}$ und Φ mit Konsistenzordnungen $\hat{p}$ und p mit $p \geq \hat{p} + 1$. Dann ist die Größe $\bar{\varepsilon}$ aus (7.3) für alle hinreichend kleinen Schrittweiten $h_i > 0$ ein Fehlerschätzer für $\hat{\varepsilon}_i$ aus (7.1).

Beweis: Wir wählen h_i so klein, dass die Abschätzung (7.2) gilt und $\theta < \theta_0 < 1$ ist. Aus der Definition von $\bar{\eta}$ folgt $\bar{\eta} = \hat{\eta}_i - \eta_i$, also

$$\frac{\|\hat{\eta}_i - \bar{\eta}\|}{\|\hat{\eta}_i\|} = \frac{\|\eta_i\|}{\|\hat{\eta}_i\|} = \frac{\varepsilon_i}{\hat{\varepsilon}_i} = \theta.$$

Damit ergibt sich

$$(1-\theta)\hat{\varepsilon}_i = (1-\theta)\|\hat{\eta}_i\| = \left(1 - \frac{\|\hat{\eta}_i - \bar{\eta}\|}{\|\hat{\eta}_i\|}\right)\|\hat{\eta}_i\| = \|\hat{\eta}_i\| - \underbrace{\|\hat{\eta}_i - \bar{\eta}\|}_{\geq \|\hat{\eta}_i\| - \|\bar{\eta}\|} \leq \|\bar{\eta}\| = \bar{\varepsilon},$$

also die untere Abschätzung mit $\kappa_1 = 1 - \theta_0$ und

$$\bar{\varepsilon} = \|\bar{\eta}\| \leq \|\hat{\eta}_i\| + \|\hat{\eta}_i - \bar{\eta}\| = \left(1 + \frac{\|\hat{\eta}_i - \bar{\eta}\|}{\|\hat{\eta}_i\|}\right)\|\hat{\eta}_i\| = (1+\theta)\|\hat{\eta}_i\| = (1+\theta)\hat{\varepsilon}_i,$$

also die obere Abschätzung mit $\kappa_2 = 1 + \theta_0$. □

Beachte, dass die Gültigkeit des Fehlerschätzers entscheidend von (7.2) abhängt, also nur für bereits hinreichend kleine Schrittweiten gilt.

7.2 Schrittweitenberechnung und adaptiver Algorithmus

Wir wollen nun untersuchen, wie man aus dem geschätzten Fehler effektiv eine neue Schrittweite berechnen kann. Hierzu benötigen wir eine weitere Annahme, nämlich dass der Fehler $\hat{\varepsilon}_i$ für kleine h_i von der Form

$$\hat{\varepsilon}_i \approx c_i h_i^{\hat{p}+1} \tag{7.4}$$

ist. Für Runge-Kutta-Verfahren ist dies erfüllt, falls f $\hat{p}+2$-mal stetig differenzierbar ist, wobei sich die c_i gerade aus dem zu $h_i^{\hat{p}+1}$ gehörigen Koeffizienten der Taylor-Entwicklung ergeben. Allerdings ist der exakte Wert von c_i unbekannt bzw. kann nur mit unverhältnismäßig großem Aufwand berechnet werden.

Sei nun eine Fehlerschranke $tol > 0$ für den lokalen Fehler vorgegeben. Wir führen jeweils einen Schritt mit beiden Verfahren $\widehat{\Phi}$ und Φ zum Zeitschritt h_i durch. Sei $\bar{\varepsilon}$ der gemäß (7.3)

berechnete Fehlerschätzer. Für kleine Schrittweiten gilt $\kappa_1 \approx \kappa_2 \approx 1$, also $\bar{\varepsilon} \approx \hat{\varepsilon}_i \approx c_i h_i^{\hat{p}+1}$. Hieraus können wir einen Schätzwert

$$\bar{c}_i = \frac{\bar{\varepsilon}}{h_i^{\hat{p}+1}}$$

für c_i berechnen. Die gewünschte Fehlertoleranz wird damit (approximativ) für diejenige Schrittweite $h_{i,neu}$ eingehalten, für die die Gleichung

$$tol = \bar{c}_i h_{i,neu}^{\hat{p}+1} = \frac{\bar{\varepsilon}}{h_i^{\hat{p}+1}} h_{i,neu}^{\hat{p}+1}$$

bzw.

$$h_{neu} = \sqrt[\hat{p}+1]{\frac{tol}{\bar{\varepsilon}}}\, h$$

gilt. Da diese Gleichungen (wegen der verschiedenen "$\approx$") nur näherungsweise gelten, führt man in der Praxis noch einen "Sicherheitsfaktor" $\rho \in (0,1)$ ein, um die Fehlerquellen bei der Fehlerschätzung zu kompensieren: man setzt

$$h_{i,neu} = \sqrt[\hat{p}+1]{\rho \frac{tol}{\bar{\varepsilon}}}\, h_i.$$

Eine typische Wahl hierfür ist $\rho = 0.9$.

Nach der Durchführung eines Schrittes mit Schrittweite h_i und der Schätzung des Fehlers $\bar{\varepsilon}$ können nun zwei Fälle auftreten:

(i) $\bar{\varepsilon} > tol$:
In diesem Fall wird der Schritt mit $h_i = h_{i,neu}$ erneut durchgeführt ("zurückweisen und wiederholen").

(ii) $\bar{\varepsilon} \leq tol$:
In diesem Fall wurde die gewünschte Genauigkeit tol erreicht. Der Schritt wird akzeptiert und die neue Schrittweite $h_{i,neu}$ wird als Schrittweite h_{i+1} für den nächsten Schritt verwendet ("akzeptieren").

Beachte, dass die Schrittweite in Fall (i) immer verkleinert wird. Die Wahl von $h_{i,neu}$ als Schrittweitenvorschlag für h_{i+1} in (ii) ist also ein notwendiger Schritt, damit auch Vergrößerungen der Schrittweite ermöglicht werden und darf daher auf keinen Fall weggelassen werden.

Formal lassen sich unsere Überlegungen in dem folgenden Grundalgorithmus zusammenfassen.

Algorithmus 7.3 (Einschrittverfahren mit Schrittweitensteuerung)

Eingabe: Anfangsbedingung (t_0, x_0), Endzeit T, Toleranz $tol > 0$, Sicherheitsfaktor ρ, Einschrittverfahren $\widehat{\Phi}$ und Φ mit unterschiedlichen Kosistenzordnungen $p \geq \hat{p} + 1$, Schrittweitenvorschlag h_0 für den ersten Schritt

(1) Setze $\tilde{x}_0 = x_0$, $i = 0$

(2) Falls $t_i = T$, beende den Algorithmus; falls $t_i + h_i > T$, setze $h_i = T - t_i$.

(3) Berechne $t_{i+1} = t_i + h_i$, $\tilde{x}^1_{i+1} = \Phi(t_i, \tilde{x}_i, h_i)$, $\tilde{x}^2_{i+1} = \widehat{\Phi}(t_i, \tilde{x}_i, h_i)$, den Fehlerschätzer $\bar{\varepsilon}$ und den Schrittweitenvorschlag $h_{i,neu}$

(4) Falls $\bar{\varepsilon} > tol$ setze $h_i = h_{i,neu}$ und gehe zu (3)

(5) Falls $\bar{\varepsilon} \leq tol$ setze $\tilde{x}_{i+1} := \tilde{x}^1_{i+1}$, $h_{i+1} := h_{i,neu}$, $i := i + 1$ und gehe zu (2)

Ausgabe: Werte der Gitterfunktion $\tilde{x}(t_i) = \tilde{x}_i$ in $t_0, \ldots, t_N = T$, □

Beachte, dass wir in (5) die genauere Lösung $\tilde{x}^1_{i+1}$ zum Weiterrechnen und für die Ausgabe verwenden. Diese Praxis wurde früher (und zum Teil noch heute) abgelehnt, da der Fehlerschätzer ja den Fehler in $\tilde{x}^2_{i+1}$ misst. Da das gesamte Verfahren aber auf der Annahme (7.2) beruht, die gerade besagt, dass Φ (also $\tilde{x}^1_{i+1}$) eine genauere Approximation ist, ist es durchaus gerechtfertigt, diesen Wert zu verwenden.

In der Praxis wird der Algorithmus in mehreren Punkten verfeinert:

(i) Statt in der euklidischen Norm wird $\bar{\varepsilon}$ in der Maximumsnorm

$$\bar{\varepsilon} = \|\bar{\eta}\|_\infty = \max_{i=1,\ldots,n} |\bar{\eta}_i|$$

berechnet, da diese schneller auszuwerten ist.

(ii) Der Bruch $tol/\bar{\varepsilon}$ in der Berechnung der neuen Schrittweite wird durch einen Wert ersetzt, in dem der absolute und der relative Fehler eingeht. Z.B. verwendet man statt $tol/\bar{\varepsilon}$ den Wert $1/err$ mit

$$err = \max_{j=1,\ldots,n} \frac{|\bar{\eta}_j|}{atol + |\widehat{\Phi}_j| \cdot rtol}$$

für absolute und relative Fehlertoleranzen $atol$ und $rtol > 0$; das Fehlerkriterium $\bar{\varepsilon} \leq tol$ wird dabei zu $err \leq 1$. Damit wird bei betragsmäßig großen Lösungskomponenten $|\widehat{\Phi}_j|$ ein größerer Fehler erlaubt, was Probleme mit Rundungsfehlern vermeidet, die bei sehr großen Komponenten ebenfalls groß werden können, weswegen eine rein absolute Fehlertoleranz in diesem Fall nicht einzuhalten wäre.

(iii) Die erlaubte Schrittweite wird durch Schranken h_{min} und h_{max} nach unten und oben beschränkt. Falls für die berechnete Schrittweite $h_{neu} < h_{min}$ gilt, so wird eine Warnung ausgegeben oder mit einer Fehlermeldung abgebrochen.

(iv) Der Variationsfaktor der Schrittweite, der durch

$$\sqrt[\hat{p}+1]{\rho \frac{tol}{\bar{\varepsilon}}} \quad \text{bzw. allgemeiner durch} \quad \sqrt[\hat{p}+1]{\rho \frac{1}{err}}$$

gegeben ist, wird durch Schranken ρ_{min} und ρ_{max} nach unten und oben beschränkt. Dadurch werden starke Schwankungen der Schrittweite vermieden.

(v) Im Falle eines Fehlberg-Verfahrens (vgl. Abschnitt 7.3) setzt man in Schritt (5) $h_{i+1} = h_i$ falls $h_{i,neu} \approx h_i$. Damit kann man Zwischenergebnisse aus dem i-ten Schritt effizient im $i+1$-ten Schritt verwenden.

Einige dieser Punkte werden in der Programmieraufgabe auf dem aktuellen Übungsblatt berücksichtigt; dort ist auch der oben angegebene Algorithmus noch einmal in etwas anderer Form dargestellt.

Abbildung 7.1 zeigt die Anwendung dieses Algorithmus auf das aus den Übungen bekannte restringierte Dreikörperproblem (Satellitenlaufbahn). Die Gitterpunkte t_i sind auf der in Kurvenform dargestellten Lösung markiert. Das Beispiel wurde mit der Routine `ode45` in MATLAB mit $atol = rtol = 10^{-7}$ gerechnet; die Routine verwendet zwei Runge-Kutta-Verfahren der Konsistenzordnung 4 und 5.

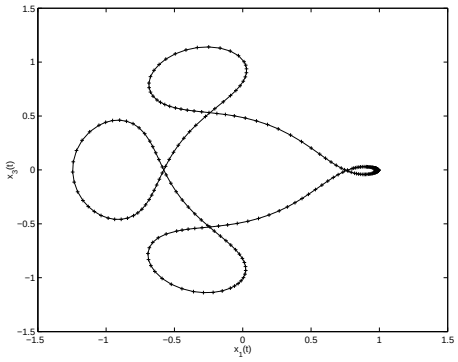

Abbildung 7.1: Adaptive Schrittweitensteuerung an einem Beispiel

7.3 Eingebettete Verfahren

Die in vielen Beispielen sehr effiziente Schrittweitensteuerung hat den Nachteil, dass man zur Berechnung des Fehlerschätzers zwei Einschrittverfahren $\widehat{\Phi}$ und Φ in jedem Schritt auswerten muss. Der Aufwand dieser Auswertungen kann allerdings beträchtlich reduziert werden, wenn man hierfür geschickt gewählte Verfahren verwendet, die sogenannten *eingebetteten Runge-Kutta-Verfahren*.

Wir betrachten zur Erläuterung zwei Verfahren $\widehat{\Phi}$ und Φ mit Konsistenzordnungen $\hat{p}$ und $p \geq \hat{p} + 1$. Bezeichnen wir die Stufen der Verfahren mit $\hat{k}_i$ bzw. k_i, so besteht die Idee der Einbettung einfach darin, dass die Verfahren so konstruiert werden, dass $\hat{k}_i = k_i$ für $i = 1, \ldots, s$ gilt. Für die Koeffizienten der Verfahren muss also $\hat{a}_{ij} = a_{ij}$ und $\hat{c}_i = c_i$ gelten, weswegen wir bei den alten Bezeichnugen a_{ij} und c_i bleiben. Lediglich $\hat{b}_i$ und b_i unterscheiden sich. Ein solches Paar $(\Phi, \widehat{\Phi})$ eingebetteter Verfahren wird mit RK$p(\hat{p})$ bezeichnet. Sie werden in einem Butcher-Tableau der Form

$$
\begin{array}{c|ccccc}
c_1 & & & & & \\
c_2 & a_{2\,1} & & & & \\
c_3 & a_{3\,1} & a_{3\,2} & & & \\
\vdots & \vdots & \vdots & \ddots & & \\
c_s & a_{s\,1} & a_{s\,2} & \cdots & a_{s\,s-1} & \\
\hline
 & b_1 & b_2 & \cdots & b_{s-1} & b_s \\
\hline
 & \hat{b}_1 & \hat{b}_2 & \cdots & \hat{b}_{s-1} & \hat{b}_s
\end{array}
$$

dargestellt. Um zu zeigen, dass eine solche Einbettung nicht ganz trivial ist, betrachten wir das klassische Runge-Kutta-Verfahren mit Ordnung 4, das durch die Koeffizienten

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} \end{array}$$

gegeben ist. Wir wollen dieses als Verfahren Φ der Ordnung $p = 4$ verwenden und versuchen, Koeffizienten $\hat{b}^T = (\hat{b}_1, \hat{b}_2, \hat{b}_3, \hat{b}_4)$ zu finden, so dass

$$\widehat{\Phi}(t, x, h) = x + h \sum_{i=1}^{4} \hat{b}_i k_i$$

ein Verfahren $\widehat{\Phi}$ der Ordnung $p = 3$ ergibt, womit wir ein RK4(3)-Verfahren erhalten würden. Wenn man die Bedingungsgleichungen aus Satz 4.5 (iii) löst, so stellt man fest, dass die einzige Lösung durch $\hat{b}^T = (1/6, 1/3, 1/3, 1/6)$ gegeben ist. Wir erhalten damit $\widehat{\Phi} = \Phi$, was keine sinnvolle Lösung ist, da sich die zwei Verfahren in der Konsistenzordnung echt unterscheiden müssen. So paradox es erscheinen mag: Um ein Verfahren niedrigerer Konsistenzordnung zu erhalten, müssen wir eine Stufe hinzunehmen, also s um 1 erhöhen.

Um die Berechnung der nötigen weiteren Stufe (nun wieder mit k_s bezeichnet) möglichst effizient zu gestalten, hilft ein Trick, den E. Fehlberg Ende der 1960er Jahre entwickelt hat: Wir wählen die letze Stufe gerade so, dass

$$k_s = k_1^* \tag{7.5}$$

gilt, wobei k_1^* die erste Stufe des nächsten Schritts des Verfahrens bezeichnet. Damit muss man trotzdem eine Stufe mehr berechnen, kann diese aber speichern und im nächsten Schritt des Verfahrens verwenden, wenn die Schrittweite $h_{i+1} = h_i$ gewählt werden kann (vgl. Punkt (v) in den praktischen Anmerkungen zu Algorithmus 7.3). Ein s-stufiges Verfahren mit diesem Trick ist also effektiv ein $s-1$-stufiges Verfahren.

Der Fehlberg-Trick lässt sich in Bedingungen an die Koeffizienten der letzten Stufe s ausdrücken. Wegen Konsistenz und Autonomieinvarianz gilt $k_1 = f(t, x)$, also $k_1^* = f(t + h, \Phi(t, x, h))$. Damit ergibt sich (7.5) zu

$$\underbrace{f(t + c_s h, x + h \sum_{j=1}^{s-1} a_{sj} k_j)}_{=k_s} = \underbrace{f(t + h, x + h \sum_{j=1}^{s} b_j k_j)}_{=k_1^*},$$

was gerade dann der Fall ist, wenn für die Koeffizienten der s-ten Stufe die Bedingungen

$$c_s = 1, \quad b_s = 0, \quad a_{sj} = b_j \text{ für } j = 1, \ldots, s-1 \tag{7.6}$$

gelten. Beachte dass es keine Garantie gibt, dass dieser Trick wirklich auf eine sinnvolle Lösung für $\hat{b}$ führt; wenn dies aber gelingt, so liefert er eine sehr effiziente Lösung.

Wir wollen diesen Trick auf das klassische Runge-Kutta-Verfahren anwenden. Dazu müssen wir die unbestimmten Koeffizienten in dem Butcher-Tableau

$$\begin{array}{c|ccccc} 0 & & & & & \\ \frac{1}{2} & \frac{1}{2} & & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & & \\ 1 & 0 & 0 & 1 & & \\ c_5 & a_{51} & a_{52} & a_{53} & a_{54} & \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} & 0 \\ \hline & \hat{b}_1 & \hat{b}_2 & \hat{b}_3 & \hat{b}_4 & \hat{b}_5 \end{array}$$

bestimmen. Aus (7.6) erhalten wir

$$c_5 = 1, \quad a_{51} = a_{54} = \frac{1}{6}, \quad a_{52} = a_{53} = \frac{1}{3}$$

und aus den Bedingungsgleichungen von Satz 4.5 (iii) erhält man die Lösungen

$$\hat{b}^T = (1/6, 1/3, 1/3, 1/6, 0) \quad \text{und} \quad \hat{b}^T = (1/6, 1/3, 1/3, 0, 1/6).$$

Die erste führt wiederum auf $\widehat{\Phi} = \Phi$, die zweite hingegen führt tatsächlich auf ein Verfahren mit maximaler Konsistenzordnung 3. Zusammenfassend erhalten wir

$$\begin{array}{c|ccccc} 0 & & & & & \\ \frac{1}{2} & \frac{1}{2} & & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & & \\ 1 & 0 & 0 & 1 & & \\ 1 & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} & \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} & 0 \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & 0 & \frac{1}{6} \end{array}$$

Dieses von Fehlberg Ende der 1960er Jahre entwickelte Verfahren ist für anspruchsvolle numerische DGL-Probleme durchaus schon zu gebrauchen.

Die heutzutage gebräuchlichsten eingebetteten Runge-Kutta-Verfahren wurden allerdings erst Anfang der 1980er Jahren von J.R. Dormand und P.J. Prince entwickelt. Es handelt sich um ein effektiv 6-stufiges RK5(4)-Verfahren mit den Koeffizienten

$$\begin{array}{c|ccccccc} 0 & & & & & & & \\ \frac{1}{5} & \frac{1}{5} & & & & & & \\ \frac{3}{10} & \frac{3}{40} & \frac{9}{40} & & & & & \\ \frac{4}{5} & \frac{44}{45} & -\frac{56}{15} & \frac{32}{9} & & & & \\ \frac{8}{9} & \frac{19372}{6561} & -\frac{25360}{2187} & \frac{64448}{6561} & -\frac{212}{729} & & & \\ 1 & \frac{9017}{3168} & -\frac{355}{33} & \frac{46732}{5247} & \frac{49}{176} & -\frac{5103}{18656} & & \\ 1 & \frac{35}{384} & 0 & \frac{500}{1113} & \frac{125}{192} & -\frac{2187}{6784} & \frac{11}{84} & \\ \hline & \frac{35}{384} & 0 & \frac{500}{1113} & \frac{125}{192} & -\frac{2187}{6784} & \frac{11}{84} & 0 \\ \hline & \frac{5179}{57600} & 0 & \frac{7571}{16695} & \frac{393}{640} & -\frac{92097}{339200} & \frac{187}{2100} & \frac{1}{40} \end{array}$$

sowie um ein 13-stufiges RK8(7)-Verfahren, das sich z.B. im Abschnitt 5.4 des Buches von Deuflhard/Bornemann findet. Diese Verfahren sind deswegen besonders gut, weil der von f unabhängige Anteil der Konstanten E in der Konsistenzabschätzung für Φ sehr klein im Vergleich zu anderen Verfahren ist. Das Dormand-Prince-RK5(4)-Verfahren ist MATLABs "Standardlöser" und ist dort unter dem Namen `ode45` implementiert. Im Internet finden sich MATLAB Implementierungen des RK8(7)-Verfahrens unter dem Namen `ode87.m` (zu finden mit Google mit dem Suchbegriff `ode87 matlab`).

Kapitel 8

Kollokationsmethoden

Eine Methode, um Runge-Kutta-Verfahren hoher Konsistenzordnung zu konstruieren, ohne dabei direkt die Bedingungsgleichungen aus Satz 3.7 zu verwenden, ist die sogenannte Kollokation. Sie beruht auf der Idee der Polynominterpolation, wobei die Lösung der Gleichung durch ein Polynom angenähert wird. Die Idee liegt darin, ein Polynom zu konstruieren, welches die Differentialgleichung an einer vorgegebenen Menge von Zeiten $t + c_1 h, \ldots, t + c_s h$ exakt erfüllt. Die Kollokation führt dabei in der Regel auf implizite Verfahren.

Definition 8.1 Sei $s \in \mathbb{N}$ und $c_1, \ldots, c_s \in [0, 1]$. Das *Kollokationspolynom* $p(t)$ vom Grad s ist das Polynom $p \in \mathcal{P}_s$, welches für gegebene $x \in \mathbb{R}^n$, $t_0 \in \mathbb{R}$, $h > 0$ und $f : D \to \mathbb{R}^n$, $D \subseteq \mathbb{R} \times \mathbb{R}^n$ die Bedingungen

$$p(t_0) = x \quad \text{und} \quad \dot{p}(t_0 + c_i h) = f(t_0 + c_i h, p(t_0 + c_i h)) \quad \text{für alle } i = 1, \ldots, s$$

erfüllt. Das *Kollokationsverfahren* ist dann gegeben durch

$$\Phi(t_0, x, h) = p(t_0 + h).$$

□

Beispiel 8.2 Für $s = 1$ ist p von der Form $p(t) = p_0 + p_1(t - t_0)$, also $\dot{p}(t) = p_1$. Aus der ersten Bedingung erhält man sofort $p_0 = x$. Für $c_1 = 0$ ergibt sich die zweite Bedingung zu

$$p_1 = f(t_0, p(t_0)) = f(t_0, x)$$

und man erhält $\Phi(t_0, x, h) = p(t_0 + h) = p_0 + h p_1 = x + h f(t_0, x)$, also das explizite Euler-Verfahren. Mit ähnlichen Rechnungen sieht man, dass man für $c_1 = 1$ das implizite Euler-Verfahren und für $c_1 = 1/2$ die Mittelpunktregel $\tilde{x}(t_0 + h) = \tilde{x}(t_0) + h f(t_0 + h/2, (\tilde{x}(t_0 + h) + \tilde{x}(t_0))/2)$ erhält.

Für $s = 2$ und $c_1 = 0$, $c_2 = 1$ erhält man die implizite Trapezregel. □

Bemerkung 8.3 Beachte, dass wir hier stillschweigend angenommen haben, dass ein Interpolationspolynom mit den angegebenen Bedingungen existiert und eindeutig ist. Letzteres ist nicht unbedingt der Fall, genauso wie implizite Runge-Kutta-Verfahren nicht unbedingt eine eindeutige Lösung besitzen müssen. Ein Gegenbeispiel werden wir in Bemerkung 8.6 betrachten. □

Der folgende Satz zeigt, dass die Kollokationsmethode tatsächlich wieder Runge-Kutta-Verfahren erzeugt.

Satz 8.4 Die Kollokationsmethode aus Definition 8.1 liefert das gleiche Einschrittverfahren Φ wie das s-stufige Runge-Kutta-Verfahren mit Koeffizienten $c_1, \dots, c_s$,

$$a_{ij} = \int_0^{c_i} L_j(\tau) d\tau, \qquad b_i = \int_0^1 L_i(\tau) d\tau \tag{8.1}$$

mit den Lagrange-Polynomen

$$L_i(\tau) = \prod_{\substack{j=1 \\ j \neq i}}^{s} \frac{\tau - c_j}{c_i - c_j}.$$

Beweis: Sei p das Kollokationspolynom und definiere $k_i := \dot{p}(t_0 + c_i h)$. Aus der Tatsache, dass $t \mapsto \sum_{j=1}^s \dot{p}(t_0 + c_j h) L_j(t)$ ein Polynom aus $\mathcal{P}_{s-1}$ ist, das an $s-1$ Stellen mit $\dot{p} \in \mathcal{P}_{s-1}$ übereinstimmt (vgl. dazu die Einführung in die Numerik), folgt, dass diese beiden Polynome übereinstimmen. Also gilt für alle $t \in \mathbb{R}$

$$\frac{d}{dt}[p(t_0 + th)] = \dot{p}(t_0 + th)h = h \sum_{j=1}^{s} \dot{p}(t_0 + c_j h) L_j(t) = h \sum_{j=1}^{s} k_j L_j(t).$$

Integration dieser Gleichung für t von 0 bis c_i liefert

$$p(t_0 + c_i h) - p(t_0) = h \int_0^{c_i} \sum_{j=1}^{s} k_j L_j(t),$$

was wegen $p(t_0) = x$ äquivalent ist zu

$$p(t_0 + c_j h) = x + h \sum_{j=1}^{s} k_j \int_0^{c_i} L_j(t) = x + h \sum_{j=1}^{s} k_j a_{ij}. \tag{8.2}$$

Einsetzen in die Gleichung für $\dot{p}$ in Definition 8.1 liefert

$$k_i = \dot{p}(t_0 + c_i h) = f(t_0 + c_i h, p(t + c_i h)) = f(t_0 + c_i h, x + h \sum_{j=1}^{s} k_j a_{ij}),$$

was genau die definierenden Gleichungen der Stufen k_i des Runge-Kutta-Verfahrens sind. Die erste Gleichung in (8.2) mit oberer Integrationsgrenze 1 statt c_i liefert

$$\Phi(t_0, x, h) = p(t_0 + h) = x + h \sum_{j=1}^{s} k_j \int_0^1 L_j(t) = x + h \sum_{j=1}^{s} k_j b_j,$$

und damit die Behauptung. □

Die Bedingungen in (8.1) kann man äquivalent auch als Gleichungssystem für die Koeffizienten ausdrücken: Da $\sum_{j=1}^s c_j^{k-1} L_j(\tau)$ für jedes $k = 1, \dots, s$ gerade das Interpolationspolynom durch (c_j, c_j^{k-1}) ist, gilt $\sum_{j=1}^s c_j^{k-1} L_j(\tau) = \tau^{k-1}$. Aus der ersten Bedingung aus (8.1) folgt daher

$$\sum_{j=1}^{s} a_{ij} c_j^{k-1} = \frac{c_i^k}{k} \quad \text{für alle } k = 1, \dots, q,\ i = 1, \dots, s \tag{8.3}$$

mit $q = s$ und aus der zweiten Bedingung folgt

$$\sum_{j=1}^{s} b_j c_j^{k-1} = \frac{1}{k} \quad \text{für alle } k = 1, \ldots, p \tag{8.4}$$

mit $p = s$. Da die c_i paarweise verschieden sind, liefern diese Gleichungen lineare Gleichungssysteme mit invertierbaren Matrizen für die Koeffizienten a_{ij} und b_j, weswegen diese eindeutig bestimmt sind. Die Bedingungen (8.3) und (8.4) mit $p = q = s$ sind also äquivalent zu (8.1).

8.1 Konsistenz

Satz 8.5 Jede Kollokationsmethode von Grad s besitzt die Konsistenzordnung s, falls $f \in C^{s+1}(D, \mathbb{R}^n)$ mit $D = \mathbb{R}\times\mathbb{R}^n$ ist. Zudem liefert das Kollokationspolynom mit $p(t_0) = x_0$ für alle $t \in [t_0, t_0 + h]$ eine Approximation der exakten Lösung $x(t) = x(t; t_0, x_0)$ der Ordnung $s + 1$, d.h.,

$$p(t) = x(t) + O(h^{s+1}) \quad \text{für alle } t \in [t_0, t_0 + h].$$

Beweis: Es genügt, die zweite Aussage zu beweisen, da die erste daraus für $t = t_0 + h$ folgt. Wir beweisen die Aussage zunächst für den Fall, dass die Lipschitzkonstante L von f bzgl. x global, also unabhängig von t und x gewählt werden kann.

Betrachte das n-dimensionale Interpolationspolynom q durch die Stützstellen $(t_0+c_ih, f(t_0+c_ih, x(t_0 + c_ih)))$. Für $E(t, h) = f(t, x(t)) - q(t)$, $t \in [t_0, t_0 + h]$ gilt dann

$$\dot{x}(t) = q(t) + E(t, h) = \sum_{i=1}^{s} f(t_0 + c_ih, x(t_0 + c_ih))L_i(t) + E(t, h)$$

Andererseits gilt für das Kollokationpolynom

$$\dot{p}(t) = \sum_{i=1}^{s} f(t_0 + c_ih, p(t_0 + c_ih))L_j(t),$$

also zusammen

$$\dot{x}(t) - \dot{p}(t) = \sum_{i=1}^{s} \underbrace{\left(f(t_0 + c_ih, x(t_0 + c_ih)) - f(t_0 + c_ih, p(t_0 + c_ih))\right)}_{=:\Delta_i f} L_i(t) + E(t, h). \tag{8.5}$$

Auf Grund der Lipschitzannahme an f können wir $\|\Delta_i f\|$ für alle $i = 1, \ldots, s$ abschätzen durch

$$\|\Delta_i f\| \le L \max_{t\in[t_0,t_0+h]} \|x(t) - p(t)\|.$$

Nach dem Satz über den Interpolationsfehler bei der Polynominterpolation gilt zudem

$$\|E(t, h)\| \le h^s \max_{t\in[t_0,t_0+h]} \frac{\|x^{(s+1)}(t)\|}{s!}.$$

Integration von (8.5) von t_0 nach $t \le t_0 + h$ und Ausnutzen von $x(t_0) - p(t_0) = 0$ liefert

$$x(t) - p(t) = \sum_{i=1}^{s} \Delta_i f \int_{t_0}^{t} L_i(\tau) d\tau + \int_{t_0}^{t} E(\tau, h) d\tau \tag{8.6}$$

und damit

$$\max_{t \in [t_0, t_0+h]} \|x(t) - p(t)\| \le hC_1 L \max_{t \in [t_0, t_0+h]} \|x(t) - p(t)\| + C_2 h^{s+1},$$

woraus die Behauptung folgt wenn $hC_1 L < 1$.

Falls f nicht global Lipschitz stetig in x ist, betrachten wir eine kompakte Menge K von Anfangsbedingungen und die kompakte Menge K_2 aus dem Beweis von Satz 2.7 mit $T = t_0 + h$. Sei $B = B_R(0) \in \mathbb{R} \times \mathbb{R}^n$, wobei $R > 0$ so groß ist, dass $K_2 \subset B_R(0)$. Wir betrachten nun eine C^{s+1}-Funktion $\rho : \mathbb{R} \to \mathbb{R}$ mit $\rho(r) = 1$ für $r \le R$, $\rho(r) \in [0, 1]$ für $r \in [R, R + 1]$ und $\rho(r) = 0$ für $r \ge R$ und setzen $\tilde{f}(t, x) = \rho(\|(t, x)\|) f(t, x)$. Dann ist $\tilde{f} \in C^{s+1}(D, \mathbb{R}^n)$ und damit Lipschitz und weil die Lipschitz-Konstante L für $(t, x), (t, y) \notin B_{R+1}(0)$ offenbar gleich 0 ist, ist $\tilde{f}$ global Lipschitz. Auf $\tilde{f}$ kann dann der erste Teil des Beweises angewendet werden. Für hinreichend kleines $h > 0$ folgt daraus $p(t) \in K_2$ für alle $t \in [t_0, t_0 + h]$. Dort stimmt f aber mit $\tilde{f}$ überein, weswegen die Konsistenz auch für f gilt. □

Bemerkung 8.6 Der Satz wird i.A. falsch, wenn $D \neq \mathbb{R} \times \mathbb{R}^n$ ist. Betrachte z.B. die eindimensionale autonome Gleichung mit

$$f(x) = \frac{x}{1 - x}$$

und $D = \mathbb{R} \setminus \{1\}$. Für $(t_0, x_0) = (0, 0)$ lautet die Lösung offensichtlich $x(t; 0, 0) \equiv 0$. Kollokation mit $s = 1$ und $c_1 = 1$ (also mit dem impliziten Euler) liefert aber das Interpolationspolynom $p(t) = (1 - h)t/h$, denn es gilt

$$p(0) = 0 \quad \text{und} \quad \dot{p}(0 + h) = \frac{1 - h}{h} = \frac{1 - h}{1 - (1 - h)} = f(1 - h) = f(p(0 + h)).$$

Daraus folgt $\Phi(0, 0, h) = p(0 + h) = 1 - h$, was für $h \to 0$ gegen 1 und nicht wie für Konsistenz nötig gegen 0 konvergiert.

Tatsächlich ist das obige Polynom nicht das eindeutige Interpolationspolynom. Man prüft leicht nach, dass $p(t) \equiv 0$ ebenfalls die Bedingungen des Kollokationsverfahrens erfüllt; für dieses Polynom gilt dann auch die Konsistenz. In der Regel bewirkt ein gut gewählter Startwert, dass man beim iterativen Lösen der nichtlinearen Gleichungen zur Bestimmung von Φ gegen die "richtige", also die konsistente Lösung konvergiert. □

Der folgende Satz zeigt, dass man die Konsistenzordnung unter gewissen Bedingungen noch verbessern kann. Da wir mit diesem Satz eine Konsistenz- und damit auch Konvergenzordnung $p > s$ erhalten kann, spricht man auch von „Superkonvergenz".

Satz 8.7 Betrachte ein Kollokationsverfahren, welches die Bedingung (8.4) für ein $p \in \mathbb{N}$ mit $s \le p \le 2s$ erfüllt. Dann besitzt das Verfahren die Konsistenzordnung p falls $f \in C^{p+1}(D, \mathbb{R}^n)$ und $D = \mathbb{R} \times \mathbb{R}^n$.

Beweis: Wir betrachten das Kollokationspolynom p als Lösung der gestörten Gleichung

$$\dot{p}(t) = f(t, p(t)) + \delta(t)$$

mit $\delta(t) = \dot{p}(t) - f(t, p(t))$. Ziehen wir die exakte Gleichung von dieser Gleichung ab, so erhalten wir mit Taylor-Entwicklung der Ordnung 1

$$\dot{p}(t) - \dot{x}(t) = f(t, p(t)) + \delta(t) - f(t, x(t)) = \frac{\partial f}{\partial x}(t, x(t))\Big(p(t) - x(t)\Big) + \delta(t) + r(t)$$

mit $r(t) = O(\|p(t) - x(t)\|^2) = O(h^{2s+2})$ gemäß Satz 8.5. Aus der Lösungsformel für inhomogene lineare Differentialgleichungen („Variation der Konstanten“) und $p(t_0) - x(t_0) = 0$ folgt

$$p(t_0 + h) - x(t_0 + h) = \int_{t_0}^{t_0+h} \Theta(t_0 + h, \tau)\,(\delta(\tau) + r(\tau))\,d\tau,$$

wobei Θ die Fundamentallösung der homogenen Gleichung $\dot{y}(t) = \frac{\partial f}{\partial x}(t, x(t))y(t)$ bezeichnet. Das Integral über $\Theta(t_0 + h, \tau)r(\tau)$ ist von der Ordnung $O(h^{2s+3})$. Die Funktion $g(\tau) := \Theta(t_0 + h, \tau)\delta(\tau)$ besitzt gerade die Nullstellen $\tau = t_0 + hc_1, \ldots, t_0 + hc_s$. Wenden wir nun die Quadraturformel

$$\int_{t_0}^{t_0+h} g(\tau)d\tau \approx \sum_{j=1}^{s} b_j g(t_0 + hc_j)$$

an, so impliziert Bedingung (8.4), dass diese Quadraturformel Polynome vom Grad $p - 1$ exakt integriert, woraus mit Satz 5.1 aus der „Einführung in die Numerischen Mathematik“

$$\left\| \int_{t_0}^{t_0+h} g(\tau)d\tau - \sum_{j=1}^{s} b_j g(t_0 + hc_j) \right\| \le Ch^{p+1}$$

und wegen $g(t_0 + hc_j) = 0$ also $\| \int_{t_0}^{t_0+h} g(\tau)d\tau\| \le Ch^{p+1}$ folgt. Die Konstanten in dem O-Term hängen dabei von den Ableitungen von p ab und ähnlich wie im Beweis von Satz 8.5 kann man zeigen, dass diese durch eine von h unabhängigen Konstante beschränkt sind. Damit folgt die behauptete Konsistenz für $\Phi(t_0, x, h) = p(t_0 + h)$. □

Bemerkung 8.8 Der Beweis zeigt insbesondere, dass die Ordnung des Kollokationsverfahrens durch die Ordnung des Quadraturverfahrens mit Stützstellen c_j und Gewichten b_j bestimmt ist. □

8.2 Beispiele

In diesem Abschnitt geben wir einige Beispiele für Kollokationsverfahren an.

Gauß-Verfahren Wählt man $c_1, \ldots, c_s$ als die Nullstellen des s-ten verschobenen Legendre Polynoms

$$\frac{d^s}{dx^s}\left(x^s(x-1)^s\right),$$

so erhält man eine Quadraturformel mit Ordnung $2s$, vgl. Abschnitt 5.3 der „Einführung in die Numerische Mathematik“. Nach Bemerkung 8.8 besitzt die zugehörige Quadraturformel also die gleiche Ordnung. Ihre Butcher-Tableaus für $s = 2$ und $s = 3$, d.h. mit Ordnungen $p = 4$ und $p = 6$ sind gegeben durch

$$\begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

und

$$\begin{array}{c|ccc} \frac{1}{2} - \frac{\sqrt{15}}{10} & \frac{5}{36} & \frac{2}{9} - \frac{\sqrt{15}}{15} & \frac{5}{36} - \frac{\sqrt{15}}{30} \\ \frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\ \frac{1}{2} + \frac{\sqrt{15}}{10} & \frac{5}{36} + \frac{\sqrt{15}}{30} & \frac{2}{9} + \frac{\sqrt{15}}{15} & \frac{5}{36} \\ \hline & \frac{5}{18} & \frac{4}{9} & \frac{5}{18} \end{array}$$

Radau-Verfahren Bei den Radau-Methoden legt man entweder $c_1 = 0$ oder $c_s = 1$ fest und bestimmt die restlichen Koeffizienten dann so, dass die Ordnung maximal, d.h. gleich $2s - 1$ wird. Die Verfahren mit $c_s = 1$ werden Radau IIA-Methoden genannt. Für ein Butcher-Tableau siehe Abschnitt 6.3.

Lobatto IIIA-Verfahren Diese Verfahren besitzen die höchste Ordnung $p = 2s-2$ unter den Bedingungen $c_1 = 0$ und $c_s = 1$. Die Stützstellen $c_2, \ldots, c_{s-1}$ müssen dazu gerade die Nullstellen des Polynoms

$$\frac{d^{s-2}}{dx^{s-2}} \left(x^{s-1}(x-1)^{s-1} \right)$$

sein. Für $s = 2$ erhält man die implizite Trapezregel, für $s = 3$ und $s = 4$ ergeben sich die Butcher-Tableaus

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{5}{24} & \frac{1}{3} & -\frac{1}{24} \\ 1 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

und

$$\begin{array}{c|ccccc} 0 & 0 & 0 & 0 & 0 \\ \frac{5-\sqrt{5}}{10} & \frac{11+\sqrt{5}}{120} & \frac{25-\sqrt{5}}{120} & \frac{25-13\sqrt{5}}{120} & \frac{-1+\sqrt{5}}{120} \\ \frac{5+\sqrt{5}}{10} & \frac{11-\sqrt{5}}{120} & \frac{25+13\sqrt{5}}{120} & \frac{25+\sqrt{5}}{120} & \frac{-1-\sqrt{5}}{120} \\ 1 & \frac{1}{12} & \frac{5}{12} & \frac{5}{12} & \frac{1}{12} \\ \hline & \frac{1}{12} & \frac{5}{12} & \frac{5}{12} & \frac{1}{12} \end{array}$$

8.3 Unstetige Kollokation

Die Klasse der Kollokationsverfahren enthält nicht alle in der Praxis gebräuchlichen impliziten Verfahren. Um die durch die Kollokation möglichen relativ einfachen Konsistenzbeweise auf größere Klassen von Verfahren anzuwenden, wurde die Idee der unstetigen Kollokation entwickelt. Dabei werden in der Kollokation $c_1 = 0$ und $c_s = 1$ gesetzt und die vier Bedingungen

$$p(t_0) = x, \quad \Phi(t_0, x, h) = p(t_0 + h) \quad \text{und} \quad \dot{p}(t_0 + c_i h) = f(t_0 + c_i h, p(t + c_i h))$$

für $i = 1$ und $i = s$ kombiniert zu den zwei schwächeren Bedingungen

$$p(t_0) = x - hb_1\Big(\dot{p}(t_0) - f(t_0), p(t_0)\Big)$$

und

$$\Phi(t_0, x, h) = p(t_0 + h) - hb_s\Big(\dot{p}(t + h) - f(t + h, p(t + h))\Big).$$

Da damit nur noch $s - 1$ statt $s + 1$ Bedingungen an p gestellt werden, ist p nun aus $\mathcal{P}_{s-2}$.

Auch diese Klasse von Verfahren ist äquivalent zu s-stufigen impliziten Runge-Kutta-Verfahren und die Sätze 8.5 und 8.7 gelten weiterhin, allerdings wegen der geringeren Ordnung der Polynome mit $O(h^{s-1})$ bzw. für $s \le p \le 2s - 2$. Die geänderten Bedingungen wirken sich in den Beweisen in Abschätzung (8.6) aus, wo ein zusätzlicher Fehlerterm der Ordnung $O(h^{s-1})$ entsteht.

Zu den Methoden dieser Klasse gehören gewisse Radau und Lobatto-Verfahren, z.B. die **Lobatto IIIB-Verfahren**, bei denen $a_{i1} = b_1$ und $a_{is} = 0$ festgelegt wird und die restlichen Koeffizienten so gewählt werden, dass die Ordnung maximal, also $p = 2s - 2$ wird. Für $s = 3$ und $s = 4$ ergeben sich so die Tableaus

$$\begin{array}{c|ccc} 0 & \frac{1}{6} & -\frac{1}{6} & 0 \\ \frac{1}{2} & \frac{1}{6} & \frac{1}{3} & 0 \\ 1 & \frac{1}{6} & \frac{5}{6} & 0 \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array} \quad \text{und} \quad \begin{array}{c|cccc} 0 & \frac{1}{12} & \frac{-1-\sqrt{5}}{24} & \frac{2-1+\sqrt{5}}{24} & 0 \\ \frac{5-\sqrt{5}}{10} & \frac{1}{12} & \frac{25+\sqrt{5}}{120} & \frac{25-13\sqrt{5}}{120} & 0 \\ \frac{5+\sqrt{5}}{10} & \frac{1}{12} & \frac{25+13\sqrt{5}}{120} & \frac{25-\sqrt{5}}{120} & 0 \\ 1 & \frac{1}{12} & \frac{11-\sqrt{5}}{24} & \frac{11+\sqrt{5}}{24} & 0 \\ \hline & \frac{1}{12} & \frac{5}{12} & \frac{5}{12} & \frac{1}{12} \end{array}$$

Kapitel 9

Mehrschrittverfahren

Die Mehrschrittverfahren unterscheiden sich von den Einschrittverfahren dadurch, dass der Wert $\tilde{x}(t_{i+1})$ nicht nur von $\tilde{x}(t_i)$ sondern von einer ganzen Reihe von Vorgängerwerten $\tilde{x}(t_{i-k+1}), \ldots, \tilde{x}(t_i)$ abhängt. Wie schon bei den Einschrittverfahren gibt es explizite und implizite Mehrschrittverfahren; erstere geben einen expliziten Ausdruck für $\tilde{x}(t_{i+1})$, während bei letzteren noch eine Fixpunktgleichung zu lösen ist. Man hofft dabei, dass man — da ja durch die größere Anzahl von Punkten mehr Information zur Verfügung steht — im Vergleich zu Einschrittverfahren gleicher Konsistenzordnung mit weniger Auswertungen von f pro Schritt auskommt. Tatsächlich werden wir sehen, dass diese Hoffnung berechtigt ist.

Zur Motivation betrachten wir wieder Verfahren, die wir heuristisch aus numerischen Integrationsformeln ableiten. Wir nehmen dabei konstante Schrittweite $h_i = h$ an. Wenn wir in der Integralgleichung

$$x(t_{i+1}) = x(t_{i-1}) + \int_{t_{i-1}}^{t_{i+1}} f(t, x(t))dt$$

das Integral durch die Mittelpunktregel

$$\int_{t_{i-1}}^{t_{i+1}} f(t, x(t))dt \approx 2hf(t_i, x(t_i))$$

ersetzen, so erhalten wir die *explizite Mittelpunktregel*

$$\tilde{x}(t_{i+1}) = \tilde{x}(t_{i-1}) + 2hf(t_i, \tilde{x}(t_i)).$$

Wählen wir die Simpson-Regel

$$\int_{t_{i-1}}^{t_{i+1}} f(t, x(t))dt \approx \frac{h}{3}\Big(f(t_{i+1}, x(t_{i+1})) + 4f(t_i, x(t_i)) + f(t_{i-1}, x(t_{i-1}))\Big),$$

so erhalten wir das (implizite) *Milne-Simpson-Verfahren*

$$\tilde{x}(t_{i+1}) = \tilde{x}(t_{i-1}) + \frac{h}{3}(f(t_{i+1}, \tilde{x}(t_{i+1})) + 4f(t_i, \tilde{x}(t_i)) + f(t_{i-1}, \tilde{x}(t_{i-1}))).$$

Eine Verallgemeinerung, die diese beiden Verfahren umfasst, ist die folgende Klasse der *linearen Mehrschrittverfahren (MSV)*.

Definition 9.1 Ein *k-stufiges lineares Mehrschrittverfahren (MSV)* ist gegeben durch die Gleichung

$$\begin{aligned} & a_k\tilde{x}(t_{i+k}) + a_{k-1}\tilde{x}(t_{i+k-1}) + \ldots + a_0\tilde{x}(t_i) \\ &= \; h\Big(b_k\tilde{f}(t_{i+k}) + b_{k-1}\tilde{f}(t_{i+k-1}) + \ldots + b_0\tilde{f}(t_i)\Big) \end{aligned} \tag{9.1}$$

mit der Abkürzung $\tilde{f}(t_j) = f(t_j, \tilde{x}(t_j))$, wobei $a_k \neq 0$ ist □

Mit dieser Klasse von Verfahren wollen wir uns schwerpunktmäßig beschäftigen. Wenn $b_k = 0$ ist, so ist das Verfahren explizit, da es direkt nach $\tilde{x}(t_{i+k})$ aufgelöst werden kann. Falls $b_k \neq 0$ ist, so kann man die entstehenden Gleichungen analog zu den impliziten Einschrittverfahren lösen (algebraisch, Fixpunkt-Iteration, Newton-Verfahren,...). Wir beschränken uns zunächst auf den Fall äquidistanter Schrittweiten $h_i = h$ und gehen am Schluss dieses Kapitels (kurz) auf variable Schrittweiten und Schrittweitensteuerung ein.

Bemerkung 9.2 (i) Zum Start eines Mehrschrittverfahrens benötigt man neben dem Anfangswert $\tilde{x}(t_0)$ noch die Werte $\tilde{x}(t_1), \ldots, \tilde{x}(t_{k-1})$. Diese werden üblicherweise durch ein geeignetes Einschrittverfahren bestimmt. Details dazu besprechen wir etwas später.
(ii) Wenn man die $\tilde{f}$-Werte eines Schrittes zwischenspeichert, so muss in jedem Schritt lediglich der Wert $\tilde{f}(t_{i+k-1})$ neu berechnet werden. Ein explizites lineares MSV kommt also mit einer f-Auswertung pro Schritt aus. □

Zur Analyse von MSV hat sich der folgende (aus der Theorie der dynamischen Systeme stammende) Formalismus als sehr geeignet erwiesen.

Definition 9.3 Auf dem Raum der Gitterfunktionen $\Delta_{\mathcal{T}} := \{f : \mathcal{T} \to \mathbb{R}^n\}$ definieren wir den *Shift-Operator* $E : \Delta_{\mathcal{T}} \to \Delta_{\mathcal{T}}$ mittels

$$E(f)(t_i) = f(t_{i+1}).$$

Hierbei erweitern wir unser Gitter formal zu einem Gitter mit unendlich vielen Gitterpunkten $\mathcal{T} = \{t_0, t_1, t_2, \ldots\}$. □

Beispiel 9.4 Für eine Gitterfunktion mit $f(t_i) = a_i$ mit $a_i = (2, 4, 8, 16, 32, \ldots)$ gilt also $E(f) = \tilde{f}$ mit $\tilde{f}(t_i) = \tilde{a}_i$ mit $\tilde{a}_i = (4, 8, 16, 32, 64, \ldots)$. Die Wertefolge wird also um eine Stelle nach links verschoben, woraus sich der Name Shift-Operator (manchmal auch 'Linksshift' genannt) ergibt. □

Der Shift-Operator erlaubt die folgende, sehr kompakte Schreibweise von Mehrschrittverfahren: Mit den Polynomen

$$\begin{aligned} P_a(z) &= a_0 + a_1 z + \ldots + a_k z^k \\ P_b(z) &= b_0 + b_1 z + \ldots + b_k z^k \end{aligned}$$

kann man (9.1) als

$$P_a(E)(\tilde{x})(t_i) = hP_b(E)(\tilde{f})(t_i) \tag{9.2}$$

schreiben, wobei die Potenz E^j des Shift-Operators die j-malige Hintereinanderausführung des Operators bedeutet.

Wir wollen nun die Konvergenz von Mehrschrittverfahren untersuchen und dabei das für die Einschrittverfahren bewiesene Resultat "Konsistenz + Lipschitzbedingung ⇒ Konvergenz" verallgemeinern. Wir beginnen mit der Konsistenz.

9.1 Konsistenz

Bei der Untersuchung der Konsistenz bei Einschrittverfahren haben wir mittels

$$\varepsilon := \|\Phi(t, x, h) - x(t + h; t, x)\|$$

den Konsistenzfehler durch Vergleich des numerischen Verfahrens mit der exakten Lösung erhalten. Die Größe ε lässt sich aber auch anders interpretieren:
Für die numerisch berechnete Gitterfunktion gilt gerade die Gleichung

$$\|\tilde{x}(t_{i+1}) - \Phi(t_i, \tilde{x}(t_i), h)\| = 0$$

Setzen wir hier nun die exakte Lösungsfunktion $x(t) = x(t; t_0, x_0)$ ein, so erhalten wir

$$\|x(t_{i+1}) - \Phi(t_i, x(t_i), h)\| = \varepsilon,$$

also gerade wieder unseren Konsistenzfehler für $x = x(t_i)$. Beachte, dass jede Funktion $x : [t_0, T] \to \mathbb{R}^n$ auch eine Gitterfunktion auf den in $[t_0, T]$ liegenden Gitterpunkten ist.

Dieses Verfahren "Einsetzen der exakten Lösung in die numerische Gleichung" lässt sich auf viele numerische Verfahren anwenden, z.B. auf unsere Mehrschrittverfahren. In der kompakten Schreibweise (9.2) müssen wir also die Norm des Konsistenzfehlers

$$L(x, t, h) = P_a(E)(x)(t) - hP_b(E)(f)(t) = P_a(E)(x)(t) - hP_b(E)(\dot{x})(t)$$

bestimmen. Beachte, dass der Parameter x hier eine Funktion $x : [t_0, T] \to \mathbb{R}^n$ und dass L nur für solche Parametertripel (x, t, h) definiert ist, für die $[t, t + hk] \subset [t_0, T]$ gilt.

Definition 9.5 Ein lineares Mehrschrittverfahren besitzt die *Konsistenzordnung* p, falls für jede $p + 1$-mal stetig differenzierbare Lösung $x : [t_0, T] \to \mathbb{R}^n$ der Differentialgleichung (1.1) die Abschätzung

$$L(x, t, h) = O(h^{p+1})$$

gleichmäßig in t gilt für alle t, h, in denen $L(x, t, h)$ definiert ist. □

Interessanterweise hängt die Definition des Konsistenzfehlers L *nicht* von f ab, da wir die auftretenden Werte des Vektorfeldes f durch die Ableitungen $\dot{x}$ ersetzt haben. Dies nutzt der folgende Satz aus, der Bedingungen angibt, anhand derer man die Konsistenzordnung eines Mehrschrittverfahrens überprüfen kann.

Satz 9.6 Ein lineares Mehrschrittverfahren besitzt genau dann die Konsistenzordnung $p \in \mathbb{N}$, wenn eine der folgenden äquivalenten Bedingungen erfüllt ist.

(i) Für jede beliebige $p+1$-mal stetig differenzierbare Funktion $x : [t_0, T] \to \mathbb{R}^n$ gilt die Abschätzung

$$L(x, t, h) = O(h^{p+1})$$

gleichmäßig in t für alle t, h, in denen $L(x, t, h)$ definiert ist.

(ii) $L(Q, 0, h) = 0$ für alle Polynome $Q \in \mathcal{P}_p$.

(iii) Es gilt

$$\sum_{j=0}^{k} a_j = 0, \ \sum_{j=0}^{k} a_j j^l = l \sum_{j=0}^{k} b_j j^{l-1} \text{ für } l = 1, \ldots, p$$

mit der Konvention $0^0 = 1$.

Beweis: Wir zeigen die Äquivalenz durch die Implikationen

$$\text{(i)} \Rightarrow \text{Konsistenzordnung } p \Rightarrow \text{(ii)} \Rightarrow \text{(i)} \Rightarrow \text{(iii)} \Rightarrow \text{(i)}$$

"(i) $\Rightarrow$ Konsistenzordnung p": Dies folgt direkt, da mit jeder beliebigen Funktion auch jede Lösung die behauptete Abschätzung erfüllt.

"Konsistenzordnung $p \Rightarrow$ (ii)": Gegeben sei ein beliebiges Polynom $Q \in \mathcal{P}_p$. Mit $f(t, x) = \dot{Q}(t)$ erhalten wir eine "triviale" Differentialgleichung, deren Lösung Q ist. Nach Definition der Konsistenzordnung folgt also

$$L(Q, 0, h) = O(h^{p+1}).$$

Da Q ein Polynom vom Grad $\leq p$ ist, muss auch $L(Q, 0, h)$ ein Polynom vom Grad $\leq p$ in h sein, weswegen $L(Q, 0, h) = 0$ sein muss.

"(ii) $\Rightarrow$ (i)": Sei x eine beliebige $p+1$-mal differenzierbare Funktion und sei Sei $Q \in \mathcal{P}_p$ das Polynom, das durch die ersten p Terme der Taylorentwicklung von x in t^* definiert ist. Dann gilt

$$x(t) = Q(t) + O(h^{p+1}) \text{ für alle } t \in [t^* - kh, t^* + kh].$$

Aus der Struktur von L folgt damit sofort die Abschätzung

$$L(x, t, h) = L(Q, t, h) + O(h^{p+1}).$$

Diese Abschätzung ist gleichmäßig in $t \in [t_0, T]$, da das den $O(h^{p+1})$-Term bestimmende Taylor-Restglied gleichmäßig beschränkt auf kompakten Intervallen ist. Aus (ii) wissen wir, dass $L(Q, 0, h) = 0$ gilt, woraus (durch "Verschieben" des Polynoms) auch $L(Q, t, h) = 0$ folgt, was schließlich die Behauptung liefert.

"(i) $\Rightarrow$ (iii)": Die Implikation aus (i) gilt insbesondere für konstante Funktionen $x \equiv c$. Für diese gilt

$$O(h^{p+1}) = L(x, 0, h) = P_a(E)x(t) - \underbrace{hP_b(E)\dot{x}(t)}_{=0} = \sum_{j=0}^{k} a_j c.$$

Da die rechte Seite unabhängig von h ist, kann dies nur gelten, wenn die Summe der a_j gleich Null ist, was die erste Gleichung in (iii) zeigt.

Für die weiteren Gleichungen in (iii) betrachten wir (i) mit $x(t) = \exp(t)$. Wegen

$$E^j(\exp)(0) = \exp(jh) = \exp(h)^j \quad \text{und} \quad \frac{d}{dt}\exp(t) = \exp(t)$$

folgt

$$L(\exp, 0, h) = P_a(\exp(h)) - hP_b(\exp(h))$$

Wir betrachten die Taylorentwicklung dieses Ausdrucks in $h = 0$. Diese lautet

$$L(\exp, 0, h) = \sum_{l=0}^{p} \frac{1}{l!} \sum_{j=0}^{k} a_j j^l h^l - \sum_{l=0}^{p-1} \frac{1}{l!} \sum_{j=0}^{k} b_j j^l h^{l+1} + O(h^{p+1}).$$

Aus (i) wissen wir $L(\exp, 0, h) = O(h^{p+1})$, weswegen

$$\sum_{l=0}^{p} \frac{1}{l!} \sum_{j=0}^{k} a_j j^l h^l - \sum_{l=0}^{p-1} \frac{1}{l!} \sum_{j=0}^{k} b_j j^l h^{l+1} = O(h^{p+1})$$

sein muss. Dieser Summenausdruck ist ein Polynom vom Grad $\leq p$ in h, und kann daher nur von der Ordnung $O(h^{p+1})$ sein, wenn er bereits Null ist. Dies wiederum kann nur dann gelten, wenn sich die Koeffizienten zu gleichen Potenzen von h zu Null addieren, also

$$\frac{1}{l!} \sum_{j=0}^{k} a_j j^l - \frac{1}{(l-1)!} \sum_{j=0}^{k} b_j j^{l-1} = 0$$

gilt. Dies sind gerade die weiteren Gleichungen aus (iii).

"(iii) $\Rightarrow$ (i)": Die Taylorentwicklung von L für allgemeine x in $h = 0$ lautet

$$\begin{aligned} L(x, t, h) \quad = \quad & \sum_{l=0}^{p} \frac{1}{l!} \sum_{j=0}^{k} a_j j^l h^l x^{(l)}(t) \\ & - \; h \; \left(\sum_{l=0}^{p-1} \frac{1}{l!} \sum_{j=0}^{k} b_j j^l h^l x^{(l+1)}(t) \right) + O(h^{p+1}). \end{aligned}$$

Wenn die Gleichungen aus (iii) gelten, so fallen alle diese Summanden weg, so dass nur $O(h^{p+1})$ übrig bleibt. Diese Abschätzung ist wegen der gleichmäßigen Beschränktheit des Taylor-Restgliedes gleichmäßig in $t \in [t_0, T]$, weswegen (i) folgt. □

Bemerkung 9.7 Der Fall $p = 1$ ist hierbei besonders interessant, da er die Frage beantwortet, wann ein Verfahren überhaupt konsistent ist. Für $p = 1$ erhalten wir aus (iii) die Bedingungen

$$\sum_{j=0}^{k} a_j = 0 \quad \text{und} \quad \sum_{j=0}^{k} a_j j = \sum_{j=0}^{k} b_j.$$

Beide Bedingungen lassen sich mit Hilfe der Polynome P_a und P_b ausdrücken, sie sind gerade äquivalent zu

$$P_a(1) = 0 \quad \text{und} \quad P_a'(1) = P_b(1).$$

Diese Bedingungen entsprechen der Bedingung $\sum b_i = 1$ bei den Runge-Kutta-Verfahren. Insbesondere muss für konsistente Verfahren die 1 eine Nullstelle von P_a sein. Wir werden im nächsten Teilabschnitt sehen, dass auch die weiteren Nullstellen von P_a eine wichtige Rolle bei der Konvergenzanalyse von Mehrschrittverfahren spielen. □

9.2 Stabilität

Wir wollen nun ein geeignetes Analogon der Lipschitzbedingung für Einschrittverfahren entwickeln. In der Konvergenztheorie der Einschrittverfahren haben wir diese Bedingung verwendet, um sicher zu stellen, dass sich die in vergangenen Schritten gemachten Fehler im aktuellen Schritt nicht zu sehr verstärken.

Sicherlich sollte die rechte Seite unseres Mehrschrittverfahrens (9.1) eine ähnliche Lipschitzbedingung erfüllen, diese erhalten wir aber "geschenkt", da wir ja nur Lipschitz-stetige Vektorfelder f betrachten. Leider reicht es aber nicht aus, wenn f Lipschitz-stetig ist. Diese Bedingung besagt ja nur, dass sich kleine Fehler in den vergangenen $\tilde{x}$ in der *rechten* Seite unseres Verfahrens wenig auswirken. Wir benötigen zusätzlich noch eine Bedingung, die uns garantiert, dass kleine Fehler auf der *linken* Seite von (9.1) auch nur kleine Fehler in $\tilde{x}(t_{i+k})$ hervorrufen.

Um zu sehen, dass dies ein nichttriviales Problem ist, betrachten wir zwei Mehrschrittverfahren, die wir auf das Anfangswertproblem

$$\dot{x}(t) = 0, \qquad x(0) = 0 \tag{9.3}$$

anwenden. Da die rechte Seite in (9.1) wegen $f \equiv 0$ verschwindet, reicht es, die Koeffizienten a_i anzugeben. Wir betrachten nun die Verfahren mit

$$a_2 = 1,\ a_1 = -3,\ a_0 = 2 \quad \text{und} \quad \tilde{a}_2 = 1,\ \tilde{a}_1 = -3/2,\ \tilde{a}_0 = 1/2. \tag{9.4}$$

Man sieht leicht, dass beide Verfahren wegen $\sum a_i = 0$ bzw. $\sum \tilde{a}_i = 0$ konsistent sind. Für die DGL (9.3) ergeben sich daraus die Iterationsvorschriften

$$\tilde{x}(t_{i+1}) = -a_1\tilde{x}(t_i) - a_0\tilde{x}(t_{i-1}) = 3\tilde{x}(t_i) - 2\tilde{x}(t_{i-1}) \tag{9.5}$$

und

$$\tilde{x}(t_{i+1}) = -\tilde{a}_1\tilde{x}(t_i) - \tilde{a}_0\tilde{x}(t_{i-1}) = 3/2\tilde{x}(t_i) - 1/2\tilde{x}(t_{i-1}). \tag{9.6}$$

Man sieht leicht, dass beide Verfahren für exakte Startwerte $\tilde{x}(t_0) = \tilde{x}(t_1) = 0$ die exakte Lösung $\tilde{x}(t_i) \equiv 0$ liefern. Wenn wir den Startwert $\tilde{x}(t_1)$ allerdings leicht stören, so unterscheidet sich das Verhalten der beiden Verfahren erheblich. Abbildung 9.1 zeigt das unterschiedliche Verhalten für $\tilde{x}(t_0) = 0$ und den (nur ganz leicht gestörten Wert) $\tilde{x}(t_1) = 10^{-12}$.

Offenbar reproduziert das zweite Verfahren die exakte konstante Lösung trotz der kleinen Störung in $\tilde{x}(t_1)$ gut, während das erste Verfahren nach nur etwa 35 Schritten riesige Fehler produziert.

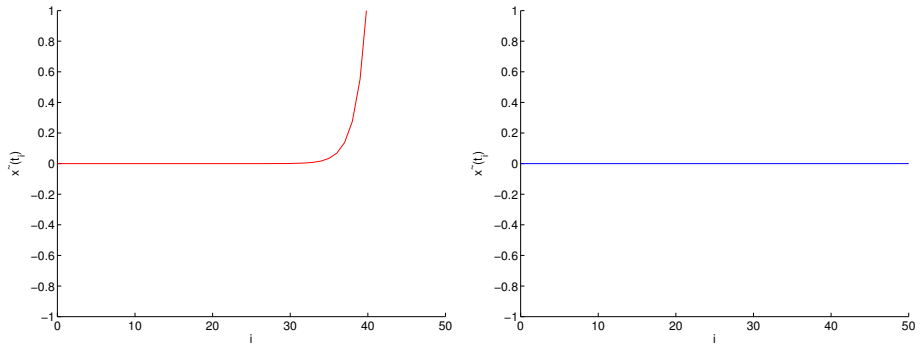

Abbildung 9.1: MSV (9.5) (links) und (9.6) (rechts) mit $\tilde{x}(t_0) = 0$, $\tilde{x}(t_1) = 10^{-12}$

Wir wollen nun untersuchen, warum dies so ist und wie man erkennen kann, ob ein Mehrschrittverfahren stabil gegenüber solchen kleinen Fehlern ist. Wegen der Linearität der linken Seite des Verfahrens genügt es, dazu das einfache Anfangswertproblem (9.3) zu betrachten (später im Beweis der Konvergenz werden wir genauer sehen, warum). Aus (9.1) folgt sofort, dass für (9.3) mit $\tilde{x}(t_0) = \ldots = \tilde{x}(t_{k-1}) = 0$ die Gleichung $\tilde{x} \equiv 0$ gilt, d.h. die exakte Lösung wird ohne Fehler reproduziert, falls die Startwerte exakt sind. Wie im obigen Beispiel betrachten wir nun den Fall, dass die bis zum Schritt $i^* \in \mathbb{N}$ erhaltenen Werte $\tilde{x}(t_i)$, $i = 0, \ldots, i^*$ durch Rechenfehler etwas gestört sind, wobei $\|\tilde{x}(t_i)\| \leq \varepsilon$ gelte. Für kleines $\varepsilon > 0$ sollten nun auch die nachfolgenden Werte $\tilde{x}(t_j)$, $j \geq i^*$ nur leicht gestört werden. Sicherlich kann man das nicht für alle Zeiten verlangen, aber doch zumindest auf vorgegebenen kompakten Zeitintervallen. Eine vernünftige Bedingung an das Verfahren für $f \equiv 0$ wäre also

$$\|\tilde{x}(t_i)\| \leq \varepsilon \text{ für } i = 0, \ldots, i^* \quad \Rightarrow \quad \|\tilde{x}(t_j)\| \leq C\varepsilon \text{ für alle } t_j \in [t_{i^*}, T].$$

Die wesentliche Beobachtung ist nun, dass zwar die Werte $\tilde{x}$ unabhängig von der Schrittweite h sind (dies ist gerade der entscheidende Unterschied zwischen der *linken* und der *rechten* Seite von (9.1)), nicht aber die Bedingung $t_j \in [t_{i^*}, T]$, die im Gegenteil stark von h abhängt: Je kleiner h wird, desto mehr Gitterpunkte t_j liegen in diesem Intervall. Da h beliebig klein werden kann, wird jeder t_j-Wert also für geeignetes h in $[t_{i^*}, T]$ liegen, weswegen man die Schranke $\|\tilde{x}(t_j)\| \leq C\varepsilon$ tatsächlich für alle $j \geq i^*$ fordern muss. Dies führt auf die folgende Definition, in der wir die jeweils die k Werte, die im Verfahren in Schritt i verwendet werden, gemeinsam betrachten.

Definition 9.8 Ein lineares Mehrschrittverfahren heißt *stabil*, falls ein $C > 0$ existiert, so dass für jeden Vektor $\tilde{x}^0 := (\tilde{x}(t_0), \ldots, \tilde{x}(t_{k-1}))^T$ von (reellen) Anfangswerten und alle $i \in \mathbb{N}$ die Ungleichung

$$\left\| \begin{pmatrix} \tilde{x}(t_i) \\ \vdots \\ \tilde{x}(t_{i+k-1}) \end{pmatrix} \right\| \leq C\|\tilde{x}^0\|$$

gilt. Hierbei ist die Folge $\tilde{x}(t_i)$ durch (9.1) bzw. (9.2) mit $\tilde{f}(t_i) = 0$ definiert, also kompakt geschrieben als

$$P_a(E)(\tilde{x})(t_i) = 0 \tag{9.7}$$

oder explizit ausgeschrieben als

$$\tilde{x}(t_{i+k}) = -\frac{a_{k-1}}{a_k}\tilde{x}(t_{i+k-1}) - \ldots - \frac{a_0}{a_k}\tilde{x}(t_i). \tag{9.8}$$

□

Wir werden nun ein einfaches Kriterium herleiten, das uns sagt, ob ein gegebenes Verfahren stabil ist. Hierzu stellen wir die Gleichung (9.8) zunächst in etwas anderer Form dar. Wir erinnern dazu an die linearen Differenzengleichungen (6.2), die durch eine Iterationsvorschrift der Form

$$x(t_{i+1}) = Ax(t_i)$$

mit einer Matrix $A \in \mathbb{R}^{k\times k}$ gegeben sind. Eine solche Gleichung heißt *stabil*, falls die Ungleichung

$$\|x(t_i)\| \leq C\|x(t_0)\|$$

für ein $C > 0$ und alle $i \in \mathbb{N}$ gilt. Das folgende Lemma zeigt, wie sich (9.8) als eine Matrix-Differenzengleichung schreiben lässt.

Lemma 9.9 Betrachte die lineare Differenzengleichung

$$x(t_{i+1}) = Ax(t_i) \tag{9.9}$$

mit

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & & \vdots \\ 0 & \cdots & \cdots & 0 & 1 & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \\ -\frac{a_0}{a_k} & -\frac{a_1}{a_k} & -\frac{a_2}{a_k} & \cdots & \cdots & -\frac{a_{k-1}}{a_k} \end{pmatrix} \in \mathbb{R}^{k\times k}.$$

Dann gilt für die Lösungen von (9.8) mit $\tilde{x}^0 = x(t_0)$ die Gleichung

$$\begin{pmatrix} \tilde{x}(t_i) \\ \vdots \\ \tilde{x}(t_{i+k-1}) \end{pmatrix} = x(t_i).$$

Insbesondere ist das Mehrschrittverfahren genau dann stabil, wenn (9.9) stabil ist.

Beweis: Ausschreiben der Differenzengleichung (9.9) liefert für alle $i \in \mathbb{N}_0$ die Gleichungen

$$x_j(t_{i+1}) = x_{j+1}(t_i) \text{ für } j = 1, \ldots, k-1$$

und

$$x_k(t_{i+1}) = -\frac{a_0}{a_k}x_1(t_i) - \ldots - \frac{a_{k-1}}{a_k}x_k(t_i)$$

Hiermit folgt die Behauptung per Induktion über i. □

Um ein Stabilitätskriterium für (9.1) zu erhalten, genügt uns also ein Stabilitätskriterium für (9.9). Hier hilft der folgende Satz, der eine Erweiterung von Satz 6.4(ii) darstellt.

Hierbei nennen wir einen Eigenwert *halbeinfach*, wenn seine algebraische und geometrische Vielfachheit übereinstimmen. Dies ist genau dann der Fall ist, wenn er eine einfache Nullstelle des Minimalpolynoms m_A ist. Das Minimalpolynom m_A ist dabei das Polynom mit minimalem Grad $p \geq 1$, für das $m_A(A) = 0$ gilt. Das Minimalpolynom m_A teilt immer das charakteristische Polynom χ_A.

Satz 9.10 Eine lineare Differenzengleichung $x(t_{i+1}) = Ax(t_i)$ ist genau dann stabil, wenn alle Eigenwerte λ_i von A die Bedingung $|\lambda_i| \leq 1$ erfüllen und alle Eigenwerte λ_i mit $|\lambda_i| = 1$ halbeinfach sind.

Beweis: Wir nummerieren die Eigenwerte gemäß der Ordnung $|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_d|$. Sei J die Jordan'sche Normalform von A mit Transformationsmatrix T, also $T^{-1}AT = J$. Wir schreiben kurz $x_i = x(t_i)$ und erinnern an die explizite Lösungsdarstellung $x_i = A^i x_0 = TJ^iT^{-1}x_0$. Wir schreiben $y_0 = T^{-1}x_0$ und $y_i = J^i y_0$.

Wir nehmen zunächst an, dass die Eigenwertbedingung erfüllt ist. Der Vektor y_0 lässt sich zerlegen in $y_0 = y_0^1 + y_0^2$ mit $y_0^1 = (y_1, \ldots, y_p, 0 \ldots, 0)^T$ und $y_0^2 = (0 \ldots, 0, y_{p+1}, \ldots, y_k)^T$, wobei $|\lambda_p| = 1$ und $|\lambda_{p+1}| < 1$ gilt. Mit $E_1 = \langle e_1, \ldots, e_p \rangle$ und $E_2 = \langle e_{p+1}, \ldots, e_k \rangle$ bezeichen wir die zugehörigen Unterräume. Für den Vektor y_i gilt nun

$$y_i = J^i y_0 = J^i(y_0^1 + y_0^2) = \underbrace{J^i y_0^1}_{=:y_i^1} + \underbrace{J^i y_0^2}_{=:y_i^2}.$$

Beachte, dass $y_i^1 \in E_1$ und $y_i^2 \in E_2$ liegt. Da die Einschränkung von J auf den Unterraum E_2 die Bedingung von Satz 6.4(ii) erfüllt (alle Eigenwerte im Betrag kleiner als 1), folgt die Existenz von $C_1 > 0$ und $\sigma > 0$ mit

$$\|y_i^2\| \leq C_1 \underbrace{e^{-\sigma(t_i - t_0)}}_{\leq 1} \|y_0^2\| \leq C_1 \|y_0^2\|.$$

Für y_i^1 gilt

$$\|y_i^1\| = \|J^i y_0^1\| \leq \|y_0\|,$$

wobei die letze Gleichung aus der Eigenwertstruktur folgt, denn J^i eingeschränkt auf E_1 ist wegen der Halbeinfachheit der Eigenwerte eine Diagonalmatrix mit Diagonalelementen λ_i mit $|\lambda_i| = 1$. Zusammen folgt also unter Verwendung der Definition der euklidischen Norm

$$\begin{aligned} \|x_i\| &\leq \|T\|\|y_i\| = \|T\|(\|y_i^1\| + \|y_i^2\|) \leq \|T\|(C_1\|y_0^2\| + |y_0^1\|) \\ &\leq (C_1 + 1)\|T\|\|y_0\| \leq (C_1 + 1)\|T\|\|T^{-1}\|\|x_0\| = C\|x_0\| \end{aligned}$$

für die Konstante $C = (C_1 + 1)\|T\|\|T^{-1}\|$.

Sei umgekehrt die Eigenwertbedingung nicht erfüllt. Falls ein Eigenwert λ_j mit $|\lambda_j| > 1$ existiert, so gilt für den zugehörigen Eigenvektor x_0

$$\|A^i x_0\| = |\lambda_j|^i \|x_0\| \to \infty \text{ für } i \to \infty,$$

was der Stabilität widerspricht. Falls ein nicht halbeinfacher Eigenwert λ_j mit $|\lambda_j| = 1$ existiert, so gibt es einen Eigenvektor x_0 sowie einen verallgemeinerten Eigenvektor x_1, für die die Gleichungen

$$Ax_0 = \lambda_j x_0 \quad \text{und} \quad Ax_1 = x_0 + \lambda_j x_1$$

gelten (dies folgt, da das Jordan-Kästchen zu dem nicht halbeinfachen Eigenwert λ_j eine 1 über der Diagonale besitzt). Per Induktion ergibt sich

$$A^i x_1 = i\lambda_j^{i-1} x_0 + \lambda_j^i x_1.$$

Da $\|\lambda_j^{i-1} x_0\| = \|x_0\|$ und $\|\lambda_j^i x_1\| = \|x_1\|$ (wegen $|\lambda_j| = 1$), folgt

$$\|A^i x_1\| \geq i\|x_0\| - \|x_1\| \to \infty \text{ für } i \to \infty,$$

was wiederum der Stabilität widerspricht. □

Zur Bestimmung der Stabilität genügt es also, die Eigenwerte der Matrix A zu bestimmen. Dies ist aber recht einfach, wie das folgende Lemma zeigt.

Lemma 9.11 Die Eigenwerte von A aus (9.9) sind genau die Nullstellen des Polynoms P_a aus (9.2). Ihre Vielfachheit im Minimalpolynom stimmt dabei mit ihrer Vielfachheit in P_a überein.

Beweis: Man rechnet nach, dass das charakteristische Polynom von A gerade durch

$$\chi_A(z) = z^k + \frac{a_{k-1}}{a_k} z^{k-1} + \ldots + \frac{a_1}{a_k} z + \frac{a_0}{a_k}$$

gegeben ist. Da die ersten Zeilen von $A^0, A^1, A^2, \ldots, A^{k-1}$ linear unabhängig sind (was aus der Verteilung der 0-Einträge leicht zu sehen ist), muss dies auch das Minimalpolynom m_A sein. Da $a_k \neq 0$ ist, stimmen die Nullstellen und Vielfachheiten von χ_A mit denen von

$$P_a(z) = a_0 + a_1 z + \ldots + a_k z^k = a_k \chi_A(z)$$

überein. □

Unsere Überlegungen führen nun direkt auf den folgenden Satz.

Satz 9.12 Ein lineares Mehrschrittverfahren (9.1) ist genau dann stabil, wenn alle Nullstellen λ_i von P_a die Bedingung $|\lambda_i| \leq 1$ erfüllen und alle Nullstellen λ_i von P_a mit $|\lambda_i| = 1$ einfache Nullstellen sind.

Beweis: Folgt sofort aus den vorangegangenen Aussagen.

Beachte, dass das Polynom P_a nach Bemerkung 9.7 für jedes konsistente Mehrschrittverfahren die Nullstelle 1 besitzen muss, also mindestens eine Nullstelle mit $|\lambda_i| = 1$ besitzt. Falls dies die einzige Nullstelle mit $|\lambda_i| = 1$ ist, nennt man das Verfahren *strikt stabil*. Falls es weitere Nullstellen λ_i mit $|\lambda_i| = 1$ gibt, so heißt das Verfahren *marginal stabil* oder *schwach stabil*. Obwohl sie theoretisch stabil sind, können solche Verfahren für bestimmte

Differentialgleichungen numerische Instabilitäten aufweisen, die z.B. durch Rundungsfehler hervorgerufen werden (vgl. das aktuelle Übungsblatt).

Für die explizite Mittelpunktregel z.B. berechnet man $P_a(z) = z^2 - 1$, das Polynom besitzt also die Nullstellen $z_{1/2} = \pm 1$ und ist damit stabil, genauer marginal stabil.

Für Einschrittverfahren, die als Spezialfall der Mehrschrittverfahren aufgefasst werden können, muss das Polynom P_a vom Grad $k = 1$ sein, denn nur x_{i+1} und x_i treten auf. Wegen $P_a(1) = 0$ kommt also nur $P_a(z) = z - 1$ in Frage, das als einzige Nullstelle $\lambda = 1$ besitzt. Also sind alle Einschrittverfahren stabil, weswegen wir die Stabilität dort nicht betrachten mussten. Dies ist auch der Grund, warum wir die Lipschitzbedingung für Einschrittverfahren nicht (wie in vielen Lehrbüchern) als Stabilitätsbedingung bezeichnet haben: Die Bedingungen bezeichnen verschiedene Sachverhalte, auch wenn sie den gleichen Zweck im Konvergenzbeweis erfüllen, nämlich zu garantieren, dass sich die in jedem Schritt gemachten lokalen Fehler nicht aufschaukeln können.

Auf Basis von Satz 9.12 können wir nun auch verstehen, warum die zwei Mehrschrittverfahren in dem einführenden Beispiel (9.4) so unterschiedliches Verhalten aufweisen. Für das Verfahren mit den Koeffizienten a_i ist das zugehörige Polynom $P_a(z) = z^2 - 3z + 2 = (z-1)(z-2)$, das gerade die Nullstellen 1 und 2 besitzt und das deswegen instabil ist. Für das zweite Verfahren mit den Koeffizienten $\tilde{a}_i$ gilt $P_{\tilde{a}}(z) = z^2 - 3/2z + 1/2 = (z-1)(z-1/2)$. Dieses Polynom hat die Nullstellen 1 und $1/2$, weswegen das Verfahren stabil ist.

9.3 Konvergenz

Ganz analog zu den Einschrittverfahren werden wir in diesem Abschnitt unser Hauptkonvergenzresultat

"Konsistenz (mit Ordnung p) + Stabilität $\Rightarrow$ Konvergenz (mit Ordnung p)"

formulieren und beweisen.

Zur Vorbereitung des Konvergenzsatzes benötigen wir noch ein Resultat über Lösungen von Differenzengleichungen, das im folgenden Lemma bereitgestellt wird.

Lemma 9.13 Betrachte die aus (9.7) hervorgehende *inhomogene Gleichung*

$$P_a(E)(y)(t_i) = c(t_i)$$

für eine Gitterfunktion $c : \mathcal{T} \to \mathbb{R}$ und ein stabiles Mehrschrittverfahren. Dann erfüllen die Lösungen dieser Gleichung die Abschätzung

$$|y(t_{i+k})| \leq C \left(\max_{l=0,\ldots,k-1} |y(t_l)| + \sum_{l=0}^{i} |c(t_l)| \right)$$

für eine geeignete Konstante $C > 0$.

Beweis: Für die vektorwertige Funktion $\hat{c}(t_i) = (0, \ldots, 0, c(t_i)/a_k)^T$ kann man die Gleichung in Matrixform

$$x(t_{i+1}) = Ax(t_i) + \hat{c}(t_i)$$

mit der Matrix A aus (9.9) und

$$\begin{pmatrix} y(t_i) \\ \vdots \\ y(t_{i+k-1}) \end{pmatrix} = x(t_i).$$

schreiben. Für diese Gleichung kann man die allgemeine Lösung per Induktion als

$$x(t_i) = A^i x(t_0) + \sum_{k=0}^{i-1} A^k \hat{c}(t_{i-k-1})$$

berechnen. Da A stabil ist, folgt aus der Definition der Matrixnorm sofort $\|A^k\|_\infty \leq \widetilde{C}$ für alle $k \in \mathbb{N}$ für ein $\widetilde{C} > 0$. Damit ergibt sich

$$\begin{aligned} |y(t_{i+k})| \;\leq\; \|x(t_{i+1})\|_\infty \;&\leq\; \widetilde{C}\|x(t_0)\|_\infty + \widetilde{C}\sum_{k=0}^{i} \|\hat{c}(t_{i-k})\|_\infty \\ &=\; \widetilde{C}\|x(t_0)\|_\infty + \widetilde{C}\sum_{k=0}^{i} |c(t_{i-k})/a_k| \\ &\leq\; \widetilde{C} \max_{l=0,\ldots,k-1} |y(t_l)| + \widetilde{C}/|a_k| \sum_{k=0}^{i} |c(t_i)|, \end{aligned}$$

also die Behauptung mit $C = \max\{\widetilde{C}, \widetilde{C}/|a_k|\}$. □

Wir kommen nun zum Konvergenzsatz. Wir formulieren das Resultat hier etwas schwächer als im Satz 2.7, da wir keine kompakte Menge von Anfangswerten, sondern nur einen einzelnen Anfangswert betrachten. Dies dient lediglich der Vermeidung allzu technischer Formulierungen in der Aussage und im Beweis des Satzes und hat keine prinzipiellen Gründe.

Satz 9.14 Gegeben sei ein Anfangswertproblem (1.1), (1.2) mit Anfangsbedingung (t_0, x_0) und p-mal stetig differenzierbarem Vektorfeld f. Gegeben seien weiterhin ein k-stufiges stabiles und konsistentes lineares Mehrschrittverfahren mit Ordnung $p \in \mathbb{N}$ und Näherungswerte $\tilde{x}(t_1), \ldots, \tilde{x}(t_{k-1})$ mit

$$\|\tilde{x}(t_i) - x(t_i; t_0, x_0)\| \leq \varepsilon_0 \;\text{ für }\; i = 1, \ldots, k-1.$$

Dann gilt für die durch das Verfahren auf dem Gitter $t_i = t_0 + hi$ zur Schrittweite h erzeugte Gitterfunktion $\tilde{x}(t_i)$ für alle Zeiten $t_i \in [t_0, T]$ und alle hinreichend kleinen $h > 0$ die Abschätzung

$$\|\tilde{x}(t_i) - x(t_i; t_0, x_0)\| \leq C(\varepsilon_0 + h^p)$$

für eine geeignete Konstante $C > 0$.

Beweis: Wir bezeichnen die exakte Lösung kurz mit $x(t)$ und wählen eine kompakte Umgebung $K \subset \mathbb{R} \times \mathbb{R}^n$ des exakten Lösungsgraphen $\{(t, x(t)) \,|\, t \in [t_0, T]\}$. Dann existiert ein $\delta_K > 0$, so dass für alle $t \in [t_0, T]$ die Folgerung $\|x - x(t)\| \leq \delta_K \Rightarrow (t, x) \in K$ gilt. Zudem existiert eine Konstante $L > 0$, so dass f auf K Lipschitz-stetig in x mit Konstante L ist. Mit N bezeichnen wir die größte ganze Zahl mit $N \leq (T - t_0)/h$.

Wie im Beweis von Satz 2.7 nehmen wir zunächst an, dass die numerische Lösung für alle $t_i \in [t_0, T]$ in K verläuft. Wir definieren den vektorwertigen Fehler als

$$\varepsilon_h(t_i) := x(t_i) - \tilde{x}(t_i).$$

Aus der Definition des Konsistenzfehlers folgt

$$P_a(E)(x)(t_i) = L(x, t_i, h) + hP_b(E)(\dot{x})(t_i) = L(x, t_i, h) + hP_b(E)(f)(t_i)$$

(wiederum mit der Abkürzung $f(t_i) = f(t_i, x(t_i))$). Von dieser Gleichung subtrahieren wir die Gleichung (9.2)

$$P_a(E)(\tilde{x})(t_i) = hP_b(E)(\tilde{f})(t_i).$$

Dies ergibt

$$P_a(E)(\varepsilon_h)(t_i) = L(x, t_i, h) + hP_b(E)\left(f(t_i) - \tilde{f}(t_i)\right).$$

Dies ist eine inhomogene (vektorwertige) Gleichung für ε_h. Indem wir Lemma 9.13 auf die einzelnen Komponenten von $\varepsilon_h(t_i)$ anwenden und $\|\varepsilon_h(t_j)\| \leq \varepsilon_0$ für $j = 0, \ldots, k-1$ ausnutzen, erhalten wir

$$\|\varepsilon_h(t_{i+k})\|_\infty \leq C\left(\varepsilon_0 + \sum_{l=0}^{i} \|L(x, t_l, h)\|_\infty + h\left\|P_b(E)\left(f(t_l) - \tilde{f}(t_l)\right)\right\|_\infty\right). \tag{9.10}$$

für alle $i = 0, \ldots, N - k$. Aus der Konsistenz folgt nun die Abschätzung

$$\|L(x, t_l, h)\|_\infty \leq C_p h^{p+1}$$

und aus der Lipschitz-Stetigkeit und der Definition von P_b und E folgt

$$\left\|P_b(E)\left(f(t_l) - \tilde{f}(t_l)\right)\right\|_\infty \leq L \sum_{m=0}^{k} |b_m| \, \|\varepsilon_h(t_{l+m})\|_\infty.$$

Setzen wir diese beiden Ungleichungen in (9.10) ein, so folgt

$$\begin{aligned}
\|\varepsilon_h(t_{i+k})\|_\infty &\leq C\left(\varepsilon_0 + \underbrace{\sum_{l=0}^{i} C_p h^{p+1}}_{\leq N C_p h^{p+1} \leq (T-t_0) C_p h^p} + h \sum_{l=0}^{i} L \sum_{m=0}^{k} |b_m| \, \|\varepsilon_h(t_{l+m})\|_\infty\right) \\
&\leq \widehat{C}_1 \varepsilon_0 + \widehat{C}_2 h^p + h\widehat{C}_3 \sum_{l=0}^{i+k} \|\varepsilon_h(t_l)\|_\infty
\end{aligned}$$

für geeignete Konstanten $\widehat{C}_q > 0$. Beschränken wir nun die Schrittweite durch $h \leq 1/(2\widehat{C}_3)$, so können wir nach $\|\varepsilon_h(t_{i+k})\|_\infty$ auflösen und erhalten mit $j = i + k$ die Ungleichung

$$\|\varepsilon_h(t_j)\|_\infty \leq C_1\varepsilon_0 + C_2h^p + hC_3 \sum_{l=0}^{j-1} \|\varepsilon_h(t_l)\|_\infty$$

mit $C_q = 2\widehat{C}_q$. Beachte, dass diese Ungleichung auch für $j = 1, \ldots, k-1$ stimmt wenn wir o.B.d.A. $C_1 \geq 1$ annehmen.

Per Induktion (wie im Beweis von Satz 2.7) ergibt sich daraus die Abschätzung

$$\|\varepsilon_h(t_j)\|_\infty \leq (C_1\varepsilon_0 + C_2h^p)e^{jhC_3} = (C_1\varepsilon_0 + C_2h^p)e^{(t_j - t_0)C_3},$$

also die gewünschte Behauptung.

Der induktive Beweis, dass die numerische Lösung für hinreichend kleine $h > 0$ tatsächlich in K liegt, verläuft für explizite Verfahren ganz analog zum Beweis von Satz 2.7. Für implizite Verfahren muss dieser Beweis in jedem Schritt um ein Fixpunktargument erweitert werden, das wir hier aber nicht ausführen wollen. □

Bemerkung 9.15 (i) Das Konvergenzresultat zeigt insbesondere, wie die Startwerte $\tilde{x}(t_1)$, $\ldots$, $\tilde{x}(t_{k-1})$ bestimmt werden müssen. Um für das Mehrschrittverfahren die Konvergenzordnung p zu garantieren, müssen diese ebenfalls mit der Genauigkeit $O(h^p)$ bestimmt werden. Da es sich hier nur um endlich viele Werte handelt, deren Anzahl unabhängig von h ist, genügt es dazu, ein Einschrittverfahren mit Konsistenzordnung $p - 1$ zu verwenden. Der Beweis von Satz 2.7 zeigt nämlich, dass die ersten k Werte durch ein solches Verfahren immer die Genauigkeit $O(h^p)$ besitzen, falls k unabhängig von h ist. Der "Verlust" einer Ordnung beim Übergang von der Konsistenz- zur Konvergenzordnung ergibt sich erst dadurch, dass die Anzahl der nötigen Schritte von h abhängt.

(ii) Eine genauere Analyse zeigt, dass sogar die stärkere Aussage

$$\text{Konsistenz} + \text{Stabilität} \Leftrightarrow \text{Konvergenz}$$

gilt. Konsistenz und Stabilität sind also *notwendig und hinreichend* für die Konvergenz eines Verfahrens. □

9.4 Verfahren in der Praxis

In der Praxis haben sich zwei Klassen von Mehrschrittverfahren durchgesetzt. Beide Klassen haben gewisse Eigenschaften, die sie für gewisse Problemklassen besonders auszeichnen.

Adams-Verfahren

Historisch haben sich die Adams-Verfahren aus Quadraturformeln zur numerischen Integration entwickelt. Wir motivieren die Herleitung hier allerdings aus ihrer besonderen Eigenschaft, die ihre Vorteile in der Praxis begründet.

Wir haben gesehen, dass das Polynom P_a eines Mehrschrittverfahrens stabil sein muss, also — abgesehen von einer Nullstelle $= 1$ — nur Nullstellen mit Betrag $|\lambda_i| \leq 1$ besitzen darf. Je kleiner die Eigenwerte dabei im Betrag sind, desto "stabiler" wird das Verfahren. Bei den Adams-Verfahren wählt man P_a deswegen so, dass neben der $\lambda_1 = 1$ nur Nullstellen $\lambda_i = 0$ auftreten, also

$$P_a(z) = z^{k-1}(z-1) = z^k - z^{k-1}$$

ist. Beachte, dass damit auf der linken Seite von (9.1) nur die Werte $\tilde{x}(t_{i+k})$ und $\tilde{x}(t_{i+k-1})$ stehen bleiben.

Für jede beliebige Stufenanzahl k liefert Satz 9.6(iii) nun ein Gleichungssystem mit genau zwei Lösungen, nämlich

- genau ein *explizites* Adams-Verfahren der Konsistenzordnung $p = k$ (auch *Adams-Bashforth-Verfahren* genannt)
- genau ein *implizites* Adams-Verfahren der Konsistenzordnung $p = k + 1$ (auch *Adams-Moulton-Verfahren* genannt)

Z.B. lauten die Polynome P_b der ersten vier expliziten Adams-Verfahren

$$\begin{aligned} k=1: \quad P_b(z) &= 1 \\ k=2: \quad P_b(z) &= (3z-1)/2 \\ k=3: \quad P_b(z) &= (23z^2 - 16z + 5)/12 \\ k=4: \quad P_b(z) &= (55z^3 - 59z^2 + 37z - 9)/24 \end{aligned}$$

Interessanterweise ist das explizite Adams-Verfahren für $k = 1$ gerade das explizite Euler-Verfahren.

Für diese Verfahren hat sich ein Algorithmus durchgesetzt, der als *Prädiktor-Korrektor-Verfahren* bezeichnet wird. Ein Schritt dieses Algorithmus verläuft wie folgt:

Algorithmus 9.16 Prädiktor-Korrektor-Verfahren Gegeben seien das explizite und das implizite Adams-Verfahren der Stufe k.

1) Prädiktor-Schritt: Berechne $\tilde{x}(t_{i+k})$ mit dem expliziten Adams-Verfahren

2) Korrektor-Schritt: Führe *einen Schritt* der Fixpunktiteration zur Lösung des impliziten Verfahrens mit Startwert $\tilde{x}(t_{i+k})$ durch. □

Der Prädiktor-Schritt liefert hierbei eine Approximation mit Konsistenzfehler $O(h^{k+1})$. Für hinreichend kleine Schrittweite h ist die Kontraktionskonstante der Fixpunktiteration gleich Ch für ein $C > 0$. Also liefert der eine Iterationsschritt eine Approximation mit dem Konsistenzfehler

$$\frac{Ch}{1-Ch} O(h^{k+1}) = O(h^{k+2}).$$

Das Prädiktor-Korrektor-Verfahren besitzt also die Konsistenzordnung $k + 1$.

BDF-Verfahren

Obwohl die Familie der Adams-Verfahren implizite Verfahren enthält, sind diese (wegen ihrer recht kleinen Stabilitätsgebiete $\mathcal{S}$) schlecht für steife DGL geeignet.

Tatsächlich kann man beweisen, dass kein Mehrschrittverfahren der Ordnung $p > 2$ A-stabil ist. Die zur Lösung steifer DGL so nützliche Eigenschaft $\mathbb{C}^- \subseteq \mathcal{S}$ lässt sich also nicht erreichen. Es gibt allerdings eine Klasse impliziter Mehrschrittverfahren, die zumindest unendlich große Stabilitätsgebiete $\mathcal{S}$ besitzt, und die deswegen zur Lösung steifer DGL recht gut geeignet sind.

Dies ist die Klasse der BDF-Verfahren (BDF="backwards difference"). Hier wird gefordert, dass ein Kegel der Form $\{a + ib \in \mathbb{C}^- \,|\, |b| \leq c|a|\}$ für ein $c > 0$ in $\mathcal{S}$ liegt. Dies führt auf die Bedingung

$$P_b(z) = z^k.$$

Wiederum mit Satz 9.6(iii) erhält man dann Bedingungen, nun an die Koeffizienten von P_a, die die Konstruktion von Verfahren beliebig hoher Konsistenzordnung $p = k$ ermöglichen. Die ersten vier Polynome lauten hier

$$\begin{aligned}
k = 1: \quad & P_a(z) &&= z - 1 \\
k = 2: \quad & P_a(z) &&= \frac{3}{2}z^2 - 2z + \frac{1}{2} \\
k = 3: \quad & P_a(z) &&= \frac{11}{6}z^3 - 3z^2 + \frac{3}{2}z - \frac{1}{3} \\
k = 4: \quad & P_a(z) &&= \frac{25}{12}z^4 - 4z^3 + 3z^2 - \frac{4}{3}z + \frac{1}{4}
\end{aligned}$$

Für $k = 1$ ergibt sich gerade das implizite Euler-Verfahren. Die BDF-Verfahren sind allerdings nur bis $p = k = 6$ praktikabel, da die Verfahren für höhere Stufenzahlen instabil werden (beachte, dass die Bedingungen aus 9.6(iii) nur die Konsistenz, nicht aber die Stabilität sicher stellen).

Schrittweitensteuerung

Zuletzt wollen wir ganz kurz die Schrittweitensteuerung für Mehrschrittverfahren diskutieren. Sicherlich kann man die Fehlerschätzertheorie für Einschrittverfahren eins zu eins auf Mehrschrittverfahren übertragen und ebenso wie dort neue Schrittweiten berechnen und damit die Schrittweite adaptiv steuern.

Es ergibt sich aber ein technisches Problem, da die Schrittweite im aktuellen Schritt mit den Schrittweiten der $k-1$ vorangegangenen Schritte übereinstimmen muss, weil ansonsten die definierende Gleichung (9.1) nicht sinnvoll ausgewertet werden kann.

Abhilfe schafft hier eine alternative Darstellung, die wir für die Adams-Verfahren illustrieren: Wenn die Werte $\tilde{x}(t_i), \ldots, \tilde{x}(t_{i+k-1})$ eine Approximation der Ordnung p an die differenzierbare Funktion $x(t)$ in den Punkten $t_i, \ldots, t_{i+k-1}$ darstellen, so ist das durch die Daten

$$(t_i, \tilde{x}(t_i)), \ldots, (t_{i+k-1}, \tilde{x}(t_{i+k-1}))$$

definierte Interpolationspolynom $q(t)$ eine Approximation der Ordnung p an $x(t)$, und zwar für alle t aus einem vorgegebenen kompakten Intervall.

Für die Adams-Verfahren kann man nachrechnen, dass die Verfahren mit diesem Interpolationspolynom q gerade als

$$\tilde{x}(t_{i+k}) = \tilde{x}(t_{i+k-1}) + \int_{t_{i+k-1}}^{t_{i+k}} q(t)dt$$

gegeben sind (zum Beweis betrachtet man die Lagrange-Polynomdarstellung von q und integriert). Diese Gleichung ist nun unabhängig von der zur Berechnung von q verwendeten Schrittweite und kann daher für variable Schrittweiten ausgewertet werden.

Für die BDF-Verfahren ist ein ähnlicher Trick möglich, so dass auch hier die Schrittweitensteuerung anwendbar ist.

In MATLAB finden sich schrittweitengesteuerte Adams-Verfahren unter dem Namen `ode113` und BDF-Verfahren unter dem Namen `ode15s`.

Kapitel 10

Typen von partiellen Differentialgleichungen

Wir beginnen nun mit der Betrachtung numerischer Methoden für partielle Differentialgleichungen. Der Unterschied zu den bisher betrachteten gewöhnlichen Differentialgleichungen besteht darin, dass die unabhängige Variable nun i.A. nicht mehr wie bisher das t eindimensional ist. Die Variable ist also mehrdimensional, z.B. $x \in \mathbb{R}^2$ oder $(t, x) \in \mathbb{R} \times \mathbb{R}^2$. Dies hat zur Folge, dass die Gleichung Ableitungen nach den unterschiedlichen Komponenten der Variablen, z.B. x_1 und x_2 oder t, x_1 und x_2 enthält, also partielle Ableitungen. Daraus ergibt sich der Name partielle Differentialgleichung.

Im Gegensatz zu den gewöhnlichen Differentialgleichungen, die sich stets in eine Standardform umformen lassen, auf die dann die numerischen Methoden angewendet werden können, gibt es bei den partiellen Differentialgleichungen eine ganze Reihe von verschiedenen Typen, die sich nicht ineinander umformen lassen und die sowohl in der Theorie als auch in der Numerik mit unterschiedlichen Methoden behandelt werden müssen. Wir beschreiben im Folgenden einige wichtige Klassen und beginnen mit linearen Gleichungen zweiter Ordnung der Form

$$\sum_{i,j=1}^{n} a_{ij} u_{x_i x_j} + \sum_{i=1}^{n} b_i u_{x_i} + cu = 0 \tag{10.1}$$

mit symmetrischer Matrix $A = (a_{ij})$. Seien $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$ die Eigenwerte von A. Dann nennt man (vgl. [14, Definition 1.1]) die Gleichung

- elliptisch, falls alle $\lambda_i > 0$ oder alle $\lambda_i < 0$ sind
- hyperbolisch, falls alle $\lambda_i \neq 0$ sind und genau ein λ_i ein anderes Vorzeichen als die anderen λ_j, $j \neq i$ besitzt
- parabolisch, falls genau ein Eigenwert 0 ist und alle anderen identisches Vorzeichen haben.

Der "geometrische" Grund für diese Bezeichnungen lässt sich am besten im Fall $n = 2$ erklären und wenn wir annehmen, dass A in Diagonalform vorliegt, also $A = \mathrm{diag}(\lambda_1, \lambda_2)$.

In diesem Fall können wir die Menge der Punkte $x = (x_1, x_2)^T$ betrachten mit

$$\langle x, Ax \rangle = c$$

mit $c \neq 0$. Im elliptischen Fall ist diese Menge beschrieben durch eine Gleichung der Form $\lambda_1 x_1^2 + \lambda_2 x_2^2 = c$. Diese Gleichung beschreibt eine Ellipse, falls das Vorzeichen von c gleich dem von λ_1 und λ_2 ist.

Im hyperbolischen Fall gilt $ax_1^2 - bx_2^2 = c$ mit $a, b > 0$ oder $a, b < 0$. Diese Gleichung beschreibt eine Hyperbel: nehmen wir o.B.d.A. $a, b > 0$ an, so gilt für die Koordinaten $z_1 = \sqrt{a}x_1 + \sqrt{b}x_2$ und $z_2 = \sqrt{a}x_1 - \sqrt{b}x_2$

$$c = ax_1^2 - bx_2^2 = z_1 z_2 \quad \Leftrightarrow \quad z_1 = \frac{c}{z_2}.$$

Im parabolischen Fall müssen wir zur geometrischen Interpretation den b-Term hinzunehmen. Dabei nehmen wir o.B.d.A. an, dass $\lambda_1 \neq 0$ und $\lambda_2 = 0$ gilt. Nehmen wir dann zusätzlich an, dass $b_2 \neq 0$ ist, so erhalten wir aus $\langle x, Ax + b \rangle = c$ die Gleichung $x_1^2 + b_1 x_1 + b_2 x_2 - c = 0$, die gerade eine Parabel beschreibt.

Diese Form der Klassifizierung wird im allgemeinen Sprachgebrauch oft benutzt und ist für eine erste grobe Einteilung oft sinnvoll. Allerdings bestimmen viele weitere Faktoren über das Verhalten der Lösungen und die Wahl eines geeigneten numerischen Verfahrens. Zudem gibt es viele Gleichungen, die nicht in das obige Schema fallen.

In den folgenden Abschnitten geben wir einige Erweiterungen, Spezialfälle und Beispiele für diese Gleichungen an. Mit einigen von diesen werden wir uns im Folgenden befassen.

10.1 Elliptische Gleichungen

Elliptische partielle Differentialgleichung kann man auch mit ortsabhängigen Koeffizienten betrachten. Ein Beispiel einer solchen verallgemeinerten Gleichung ist dann

$$-\mathrm{div}(\mathcal{A}(x)\nabla u) + c(x)u - f(x) = 0,$$

wobei $u : \Omega \to \mathbb{R}$ die gesuchte Funktion ist. Dabei ist $\Omega \subset \mathbb{R}^n$ offen und beschränkt und wir betrachten die Koeffizientenfunktionen $\mathcal{A} : \Omega \to \mathbb{R}^{n \times n}$, $c, f : \Omega \to \mathbb{R}$. Die Symbole

$$\nabla v = \begin{pmatrix} \frac{\partial v}{\partial x_1} \\ \vdots \\ \frac{\partial v}{\partial x_n} \end{pmatrix} \quad \text{und} \quad \mathrm{div}(v) = \sum_{i=1}^{n} \frac{\partial v}{\partial x_i}$$

bezeichnen den Gradienten und die Divergenz einer Funktion $v : \Omega \to \mathbb{R}$.

Damit die Gleichung tatsächlich elliptisch ist, müssen die Koeffizientenfunktionen bestimmte Abschätzungen erfüllen, die wir im Abschnitt 12.2 präzise definieren werden. Dabei verwenden wir eine funktionalanalytische Definition der Elliptizität — oft V-Elliptizität genannt —, die für unsere numerischen Zwecke günstiger ist. Hinreichend für diese Form der Elliptizität ist z.B., dass $\mathcal{A}(x)$ positiv definit und $c(x) > 0$ ist für all $x \in \Omega$, beides gleichmäßig für alle $x \in \Omega$, vgl. Satz 12.4. Bei Dirichlet-Randbedingungen (siehe Abschnitt

10.4) reicht $c(x) \geq 0$ aus, vgl. Satz 12.6. In beiden Fällen sieht man leicht, dass die Gleichung auch elliptisch im Sinne der nach (10.1) gegebenen Definition ist.

Das klassische Beispiel für eine elliptische Gleichung ist die Poisson- oder Wärmeleitungsgleichung

$$\Delta u = h(x),$$

wobei Δ den Laplace-Operator $\Delta v = \sum_{i=1}^{n} \partial^2/\partial x_i^2 v$ bezeichnet. Schreiben wir diese als

$$-\Delta u + h(x) = 0,$$

so ist sie von der obigen Form mit $\mathcal{A} = \mathrm{Id}$, $c \equiv 0$ und $f = -h$. Im Spezialfall $n = 1$ lautet die Gleichung

$$u_{xx} = h(x),$$

wobei u_{xx} kurz für $\partial^2/\partial x^2 u$ steht. Die Poissson-Gleichung beschreibt z.B. die Wärmeverteilung in einem Gebiet Ω, wobei h die Wärmezufuhr (= Heizung oder Kühlung) in jedem Punkt $x \in \Omega$ beschreibt. Im Fall $n = 1$ ist Ω dabei ein eindimensionales Gebiet, das man sich als einen Stab vorstellen kann. Im Fall der Wärmeleitungsgleichung können z.B. vom Ort x abhängige Wärmekoeffizienten (z.B. für unterschiedliche Materialien) durch die von x abhängige Matrix $\mathcal{A}(x)$ modelliert werden.

10.2 Parabolische Gleichungen

Parabolische Gleichungen bestehen aus einem elliptischen Anteil (dem Unterraum, der zu den $n-1$ Eigenwerten mit gleichem Vorzeichen gehört) und einem weiteren eindimensionalen Unterraum. In vielen Anwendungen ist das Koordinatensystem an diese Unterräume angepasst, wobei die eindimensionale Variable als t und die restlichen Variablen als Vektor x bezeichnet werden. Auf diese Weise kann man jede elliptische Gleichung zu einer parabolischen Gleichung verallgemeinern, indem man die 0 auf der rechten Seite durch $u_t := du/dt$ oder durch $-u_t$ ersetzt. So kann man z.B. aus der obigen elliptischen Gleichung mit zeitvarianten Koeffizienten eine parabolische Gleichung der Form

$$u_t = \mathrm{div}(\mathcal{A}(x)\nabla u) - c(x)u + f(x)$$

machen. Dabei ist die gesuchte Funktion u nun eine Abbildung von $[t_0, T) \times \Omega$ nach $\mathbb{R}$. Wie bei den gewöhnlichen Differentialgleichungen steht t für die Zeit und x für den Ort, im Unterschied zu dort suchen wir jetzt aber keine Funktion x von t sondern eine Funktion, die von t und x gleichermaßen abhängt. Die Variablen t *und* x spielen also jetzt die Rolle der Variablen t bei den gewöhnlichen Differentialgleichungen und die Funktion $u(t, x)$ spielt die Rolle der Funktion $x(t)$.

Im Fall der Wärmeleitungsgleichung mit $h \equiv 0$ ergibt sich

$$u_t = \Delta u$$

bzw. im Eindimensionalen

$$u_t = u_{xx}.$$

Bringt man diese Gleichung durch Umbenennung von t in x_1 und x in x_2 in die Form (10.1) so prüft man leicht nach, dass sie parabolisch im obigen Sinne ist. Diese Gleichung gibt die zeitliche Änderung der Wärmeverteilung in dem Gebiet Ω an. Die Lösung der zugehörigen elliptischen Gleichung ist dann gerade ein Gleichgewicht der parabolischen Gleichung, im Fall der Wärmeleitungsgleichung also eine stationäre Wärmeverteilung.

10.3 Hyperbolische Gleichungen

Die Numerik hyperbolischer Gleichungen werden wir in dieser Vorlesung aus Zeitgründen nicht behandeln. Wir wollen nur der Vollständigkeit halber einen typischen Vertreter dieser Klasse betrachten, nämlich die Wellengleichung. Diese ist für Wellen in einem eindimensionalen Gebiet (wodurch z.B. eine schwingende Gitarrensaite modelliert wird) gegeben durch

$$u_{tt} = c^2 u_{xx}.$$

Bringt man die Terme auf eine Seite und bezeichnet die Variablen mit x_1 und x_2 statt t und x, so kann man die Gleichung in der Form (10.1) schreiben. Dann sieht man leicht, dass die Hyperbolizitätsbedingung erfüllt ist. In höheren Dimensionen ist die Wellengleichung gegeben durch

$$u_{tt} = c^2 \Delta u.$$

10.4 Anfangs- und Randbedingungen

Damit es eine eindeutige Lösung für eine partielle Differentialgleichung gibt, muss geklärt werden, wie sich die Lösung am Rand des Gebiets Ω verhält. Wir unterscheiden dabei zwischen räumlichen Randbedingungen am Rand von Ω und zeitlichen Randbedingungen am Rand von $[t_0, T)$. Hierbei wird — wie bei den gewöhnlichen Differentialgleichungen — für die zeitliche Randbedingung oft eine Anfangsbedingung zur Zeit t_0 gefordert, üblicherweise in der Form

$$u(t_0, x) = u_0(x)$$

für eine vorgegebene Funktion $u_0 : \Omega \to \mathbb{R}$. Zur Endzeit T wird in diesem Fall keine Bedingung gefordert.

Für die räumlichen Variablen werden an allen Punkten x im Rand $\partial\Omega$ Bedingungen an u festgelegt. Diese können von unterschiedlicher Form sein.

- Dirichlet-Randbedingungen
$$u|_{\partial\Omega} = \phi$$
für eine vorgegebene Funktion $\phi : \partial\Omega \to \mathbb{R}$.

- (Homogene) Neumann-Randbedingungen
$$\left.\frac{\partial u}{\partial n}\right|_{\partial\Omega} := Du\, n = 0,$$
wobei $n(x)$ mit $n : \partial\Omega \to \mathbb{R}^n$ einen äußeren Normalenvektor an den Rand $\partial\Omega$ im Punkt x bezeichnet.

- Robin-Randbedingungen

$$\frac{\partial u}{\partial n}(t,x) + \alpha(x)u(t,x) = \beta(x),$$

für alle $x \in \partial\Omega$ wobei $\alpha, \beta : \partial\Omega \to \mathbb{R}$ und n wieder die äußere Normale bezeichnet.

Für elliptische Gleichungen werden wir in Kapitel 12 eine Methode kennen lernen, um Existenz- und Eindeutigkeit für eine Klasse von Gleichungen mit geegneten Randbedingungen zu garantieren.

Für die Wärmeleitungsgleichung entspricht die Anfangsbedingung der zur Zeit $t = 0$ bestehenden Wärmeverteilung. Die Dirichlet-Randbedingung gibt an den Randpunkten eine feste Temperatur vor, die Neumann-Randbedingung legt fest, dass die Ableitung der Wärmeverteilung in Richtung des Randes gleich 0 ist, was bedeutet, dass keine Wärmeazstausch über den Rand hinweg stattfindet (also perfekte Isolation). Die Robin-Randbedingung ist für die Wärmeleitungsgleichung das realistischste Modell, da man damit das Newton'sche Abkühlungsgesetz modellieren kann (der Temperaturaustausch am Rand hängt vom Unterschied zur Umgebungstemperatur und einem Wärmeübergangskoeffizienten ab).

Kapitel 11

Finite Differenzen für die Wärmeleitungsgleichung

Zur numerischen Lösung partieller Differentialgleichungen gibt es zwei grundsätzliche Methoden: die finiten Differenzen und die finiten Elemente. Die finiten Elemente sind die vielseitigere Methode, die wir daher ab dem nachfolgenden Kapitel ausführlicher behandeln wollen. In vielen einfachen Anwendungen funktionieren finite Differenzen aber auch sehr gut. Zudem sind sie den Methoden für gewöhnliche Differentialgleichungen ähnlicher, weswegen wir diese Methode zuerst behandeln.

Wir machen dies am Beispiel eines Modellproblems, nämlich der parabolischen Wärmeleitungsgleichung

$$u_t(t,x) - u_{xx}(t,x) = 0 \tag{11.1}$$

auf $[0,T) \times \Omega$ mit $\Omega = (-a,a) \subset \mathbb{R}$. In $t = 0$ setzen wir die Anfangsbedingung

$$u(0,x) = u_0(x)$$

und am Rand $\partial\Omega = \{-a,a\}$ die Dirichlet-Randbedingungen

$$u(t,-a) = g_1(t) \quad \text{und} \quad u(t,a) = g_2(t) \qquad \text{für } t \in [0,T]$$

fest.

Die Methode kann wegen ihrer konzeptionellen Einfachheit auf viele andere Gleichungen und Randbedingungen angepasst werden.

11.1 Grundidee der Finiten Differenzen

Wir beschreiben nun die Methode der finiten Differenzen für die Wärmeleitungsgleichung (11.1). Die Lösung $u(t,x)$ der Wärmeleitungsgleichung (11.1) ist durch Ihre Ableitungen an unendlich vielen Punkten (t,x) charakterisiert. Die Methode der finiten Differenzen besteht nun darin, diese unendlich vielen Ableitungen durch eine endliche (finite) Anzahl von Differenzenquotienten (Differenzen) zu ersetzen und so eine approximative Lösung der Gleichung zu erhalten.

Die verwendeten Differenzenquotienten sind dabei für $h, s > 0$

$$u_{xx}(t,x) \approx \frac{1}{h^2}(u(t,x+h) - 2u(t,x) + u(t,x-h)) =: \Delta_x^2 u(t,x)$$

für die 2. Ableitung nach x und entweder die Vorwärtsdifferenz

$$u_t(t,x) \approx \frac{1}{s}(u(t+s,x) - u(t,x)) =: \Delta_t^v u(t,x)$$

oder die Rückwärtsdifferenz

$$u_t(t,x) \approx \frac{1}{s}(u(t,x) - u(t-s,x)) =: \Delta_t^r u(t,x)$$

für die 1. Ableitung nach t.

Bemerkung 11.1 Verwendet man die Vorwärtsdifferenz zur Approximation der linken Seite einer gewöhnlichen Differentialgeichung, so erhält man gerade das explizite Euler-Verfahren. Verwendet man die Rückwärtsdifferenz, so ergibt sich das implizite Euler-Verfahren. □

Für die Differenzenquotienten gilt das folgende Lemma. Darin (und allgemein in diesem Kapitel) verwenden wir das Landau-Symbol O wie bei den gewöhnlichen Differentialgleichungen: Für einen beliebigen reellen Ausdruck $y(t,x,z)$ mit $t,x,z \in \mathbb{R}$, $t \geq 0$, $z > 0$, schreiben wir $y(t,x,z) = O(z^p)$, falls für jede kompakte Menge $K \subset [0,\infty) \times \mathbb{R}$ eine Konstante $C_K > 0$ existiert mit $|y(t,x,z)| \leq C_K z^p$ für alle hinreichend kleinen z und alle $(t,x) \in K$.

Lemma 11.2 Für $u \in C^4$ und $h, s > 0$ gilt

$$u_{xx}(t,x) = \Delta_x^2 u(t,x) + O(h^2),$$

$$u_t(t,x) = \Delta_t^v u(t,x) + O(s)$$

und

$$u_t(t,x) = \Delta_t^r u(t,x) + O(s).$$

Beweis: Nach der Taylor-Entwicklung gelten die beiden Gleichungen

$$\begin{aligned} u(t,x+h) &= u(t,x) + u_x(t,x)h + u_{xx}(t,x)\frac{h^2}{2} + u_{xxx}(t,x)\frac{h^3}{6} + O(h^4) \\ u(t,x-h) &= u(t,x) - u_x(t,x)h + u_{xx}(t,x)\frac{h^2}{2} - u_{xxx}(t,x)\frac{h^3}{6} + O(h^4). \end{aligned}$$

Addition der Gleichungen und Division durch h^2 ergibt

$$\frac{1}{h^2}u(t,x+h) + \frac{1}{h^2}u(t,x-h) = \frac{1}{h^2}2u(t,x) + u_{xx}(t,x) + \frac{1}{h^2}O(h^4)$$

und wegen $O(h^4)/h^2 = O(h^2)$ folgt die erste Gleichung.

Die zweite und dritte Gleichung folgen direkt aus der Taylor-Entwicklung in t. Die geforderte Existenz der Konstanten C_K folgt dabei in allen Fällen aus der Stetigkeit der Ableitungen, wodurch die entsprechenden Faktoren in den Taylor-Restgliedern stetig sind und damit auf kompakten Mengen beschränkt sind. □

Da die Lösung der Wärmeleitungsgleichung C^∞ ist (vgl. [5, Section 2.3, Theorem 1]), erhalten wir damit das folgende Korollar.

Korollar 11.3 Für die Lösung $u(t,x)$ der Wärmeleitungsgleichung gilt

$$\Delta_t^v u(t,x) - \Delta_x^2 u(t,x) = O(s+h^2) \tag{11.2}$$

und

$$\Delta_t^r u(t,x) - \Delta_x^2 u(t,x) = O(s+h^2) \tag{11.3}$$

sowie für jedes $\theta \in [0,1]$

$$\Delta_t^v u(t,x) - (1-\theta)\Delta_x^2 u(t,x) - \theta\Delta_x^2 u(t+s,x) = O(s+h^2). \tag{11.4}$$

Beweis: Die Gleichungen (11.2) und (11.3) folgen sofort durch Einsetzen der Gleichungen aus Lemma 11.2 in Gleichung (11.1) und $O(h^2) + O(s) = O(h^2+s)$. Zum Beweis von Gleichung (11.4) betrachten wir (11.3) für $t+s$ an Stelle von t, also

$$\Delta_t^r u(t+s,x) - \Delta_x^2 u(t+s,x) = O(s+h^2).$$

Wegen $\Delta_t^r u(t+s,x) = \Delta_t^v u(t,s)$ folgt

$$\Delta_t^v u(t,x) - \Delta_x^2 u(t+s,x) = O(s+h^2).$$

Multiplizieren wir diese Gleichung mit θ und addieren (11.2) multipliziert mit $1-\theta$, so erhalten wir gerade (11.4). □

Für $\theta = 0$ ist (11.4) gerade (11.2), für $\theta = 1$ gerade (bis auf Verschiebung der Zeitvariablen t) (11.3). Wir werden später sehen, dass man für jedes $\theta \in [0,1]$ aus (11.4) sinnvolle numerische Schemata ableiten kann, allerdings mit unterschiedlichen Eigenschaften.

Auf dem Gebiet Ω definieren wir nun ein regelmäßiges Gitter aus $(M+1)\cdot(N+1)$ Punkten (t_j, x_i), $j = 0,\dots,M$, $i = 0,\dots,N$. Dazu setzen wir $s := T/M$ und $h := 2a/N$ und definieren

$$t_j := sj, \;\; j = 0,\dots,M \quad \text{und} \quad x_i := -a + hi, \;\; i = 0,\dots,N.$$

Beachte, dass damit $t_{j+1} = t_j + s$ und $x_{i+1} = x_i + h$ sowie $x_{i-1} = x_i - h$ gilt.

Bezeichnen wir nun die Werte der exakten Lösung u in den Gitterpunkten kurz mit

$$u_i^j := u(t_j, x_i),$$

so können wir (11.4) für $t = t_j$ und $x = x_i$ schreiben als

$$\frac{1}{s}(u_i^{j+1} - u_i^j) - \frac{1-\theta}{h^2}(u_{i+1}^j - 2u_i^j + u_{i-1}^j) - \frac{\theta}{h^2}(u_{i+1}^{j+1} - 2u_i^{j+1} + u_{i-1}^{j+1}) = O(s+h^2). \tag{11.5}$$

Die Finite Differenzenmethode beruht nun darauf, für gegebenes $\theta \in [0,1]$ Werte $w_i^j \in \mathbb{R}$ zu berechnen, indem diese Gleichungen mit rechter Seite $= 0$ gelöst werden, wobei die entsprechenden Randbedingungen berücksichtigt werden.

Wir suchen also also Werte $w_i^j \in \mathbb{R}$, $i = 0, \ldots, N$, $j = 0, \ldots, M$, welche die Gleichungen

$$\frac{1}{s}(w_i^{j+1} - w_i^j) - \frac{1-\theta}{h^2}(w_{i+1}^j - 2w_i^j + w_{i-1}^j) - \frac{\theta}{h^2}(w_{i+1}^{j+1} - 2w_i^{j+1} + w_{i-1}^{j+1}) = 0 \qquad (11.6)$$

für $i = 1, \ldots, N-1$ und $j = 0, \ldots, M-1$ sowie die Randbedingungen

$$w_i^0 = u_0(x_i), \;\; i = 0, \ldots, N \quad \text{und } w_0^j = g_1(t_j), \;\; w_N^j = g_2(t_j), \;\; j = 0, \ldots, M \qquad (11.7)$$

erfüllen. Die Hoffnung ist dabei natürlich, dass die Werte w_i^j Approximationen $w_i^j \approx u_i^j$ der exakten Funktionswerte $u_i^j = u(t_j, x_i)$ in den Gitterpunkten darstellen. Wir werden später beweisen, dass dies unter geeigneten Voraussetzungen auch tatsächlich so ist.

11.2 Lösung der Finiten Differenzengleichungen

Wie löst man nun das Gleichungssystem (11.6), (11.7) numerisch? Um (11.6) dafür noch etwas zu vereinfachen, multiplizieren wir die Gleichung mit s und setzen $\alpha := s/h^2$. Mit Umstellen der Terme (alle Terme mit $j+1$ nach links, den Rest nach rechts) ist (11.6) dann äquivalent zu

$$-\alpha\theta w_{i+1}^{j+1} + (2\alpha\theta + 1)w_i^{j+1} - \alpha\theta w_{i-1}^{j+1} = \alpha(1-\theta)w_{i+1}^j + (1 - 2\alpha(1-\theta))w_i^j + \alpha(1-\theta)w_{i-1}^j.$$

Diese Gleichung schreiben wir nun rekursiv in Matrixform und setzen die Randwerte gleich ein. Dies ergibt die implizite rekursive Vorschrift

$$Aw^{j+1} = Bw^j + d^j, \;\; j = 0, \ldots, M-1 \qquad (11.8)$$

mit den Unbekannten

$$w^j = (w_1^j, \ldots, w_{N-1}^j)^T,$$

und der Matrix

$$A = \begin{pmatrix} 2\alpha\theta+1 & -\alpha\theta & 0 & \cdots & 0 \\ -\alpha\theta & 2\alpha\theta+1 & -\alpha\theta & & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & & & 0 \\ 0 & \cdots & 0 & -\alpha\theta & 2\alpha\theta+1 \end{pmatrix},$$

die wir kurz auch als $A = \operatorname{diag}(-\alpha\theta, 2\alpha\theta+1, \alpha\theta)$ schreiben, der Matrix

$$B = \operatorname{diag}(\alpha(1-\theta), 1 - 2\alpha(1-\theta), \alpha(1-\theta))$$

und dem Vektor

$$d^j = \begin{pmatrix} \alpha(1-\theta)g_1(t_j) + \alpha\theta g_1(t_{j+1}) \\ 0 \\ \vdots \\ 0 \\ \alpha(1-\theta)g_2(t_j) + \alpha\theta g_2(t_{j+1}) \end{pmatrix},$$

der sich aus den Randbedingungen für w_0^j und w_N^j in (11.7) ergibt. Da durch (11.7) zudem die Werte w^0 eindeutig bestimmt sind, können wir (11.8) damit rekursiv für $j = 0, \ldots, M-1$ lösen, indem wir in jedem Schritt das lineare Gleichungssystem mit der angegebenen Matrix A numerisch lösen, vorausgesetzt A ist invertierbar, was wir am Ende dieses Abschnitts untersuchen.

Formal könnte man die Iteration (11.8) dann auch explizit als

$$w^{j+1} = A^{-1}Bw^j + A^{-1}d^j$$

schreiben, aus der Einführung in die Numerik ist aber bekannt, dass die explizite Verwendung von A^{-1} numerisch weniger effizient als die Lösung des zugehörigen linearen Gleichungssystems (11.8) ist.

Für $\theta = 0$ nennt man das Verfahren *Vorwärts-Differenzenverfahren*, für $\theta = 1$ *Rückwärts-Differenzenverfahren*. Für $\theta \in (0, 1)$ erhält man *gemischte Vorwärts-Rückwärts-Differenzenverfahren*, von denen besonders das Verfahren für $\theta = 1/2$ wichtig ist, das sogenannte *Crank-Nicolson-Verfahren*. Warum das so ist, sehen wir später. Tatsächlich ist das Crank-Nicolson-Verfahren nichts anderes als die implizite Mittelpunktregel, vgl. Abschnitte 5.1 und 6.3.

Für das Vorwärts-Differenzenverfahren mit $\theta = 0$ wird die Lösung von (11.8) besonders einfach, da A dann die Einheitsmatrix ist. Anders gesagt ist $\theta = 0$ der einzige Wert, mit dem man ein explizites Verfahren erhält. Wir werden aber im nächsten Abschnitt sehen, dass dies aus anderen Gründen keine besonders gute Wahl ist.

Um die Lösbarkeit von (11.8) und allgemein die Existenz eindeutiger Lösungen w^j zu garantieren, reicht es aus, die Invertierbarkeit von A zu untersuchen. Dazu müssen wir nachweisen, dass A keinen Eigenwert $\lambda = 0$ besitzt. Dies kann man leicht mit dem folgenden Satz aus der linearen Algebra beweisen.

Satz 11.4 (Satz von Gerschgorin) Für alle Eigenwerte $\lambda \in \mathbb{C}$ einer Matrix $A \in \mathbb{R}^{n\times n}$ mit Einträgen a_{ij} gilt

$$\lambda \in \bigcup_{i=1}^{n} \left\{ z \in \mathbb{C} \;\middle|\; |z - a_{ii}| \le \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| \right\}.$$

Beweis: Sei $x = (x_1, \ldots, x_n)^T \neq 0$ ein Eigenvektor von A zu λ und sei x_i der betragsmäßig größte Eintrag von x. Wegen

$$\lambda x_i = (Ax)_i = \sum_{j=1}^{n} a_{ij} x_j$$

folgt mit Division durch $x_i \neq 0$

$$|\lambda - a_{ii}| = \left| \sum_{j=1}^{n} a_{ij} \frac{x_j}{x_i} - a_{ii} \right| = \left| \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij} \frac{x_j}{x_i} \right| \le \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| \left| \frac{x_j}{x_i} \right| \le \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}|.$$

Daraus folgt die Behauptung. □

Korollar 11.5 Für alle Eigenwerte λ der Matrix A aus (11.8) gilt $|\lambda| \geq 1$ für alle $\alpha > 0$ und alle $\theta \in [0,1]$. Insbesondere ist A invertierbar.

Beweis: Sei λ ein Eigenwert von A. Nach Satz 11.4 gilt dann für ein $i = 1, \ldots, N-1$

$$|\lambda - a_{ii}| \leq \sum_{\substack{j=1 \\ j \neq i}}^{N-1} |a_{ij}| \leq 2\alpha\theta.$$

Damit folgt mit der umgekehrten Dreiecksungleichung und wegen $a_{ii} = 2\alpha\theta + 1 > 0$

$$2\alpha\theta \geq |\lambda - a_{ii}| \geq 2\alpha\theta + 1 - |\lambda|,$$

woraus die Behauptung folgt. □

11.3 Konsistenz, Stabilität und Konvergenz

Die Konvergenzanalyse des Finiten Differenzenschemas ist konzeptionell ähnlich zu den Mehrschrittverfahren für gewöhnliche Differentialgleichungen. Wieder benötigen wir zwei Eigenschaften: Konsistenz und Stabilität. Die Konsistenzbedingung stellt dabei sicher, dass die Lösung in einem Schritt von (11.8) nur leicht von der exakten Lösung abweicht, die Stabilitätsbedingung garantiert, dass sich diese kleinen Abweichungen in der Iteration in (11.8) nicht zu sehr großen Fehlern aufschaukeln. Wir werden diese beiden Begriffe nun genau definieren und beweisen, dass daraus tatsächlich die Konvergenz — in geeignetem Sinne — folgt.

Die Konsistenz definieren wir auf die gleiche Weise wie bei den Mehrschrittverfahren in Abschnitt 9.1: Wir betrachten den Fehler, den wir beim Einsetzen der exakten Lösung der Gleichung (11.1) in das numerische Schema (11.8) machen.

Definition 11.6 Das Finite Differenzenschema (11.8) heißt konsistent mit Ordnung $a > 0$ in der Zeit und Ordnung $b > 0$ im Raum, falls für die mittels $u^j = (u(t_j, x_1), \ldots, u(t_j, x_{N-1}))^T$ definierten Vektoren mit den exakten Lösungswerten die Ungleichung

$$\|Au^{j+1} - Bu^j - d^j\|_\infty \leq Cs(s^a + h^b) \tag{11.9}$$

für eine Konstante $C > 0$ und alle $j = 0, \ldots, M-1$ erfüllen. □

Beachte, dass die Bedingung einen zusätzlichen Faktor s aber keinen Faktor h in der Abschätzung verlangt. Dies liegt daran, dass über die t-Variable $M = T/s$ Schritte iteriert wird, wodurch — ganz analog zu den numerischen Schemata für gewöhnliche Differentialgleichungen — in der Konvergenzanalyse eine Potenz "verlorengeht". Da über x nicht iteriert wird, ist eine Multiplikation mit der Ortsschrittweite h nicht nötig.

Der folgende Satz zeigt, dass das Schema konsistent ist.

Satz 11.7 Die Lösung der Wärmeleitungsgleichung erfülle $u \in C^4$. Dann ist das Schema (11.8) für alle $\theta \in [0,1]$ konsistent mit $a = 1$ und $b = 2$. Im Falle $\theta = 1/2$ erhöht sich die erste Konsistenzordnung auf $a = 2$.

Beweis: Die allgemeine Aussage folgt sofort aus (11.5) wenn wir beachten, dass die Komponenten der rechten Seite von (11.9) gerade durch Multiplikation von (11.5) mit s entstehen, weswegen wir nun $O(s(s+h^2))$ erhalten. Mit $K = \Omega$ und der vor Lemma 11.2 definierten Gleichmäßigkeit der Konstanten in den O-Termen folgt die Existenz von $C = C_K$.

Die spezielle Aussage für $\theta = 1/2$ folgt, wenn wir zusätzlich zu den Abschätzungen aus Lemma 11.2 noch die zusätzlichen Taylor-Entwicklungen

$$\frac{1}{s}(u_i^{j+1} - u_i^j) = u_t(t_j, x_i) + \frac{s}{2}u_{tt}(t_j, x_i) + O(s^2),$$

$$\frac{1}{s}(u_{xx}(t_{j+1}, x_i) - u_{xx}(t_j, x_i)) = u_{xxt}(t_j, x_i) + O(s)$$

sowie (hier verwenden wir die gerade aufgeschriebene Entwicklung im letzten Schritt)

$$\begin{aligned}
&\frac{1-\theta}{h^2}(u_{i+1}^j - 2u_i^j + u_{i-1}^j) + \frac{\theta}{h^2}(u_{i+1}^{j+1} - 2u_i^{j+1} + u_{i-1}^{j+1}) \\
= \;& (1-\theta)u_{xx}(t_j, x_i) + \theta u_{xx}(t_{j+1}, x_i) + O(h^2) \\
= \;& u_{xx}(t_j, x_i) + \theta(u_{xx}(t_{j+1}, x_i) - u_{xx}(t_j, x_i)) + O(h^2) \\
= \;& u_{xx}(t_j, x_i) + \theta s u_{xxt}(t_j, x_i) + O(s^2 + h^2)
\end{aligned}$$

betrachten. Damit lässt sich (11.5) verbessern zu

$$\begin{aligned}
&\frac{1}{s}(u_i^{j+1} - u_i^j) - \frac{1-\theta}{h^2}(u_{i+1}^j - 2u_i^j + u_{i-1}^j) - \frac{\theta}{h^2}(u_{i+1}^{j+1} - 2u_i^{j+1} + u_{i-1}^{j+1}) \\
&= \frac{s}{2}(u_{tt}(t_j, x_i) - 2\theta u_{xxt}(t_j, x_t)) + O(s^2 + h^2).
\end{aligned}$$

Wegen $u_{tt} - u_{xxt} = (u_t - u_{xx})_t = 0$ (denn u löst ja gerade (11.1), also $u_t - u_{xx} = 0$) fällt der erste Summand für $\theta = 1/2$ weg. Wie oben erhalten wir durch Multiplikation dieser Gleichung mit s gerade die Komponenten von (11.9) und damit die Behauptung. □

Im Sinne der Konsistenz ist folglich $\theta = 1/2$ — also das Crank-Nicolson-Verfahren — am Besten. Dies ist im Einklang mit der Numerik gewöhnlicher Differentialgleichungen. Schreibt man das Butcher-Tableau des hier verwendeten Schemas in der Zeit auf, so erhält man

θ	θ
	1

Überprüft man die Bedingungsgleichungen aus Satz 4.5, so sieht man, dass das Verfahren wegen $b_1 = 1$ für alle $\theta \in [0, 1]$ konsistent mit Ordnung 1 ist, wegen $b_1 c_1 = \theta$ aber nur für $\theta = 1/2$ die Ordnung 2 besitzt (Satz 4.5 ist hier anwendbar, weil das Verfahren wegen $b_1 = \theta = a_{11}$ invariant unter Autonomisierung ist).

Die zweite Bedingung, die wir zur Herleitung der Konvergenz benötigen, ist die sogenannte Stabilität. Wir erläutern zunächst anschaulich, warum wir hier wie bei den Mehrschrittverfahren eine Stabilitätsbedingung benötigen, obwohl wir hier ja in der Zeit nur Einschrittverfahren betrachten:

Bezeichnen wir mit $e^j = Au^{j+1} - Bu^j - d^j$ den lokalen Konsistenzfehler und mit $\hat{e}^j = u^j - w^j$ den globalen Fehler der numerischen Lösung, so gilt für den Fehler die Rekursion

$$A\hat{e}^{j+1} = Au^{j+1} - Aw^{j+1} = Bu^j + d^j + e^j - Bw^j - d^j = B\hat{e}^j + e^j,$$

was wir auch als

$$\hat{e}^{j+1} = A^{-1}B\hat{e}^j + A^{-1}e^j$$

schreiben können. Schreiben wir kurz $\widetilde{A} = A^{-1}B$, $\tilde{e}^j = A^{-1}e^j$ und verwenden $\hat{e}_0 = 0$ (da die Randbedingungen in $t_0 = 0$ für u und w ja identisch sind), so können wir per Induktion leicht die explizite Formel

$$\hat{e}^j = \sum_{k=0}^{j-1} \widetilde{A}^{j-k-1}\tilde{e}^k \tag{11.10}$$

herleiten. Für immer kleiner werdende s und h passieren nun zwei Dinge: Die Anzahl der Schritte j wird immer größer und die Matrix A^{-1}, die den Vektor $\tilde{e}^k$ definiert, verändert sich. Das zweite Problem untersuchen wir später im Beweis von Satz 11.10, das erste betrachten wir jetzt.

Da wir nicht davon ausgehen können, dass sich die einzelnen Summanden günstig gegeneinander aufheben, muss nun, damit $\hat{e}^j$ klein wird, jeder der Summanden klein sein. Für jedes beliebige $k \in \mathbb{N}$ mit $k < j$ muss also der Term $\widetilde{A}^{j-k-1}\tilde{e}^k$ klein sein. Da wir für immer kleinere Zeitschritte s auch bei festem Zeitintervall $[0, T]$ immer mehr Schritte und damit immer größere j erhalten, können wir keine von s unabhängige obere Schranke für $j-k-1$ angeben; ebenso haben wir keine Informationen über $\tilde{e}^k$. Wir können zwar annehmen, dass dieser Fehler klein ist, er wird für $s \to 0$ aber nicht beliebig klein, da ja auch die räumliche Diskretisierung zu diesem Fehler beiträgt.

Wir brauchen also eine Eigenschaft der Matrix $\widetilde{A}$, die sicherstellt, dass $\widetilde{A}^l x$ für beliebige Vektoren x mit kleiner Norm und beliebigem $l \in \mathbb{N}$ ebenfalls klein ist.

Dies liefert uns Satz 9.10:

Satz 11.8 Für eine Matrix $A \in \mathbb{R}^{n\times n}$ existiert genau dann eine Konstante $C > 0$, so dass

$$\|A^l x\| \leq C\|x\|$$

gilt für alle $l \in \mathbb{N}$ und alle $x \in \mathbb{R}^n$, wenn alle Eigenwerte λ von A die Ungleichung $|\lambda| \leq 1$ erfüllen und alle Eigenwerte mit $|\lambda| = 1$ halbeinfach sind (d.h. dass λ eine einfache Nullstelle des Minimalpolynoms ist).

Der folgende Satz wendet dieses Kriterium auf die Matrix $A^{-1}B$ an.

Satz 11.9 Die Matrix $A^{-1}B$ erfüllt die Stabilitätsbedingung aus Satz 11.8, wenn die Ungleichungen

$$\begin{aligned} 0 < \alpha \leq \frac{1}{2-4\theta}, &\qquad \text{falls } 0 \leq \theta < \frac{1}{2} \\ 0 < \alpha, &\qquad \text{falls } \frac{1}{2} \leq \theta \leq 1 \end{aligned}$$

gelten. Für die euklidische Norm $\|\cdot\|_2$ kann die Konstante C in Satz 11.8 sogar als $C = 1$ gewählt werden.

Beweis: Wegen

$$B = \frac{1}{\theta}\mathrm{Id} - \frac{1-\theta}{\theta}A$$

gilt

$$A^{-1}B = \frac{1}{\theta}A^{-1} - \frac{1-\theta}{\theta}\mathrm{Id}.$$

Jeder Eigenwert von $A^{-1}B$ ist also von der Form $\lambda = \frac{1}{\theta\mu} - \frac{1-\theta}{\theta}$, wobei μ ein Eigenwert von A ist. Wir können nun A schreiben als $A = \mathrm{Id} + \alpha\theta G$ mit $G = \mathrm{diag}(-1, 2, -1)$. Für G kann man nachrechnen, dass die Eigenwerte gerade durch

$$4\sin^2\left(\frac{k\pi}{2N}\right), \quad k = 1, \ldots, N-1$$

gegeben sind. Die Eigenwerte von A lauten daher

$$\mu_k = 1 + 4\alpha\theta\sin^2\left(\frac{k\pi}{2N}\right), \quad k = 1, \ldots, N-1$$

und die von $A^{-1}B$ folglich

$$\lambda_k = \frac{1}{\theta}\frac{1}{1 + 4\alpha\theta\sin^2(k\pi/2N)} - \frac{1-\theta}{\theta} = 1 - \frac{4\alpha\sin^2(k\pi/2N)}{1 + 4\alpha\theta\sin^2(k\pi/2N)}.$$

Weil $A^{-1}B$ symmetrisch und damit diagonalisierbar ist, sind alle Eigenwerte halbeinfach. Es reicht also, $|\lambda_k| \leq 1$ nachzuweisen.

Wegen $\alpha > 0$ ist $|\lambda_k| \leq 1$ äquivalent zu

$$\frac{4\alpha\sin^2(k\pi/2N)}{1 + 4\alpha\theta\sin^2(k\pi/2N)} \leq 2 \quad \Leftrightarrow \quad (2 - 4\theta)\alpha\sin^2(k\pi/2N) \leq 1.$$

Für $\theta \geq 1/2$ ist diese Ungleichung immer erfüllt, für $\theta \in [0, 1/2)$ wegen $\sin^2(k\pi/2N) \leq 1$ für alle $\alpha \leq 1/(2 - 4\theta)$.

Dass die Konstante als $C = 1$ gewählt werden kann, folgt wiederum aus der Symmetrie von $A^{-1}B$. Für symmetrische Matrizen A ist die induzierte Matrixnorm $\|A\|_2$ gerade gleich dem maximalen Betrag der Eigenwerte (dies folgt aus der Darstellung der 2-Norm für Matrizen über den Spektralradius, vgl. Satz 2.5 aus der Einführung in die Numerik). Damit folgt für alle $x \in \mathbb{R}^n$ und alle $j \in \mathbb{N}$ die Ungleichung $\|A^j x\|_2 \leq \|A\|_2^j \|x\|_2 \leq \|x\|_2$, also $C = 1$. □

Im Sinne der Stabilität sind also die Verfahren mit $\theta \geq 1/2$ besonders gut, weil sie für alle Kombinationen aus Zeit- und Raumschrittweite s und h stabil sind. Man sagt, diese Verfahren sind *unbedingt stabil*.

Nun können wir schließlich die Konvergenz beweisen.

Satz 11.10 Unter den Voraussetzungen der Sätze 11.7 und 11.9 existiert eine Konstante $C_0 > 0$, so dass für alle $\theta \in [0, 1]$ und alle hinreichend kleinen $s, h > 0$ gilt

$$\max_{j=1,\ldots,M} \|w^j - u^j\|_{l_2} \leq C_0(s + h^2).$$

Im Falle $\theta = 1/2$ gilt diese Abschätzung sogar mit $C_0(s^2 + h^2)$.

Hierbei ist die *diskrete l_2-Norm* $\|\cdot\|_{l_2}$ definiert durch

$$\|x\|_{l_2} = \|x\|_2/(N-1).$$

Beweis: Betrachte die Fehlergleichung (11.10), die sich durch einfache Umnummerierung der Indizes schreiben lässt als

$$\hat{e}^j = \sum_{k=0}^{j-1} \widetilde{A}^k \tilde{e}^{j-k-1}$$

mit $\widetilde{A} = A^{-1}B$, $\tilde{e}^j = A^{-1}e^j$ und $\hat{e}^j = u^j - w^j$. Wir müssen zeigen, dass $\|\hat{e}^j\|_{l_2} \le C_0(s+h^2)$ bzw. $\le C_0(s^2+h^2)$ gilt für alle $j = 1, \ldots, M$.

Für die lokalen Fehler e^j gilt nach Satz 11.7 und der Definition der diskreten l_2-Norm die Ungleichung $\|e^j\|_{l_2} \le \|e^j\|_\infty \le Cs(s^p + h^2)$ mit $p = 1$ für allgemeines θ und $p = 2$ für $\theta = 1/2$. Nach Korollar 11.5 gilt $|\lambda| \ge 1$ für alle Eigenwerte λ von A. Also gilt $|\lambda| \le 1$ für alle Eigenwerte von A^{-1} und weil A und damit auch A^{-1} symmetrisch sind folgt $\|A^{-1}\|_2 \le 1$ (vgl. auch das Ende des Beweises von Satz 11.9). Damit folgt $\|\tilde{e}^j\|_{l_2} \le Cs(s^p + h^2)$.

Nach Satz 11.9 gilt für die Summanden von $\hat{e}^j$ nun $\|\widetilde{A}^k \tilde{e}^{j-k-1}\|_{l_2} \le \|\tilde{e}^{j-k-1}\|_{l_2}$. Damit folgt

$$\|\hat{e}^j\|_{l_2} \le \sum_{k=0}^{j-1} \|\widetilde{A}^k \tilde{e}^{j-k-1}\|_{l_2} \le jCs(s^p + h^2).$$

Wegen $j \le M \le T/s$ folgt

$$\|\hat{e}^j\|_{l_2} \le \frac{T}{s} Cs(s^p + h^2) = TC(s^p + h^2).$$

Damit ergibt sich die Behauptung mit $C_0 = TC$. □

Dieser Satz zeigt sofort, warum das Crank-Nicolson-Verfahren, also $\theta = 1/2$, das bevorzugte Verfahren ist: Es hat die höchste Konvergenzordnung und ist unbedingt — d.h. für alle Kombinationen von s und h bzw. von N und M — stabil.

Allerdings ist das Crank-Nicolson-Verfahren nicht L-stabil, vgl. Abschnitt 6.3. Für Gleichungen, bei denen diese Eigenschaft wichtig ist (also bei betragsmäßig großen Eigenwerten mit negativen Realteilen, wie sie z.B. bei hochfrequenten Schwingungen in Form von großen Imaginärteilen auftreten) ist das implizite Eulerverfahren zu bevorzugen, obwohl es niedrigere Konsistenzordnung besitzt.

Kapitel 12

Finite Elemente für elliptische Gleichungen

Finite Elemente funktionieren auf grundlegend andere Weise als Finite Differenzen. Sie beruhen auf der schwachen Form der PDG — einer Integraldarstellung der Lösung — und funktionalanalytischen Methoden. Wir betrachten daher zunächst diese mathematischen Hintergründe, bevor wir die Lösungsmethode selbst behandeln. Dabei ist im Folgenden Ω stets eine offene und beschränkte Teilmenge des $\mathbb{R}^n$.

12.1 Schwache Form der PDG

Wir betrachten in diesem Kapitel elliptische partielle Differentialgleichungen der Form

$$-\mathrm{div}(\mathcal{A}(x)\nabla u) + c(x)u - f(x) = 0 \tag{12.1}$$

mit Randbedingungen wie z.B.

$$u = g \qquad \text{auf } \partial\Omega$$

oder

$$\mathcal{A}(x)\nabla u \cdot \nu = 0 \text{ auf } \partial\Omega.$$

Dabei sind $\mathcal{A} : \overline{\Omega} \to \mathbb{R}^{n\times n}$, $c, f : \Omega \to \mathbb{R}$ stetig differenzierbare Funktionen, $\mathcal{A}(x)$ ist symmetrisch für alle $x \in \overline{\Omega}$ und $\nu = \nu(s)$ bezeichnet die äußere Normale von Ω im Punkt $s \in \partial\Omega$. Der Punkt in $\nabla u \cdot \nu$ steht für das euklidische Skalarprodukt.

In diesem Abschnitt leiten wir die schwache Form dieser Gleichung formal her. Dass das Konzept einen sinnvollen Lösungsbegriff liefert, beweisen wir dann im nachfolgenden Abschnitt.

Zur Herleitung der schwachen Form betrachten wir hinreichend oft differenzierbare Funktionen $v : \Omega \to \mathbb{R}$ — sogenannte Testfunktionen. Dann gilt für die Lösungen der obigen PDG

$$\int_\Omega \left(-\mathrm{div}(\mathcal{A}(x)\nabla u(x)) + c(x)u(x) - f(x)\right) v(x)dx = 0.$$

Der Einfachheit halber beschränken wir uns bei der folgenden Herleitung auf die Randbedingung $\mathcal{A}(x)\nabla u\nu = 0$. Gilt diese Randbedingung, so folgt

$$\int_\Omega \left(-\mathrm{div}(\mathcal{A}(x)\nabla u(x)) + c(x)u - f(x)\right) v(x)dx + \int_{\partial\Omega} (\mathcal{A}(s)\nabla u(s)\cdot\nu(s))v(s)ds = 0. \quad (12.2)$$

Mit der Abkürzung $w(x) = \mathcal{A}(x)\nabla u(x)$, dem Satz von Gauß (angewendet auf die Funktion wv) und der Produktregel für die Divergenz ergibt sich

$$\int_{\partial\Omega} (w(s)\cdot\nu(s))v(s)ds = \int_\Omega \mathrm{div}(wv)(x)dx = \int_\Omega w(x)\nabla v(x) + (\mathrm{div}w(x))v(x)dx.$$

Eingesetzt in (12.2) und wieder mit $\mathcal{A}(x)\nabla u(x)$ statt $w(x)$ geschrieben erhalten wir so

$$\int_\Omega (\mathcal{A}(x)\nabla u(x))\cdot\nabla v(x) + c(x)u(x)v(x) - f(x)v(x)dx = 0 \quad (12.3)$$

für alle Testfunktionen v. Dabei steht $\nabla v\cdot\nabla u$ wieder für das euklidische Skalarprodukt. Dies ist die schwache Form der PDG (12.1) und ihre Lösung heißt demgemäß schwache Lösung. Mit den Abkürzungen

$$a(u,v) := \int_\Omega \mathcal{A}(x)\nabla u(x)\cdot\nabla v(x) + c(x)u(x)v(x)dx \text{ und } l(v) := \int_\Omega f(x)v(x)dx \quad (12.4)$$

können wir dies kurz schreiben als

$$a(u,v) - l(v) = 0 \quad (12.5)$$

für alle Testfunktionen v.

Diese Gleichung können wir auch als Lösung eines Minimierungsproblems interpretieren, was im Folgenden nützlich sein wird. Betrachten wir das Funktional

$$\mathcal{E}(u) := \frac{1}{2}a(u,u) - l(u)$$

so gilt, weil a eine symmetrische stetige Bilinearform und l eine lineare Abbildung ist, für die Ableitung in Richtung v die Formel

$$D\mathcal{E}(u)v := a(u,v) - l(v)$$

(das werden wir im Beweis von Satz 12.1 noch formal beweisen). Ist u nun ein Minimierer von $\mathcal{E}$, so gilt $D\mathcal{E}(u)v = 0$ für alle Testfunktionen v. Dies ist gerade (12.5); diese Gleichung stellt also die notwendigen Optimalitätsbedingungen für $\mathcal{E}$ dar.

Warum schreibt man nun die Gleichung (12.1) in die schwache Form (12.3) bzw. (12.5) um? Der Grund liegt darin, dass es oft keine Lösung $u : \Omega \to \mathbb{R}$ gibt, die (12.1) punktweise erfüllt, weil die in Frage kommenden Kandidaten u gar nicht für alle $x \in \Omega$ zweimal differenzierbar sind. In der schwachen Form braucht man zunächst einmal nur die erste Ableitung von u, was die Suche bereits vereinfacht. Zum anderen braucht man noch nicht einmal diese, da die Ableitung nur unter dem Integral vorkommt und deswegen durch geschickte Umformulierung komplett vermieden werden kann. Dies führt zum Konzept der schwachen Ableitung, vgl. Definition 12.2, für das wir im folgenden Abschnitt zeigen, dass es unter geeigneten Annahmen einen Existenz- und Eindeutigkeitssatz für die sogenannte schwache Lösung liefert.

12.2 Lösungstheorie

Wir gehen hier wie folgt vor: wir zeigen zunächst, dass das abstrakte Minimierungsproblem

$$\min_{u \in H} \mathcal{E}(u) \tag{12.6}$$

mit

$$\mathcal{E}(u) := \frac{1}{2} a(u,u) + l(u) \tag{12.7}$$

unter geeigneten Annahmen eine eindeutige Lösung besitzt, die (12.5) erfüllt. In einem zweiten Schritt zeigen wir dann, dass das aus (12.1) gemäß der gerade durchgeführten Herleitung hervorgehende Funktional $\mathcal{E}$ die dafür nötigen Annahmen erfüllt.

Existenz eines Minimierers

Für den ersten Teil sei H ein Hilbertraum, also ein vollständiger normierter Vektorraum mit Skalarprodukt $\langle \cdot, \cdot \rangle$, dessen Norm durch $\|u\|_H = \sqrt{\langle u, u \rangle}$ gegeben ist. Die Abbbildung $a : H \times H \to \mathbb{R}$ in (12.7) sei eine symmetrische Bilinearform und $l : H \to \mathbb{R}$ eine stetige lineare Abbildung. Die Menge dieser Abbildungen bezeichnen wir mit H^*. Die Menge H^* ist dann selbst wieder ein Vektorraum, der sogenannte Dualraum von H. Als Norm auf H^* verwenden wir die Operatornorm

$$\|l\|_{H^*} = \sup_{\substack{u \in H \\ \|u\|_H = 1}} |l(u)|.$$

Wir sagen,

- a ist *stetig*[1], falls $C > 0$ existiert mit $|a(u,v)| \leq C \|u\|_H \|v\|_H$ für alle $u, v \in H$
- a ist *elliptisch*[2], falls $\alpha > 0$ existiert mit $a(u,u) \geq \alpha \|u\|_H^2$ für alle $u \in H$.

Dann gilt der folgende Satz.

Satz 12.1 (Lax-Milgram) Sei a stetig und elliptisch mit Konstante $\alpha > 0$. Dann existiert ein eindeutiger Minimierer $u^* \in H$ von (12.6). Dieser erfüllt die Gleichung (12.5) für alle $v \in H$ und es gilt

$$\|u^*\|_H \leq \frac{1}{\alpha} \|l\|_{H^*}.$$

[1] Diese Definition ist tatsächlich äquivalent zu der üblichen Stetigkeitsdefinition. Dies zu sehen erfordert allerdings ein paar Zeilen Beweis, die wir hier aus Zeitgründen auslassen.

[2] Da diese Definition vom zu Grunde liegenden Raum H abhängt, wird sie auch "H-Elliptizität" genannt. Da der zu Grunde liegende Raum in der Literatur oft mit V statt wie bei uns mit H bezeichnet wird, ist auch die Bezeichnung "V-Elliptizität" gebräuchlich.

Beweis: Wir betrachten eine beliebige Folge u_k in H mit $\mathcal{E}(u_k) \to \inf_{u\in H} \mathcal{E}(u)$ und zeigen, dass diese eine Cauchy-Folge ist. Da für alle $u \in H$ gilt

$$\mathcal{E}(u) \geq \frac{1}{2}\alpha\|u\|_H^2 - \|l\|_{H^*}\|u\|_H = \frac{1}{2\alpha}(\alpha\|u\|_H - \|l\|_{H^*})^2 - \frac{\|l\|_{H^*}^2}{2\alpha} \geq -\frac{\|l\|_{H^*}^2}{2\alpha},$$

ist $\mathcal{E}(u_k)$ nach unten beschränkt. Es sei nun $\varepsilon > 0$ gegeben und $M \in \mathbb{N}$ so gewählt, dass $\mathcal{E}(u_k) - \inf_{u\in H} \mathcal{E}(u) < \alpha\varepsilon^2/8$ für alle $k \geq M$. Dann folgt für alle $m, n \geq M$

$$\begin{aligned}
\alpha\|u_n - u_m\|_H^2 &\leq a(u_n - u_m, u_n - u_m) \\
&= 2a(u_n, u_n) + 2a(u_m, u_m) - a(u_n + u_m, u_n + u_m) \\
&= 4\mathcal{E}(u_n) + 4\mathcal{E}(u_m) - 8\mathcal{E}((u_n + u_m)/2) \\
&\leq 4\mathcal{E}(u_n) + 4\mathcal{E}(u_m) - 8\inf_{u\in H} \mathcal{E}(u) \quad < \quad \alpha\varepsilon^2.
\end{aligned}$$

Also ist $\|u_n - u_m\|_H \leq \varepsilon$, damit ist u_k eine Cauchy-Folge und besitzt einen Grenzwert $u^* \in H$. Wegen der Stetigkeit von a ist dies ein Minimierer. Für jeden weiteren Minimierer $u^{**} \in H$ gilt mit der gleichen Rechnung wie eben

$$\alpha\|u^{**} - u^*\|_H \leq 4\mathcal{E}(u^*) + 4\mathcal{E}(u^{**}) - 8\inf_{u\in H} \mathcal{E}(u) = 0$$

und daher $u^{**} = u^*$. Aus der Definition von $\mathcal{E}$ folgt

$$\mathcal{E}(u + v) = \frac{1}{2}a(u + v, u + v) - l(u + v) = \left(\frac{1}{2}a(u, u) - l(u)\right) + (a(u, v) - l(v)) + \frac{1}{2}a(v, v).$$

Mit $D\mathcal{E}(u)v = a(u, v) - l(v)$ folgt wegen der Stetigkeit von a

$$\mathcal{E}(u + v) = \mathcal{E}(u) + D\mathcal{E}(u)v + O(\|v\|^2),$$

also ist $D\mathcal{E}(u)v = a(u, v) - l(v)$ die Richtungsableitung von $\mathcal{E}$ in u in Richtung v. Ist nun $D\mathcal{E}(u^*)v \neq 0$ für ein $v \in H$, so ist $D\mathcal{E}(u^*)\eta v + O(\|\eta v\|^2) < 0$ für ein betragsmäßig hinreichend kleines $\eta > 0$ oder $\eta < 0$. Damit ist aber $\mathcal{E}(u^* + \eta v) < \mathcal{E}(u^*)$, was der Optimalität von u^* widerspricht. Also gilt (12.5). Die letzte Ungleichung folgt aus

$$\alpha\|u^*\|_H^2 \leq a(u^*, u^*) = l(u^*) \leq \|l\|_{H^*}\|u^*\|_H.$$

□

Sobolevräume

Die Eigenschaften, die wir für a aus (12.4) also nachweisen müssen, sind Stetigkeit und Elliptizität. Dazu müssen wir geeignete Funktionenräume definieren. Wir erinnern dazu zunächst an die Definition des Raums $L_2(\Omega)$ für $\Omega \subset \mathbb{R}^n$. Dies sind alle Funktionen $v : \Omega \to \mathbb{R}$, für die das Integral

$$\int_\Omega v(x)^2 dx$$

existiert. Mit

$$\langle v, w\rangle_{L_2} := \int_\Omega v(x)w(x)dx \text{ und } \|v\|_{L_2} = \sqrt{\langle v, v\rangle_{L_2}}$$

wird ein Skalarprodukt und eine Norm auf $L_2(\Omega)$ definiert, wobei man Funktionen, die sich nur auf einer Nullmenge unterscheden, miteinander identifizieren muss. Der Raum $L_2(\Omega)$ wird damit ein Hilbertraum von Äquivalenzklassen von Funktionen.

Für L_2-Funktionen kann man nun wie folgt einen schwachen Ableitungsbegriff definieren. Dabei ist

$$D(\Omega) := \{\varphi \in C^\infty(\Omega) \,|\, \text{es gibt eine kompakte Menge } K \subset \Omega \text{ mit } \varphi(x) = 0 \text{ für alle } x \notin K\}$$

die Menge der unendlich oft differenzierbaren Funktionen mit kompaktem Träger.

Definition 12.2 Eine Funktion $u \in L^2(\Omega)$ besitzt die schwache i-te partielle Ableitung $v = \partial_i u \in L_2(\Omega)$, wenn

$$\int_\Omega u(x) \frac{\partial}{\partial x_i} \varphi(x) dx = -\int_\Omega v(x)\varphi(x) dx$$

oder kurz

$$\left\langle u, \frac{\partial}{\partial x_i}\varphi \right\rangle_{L_2} = -\langle v, \varphi\rangle_{L_2}$$

für alle $\varphi \in D(\Omega)$ gilt. □

Die Motivation für diese Definition ist die Folgende: Falls u im klassischen Sinne partiell differenzierbar auf Ω mit Ableitung $\partial u/\partial x_i$ ist, so gilt nach Produktregel für alle $\varphi \in D(\Omega)$ die Gleichung

$$\int_\Omega u(x) \frac{\partial}{\partial x_i}\varphi(x) + \left(\frac{\partial}{\partial x_i} u(x)\right) \varphi(x) dx = \int_\Omega \frac{\partial}{\partial x_i}(u\varphi)(x) dx.$$

Für das rechte Integrals gilt nach dem Satz von Gauß nun

$$\int_\Omega \frac{\partial}{\partial x_i}(u\varphi)(x) dx = \int_{\partial\Omega} (u\varphi)(s)\nu_i(s) ds$$

und dieser Ausdruck ist gleich Null, weil φ auf dem Rand von Ω gleich Null ist. Folglich ist jede klassische partielle Ableitung auch eine schwache partielle Ableitung, aber die Menge der Funktionen, die schwache Ableitungen besitzt, ist viel größer.

Das Konzept lässt sich rekursiv oder direkt durch die Gleichung

$$(-1)^{|\alpha|} \left\langle u, \frac{\partial}{\partial \alpha}\varphi \right\rangle_{L_2} = \langle v, \varphi\rangle_{L_2}$$

auf höhere schwache Ableitungen $\partial_\alpha u$ verallgemeinern, wobei $\alpha = (\alpha_1, \ldots, \alpha_k)$ ein Multiindex mit $\alpha_i \in \{0, \ldots, n\}$ und $|\alpha| = k$ ist und $\frac{\partial}{\partial \alpha}\varphi = \frac{\partial}{\partial x_{\alpha_k}} \ldots \frac{\partial}{\partial x_{\alpha_1}}\varphi$ die übliche höhere partielle Ableitung ist.

Mit diesem Konzept können wir für jedes $m \in \mathbb{N}$ ein Skalarprodukt

$$\langle u, v\rangle_{H^m} := \sum_{|\alpha| \le m} \langle \partial_\alpha u, \partial_\alpha v\rangle_{L_2}$$

und die zugehörige Norm $\|u\|_{H^m} := \sqrt{\langle u, u\rangle_{H^m}}$ definieren.

Definition 12.3 Für $m \in \mathbb{N}$ definieren wir den Sobolevraum $H^m(\Omega)$ als die Menge aller Funktionen $u \in L_2(\Omega)$, für die die schwachen Ableitungen $\partial_\alpha u$ für alle Multiindizes α mit $|\alpha| \leq m$ existieren und $\|u\|_{H^m} < \infty$ gilt. □

Man kann (unter milden Regularitätsannahmen an Ω) beweisen, dass $H^m(\Omega)$ ein Hilbertraum ist und dass der Raum $C^\infty(\Omega) \cap H^m(\Omega)$ dicht in $H^m(\Omega)$ liegt. $H^m(\Omega)$ ist also eine Vervollständigung von $C^\infty(\Omega)$ bezüglich des Skalarprodukts $\langle u, v\rangle_{H^m}$, also $H^m(\Omega) = \overline{C}^\infty(\Omega)$. Analog gilt für den Sobolevraum $H_0^m(\Omega)$ der Funktionen $v \in H^m(\Omega)$ mit $v|_{\partial\Omega} \equiv 0$ die Beziehung $H_0^m(\Omega) = \overline{D}(\Omega)$.

Die folgenden Eigenschaften von Sobolevräumen werden wir im Folgenden benutzen. Diese gelten für Gebiete Ω mit Lipschitz-Rand.

- Für $m > n/2$ besteht die stetige Einbettung $H^m(\Omega) \hookrightarrow C(\overline{\Omega})$.
- Es besteht die kompakte Einbettung $H^{m+1}(\Omega) \hookrightarrow H^m(\Omega)$.
- Es gibt eine stetige lineare Abbildung $\gamma : H^1(\Omega) \to L_2(\partial\Omega)$, die sogenannte Spurabbildung, für die $\gamma u = u|_{\partial\Omega}$ für alle $u \in C^1(\Omega)$ und $\ker \gamma = H_0^1(\Omega)$ gilt.

Wir definieren im Folgenden den Gradienten ∇u für schwach differenzierbare Funktionen als $\nabla u = (\partial_1 u, \ldots, \partial_n u)^T$. Aus

$$\|u\|_{H^1}^2 = \|u\|_{L^2}^2 + \|\partial_1 u\|_{L_2}^2 + \ldots + \|\partial_n u\|_{L_2}^2$$

und

$$\|\nabla u\|_{L^2}^2 = \|\partial_1 u\|_{L_2}^2 + \ldots + \|\partial_n u\|_{L_2}^2$$

folgen die (Un)gleichungen

$$\|u\|_{H^1}^2 = \|u\|_{L^2}^2 + \|\nabla u\|_{L^2}^2, \quad \|u\|_{L^2} \leq \|u\|_{H^1} \quad \text{und} \quad \|\nabla u\|_{L^2} \leq \|u\|_{H^1}. \tag{12.8}$$

Elliptische Gleichungen auf Sobolevräumen

Wir haben nun alle Zutaten beisammen, die wir benötigen, um die Hauptresultate dieses Abschnitts zu beweisen. Wir betrachten dazu wieder die Bilinearform a aus (12.4), also

$$a(u, v) := \int_\Omega \mathcal{A}(x)\nabla u(x) \cdot \nabla v(x) + c(x)u(x)v(x)dx.$$

Satz 12.4 Es seien $A \in L_\infty(\Omega, \mathbb{R}^{n\times n})$ und $c \in L_\infty(\Omega)$. Zudem existieren $\tilde{\alpha} > 0$ und $\beta > 0$, so dass[3] $\mathcal{A}(x) \geq \tilde{\alpha}\mathrm{Id}$ und $c(x) \geq \beta$ gelten für alle $x \in \Omega$.

Dann ist a aus (12.4) elliptisch und stetig auf dem Raum $H^1(\Omega)$, d.h. die Voraussetzungen des Satzes von Lax-Milgram (Satz 12.1) sind erfüllt.

[3] Für zwei quadratische Matrizen gleicher Dimension A und B schreiben wir $\mathcal{A} \geq B$ falls $A - B$ positiv semidefinit ist.

Beweis: Die Stetigkeit folgt mit (12.8) aus der Ungleichung

$$\begin{aligned}\left|\int_\Omega \mathcal{A}(x)\nabla u(x)\cdot\nabla v(x)+c(x)u(x)v(x)dx\right| &\leq \|A\|_\infty\|\nabla u\|_{L_2}\|\nabla v\|_{L_2}+\|c\|_\infty\|u\|_{L_2}\|v\|_{L_2}\\ &\leq C\|u\|_{H^1}\|v\|_{H^1}.\end{aligned}$$

Die Elliptizität folgt ebenfalls mit (12.8) aus

$$a(u,u)\geq\tilde{\alpha}\|\nabla u\|_{L_2}^2+\beta\|u\|_{L_2}^2\geq\min\{\tilde{\alpha},\beta\}\|u\|_{H^1}^2.$$

□

In vielen praktischen Anwendungen ist $c\equiv 0$, d.h. es gibt keinen Advektionsterm in der Gleichung. In diesem Fall erhalten wir mit dem obigen Beweis nur $a(u,u)\geq\alpha\|\nabla u\|_{L_2}^2$, woraus wir die gewünschte Ungleichung $a(u,u)\geq\alpha\|u\|_{H^1}^2$ nicht folgern können (z.B. für $u\equiv u_0\neq 0$ ist $\|\nabla u\|_{L_2}^2$ gleich Null, $\|u\|_{H^1}^2=|u_0|^2$ aber positiv).

In diesem Fall können wir mit dem nachfolgenden Lemma aber noch Elliptizität in H_0^1 erhalten.

Lemma 12.5 (Poincaré-Friedrichs-Ungleichung) Sei Ω in einer Menge der Form

$$S=\left\{x\in\mathbb{R}^n\,|\,x_i\in[-s,s],x_j\in\mathbb{R}\text{ für alle }j\in\{1,\ldots,n\}\setminus\{i\}\right\}$$

für ein $s>0$ und ein $i\in\{1,\ldots,n\}$ enthalten. Dann gilt für alle $u\in H_0^1(\Omega)$

$$\|u\|_{L_2(\Omega)}^2\leq 2s^2\|\nabla u\|_{L_2(\Omega)}^2.$$

Beweis: O.B.d.A. sei $i=1$. Wir schreiben $\hat{x}=(x_2,\ldots,x_n)^T$ und $u(x_1,\hat{x})=u(x)$. Dann gilt nach dem Hauptsatz der Differential- und Integralrechnung für $u\in D(\Omega)$

$$u(x_1,\hat{x})=u(x_1,\hat{x})-\underbrace{u(-s,\hat{x})}_{=0}=\int_{-s}^{x_1}\partial_1 u(y,\hat{x})dy.$$

Daraus folgt

$$\begin{aligned}|u(x_1,\hat{x})|^2 &= \left|\int_{-s}^{x_1}\partial_1 u(y,\hat{x})dy\right|^2\\ &\leq \left(\int_{-s}^{x_1}|\partial_1 u(y,\hat{x})|dy\right)^2\\ &\leq (x_1+s)\int_{-s}^{x_1}|\partial_1 u(y,\hat{x})|^2dy\\ &\leq (x_1+s)\int_{-s}^{x_1}\|\nabla u(y,\hat{x})\|^2dy\end{aligned}$$

und somit

$$\begin{aligned}\|u\|^2_{L_2(\Omega)} &= \|u\|^2_{L_2(S)} = \int_{\mathbb{R}^{n-1}}\int_{-s}^{s}|u(x_1,\hat{x})|^2dx_1d\hat{x}\\ &\leq \int_{\mathbb{R}^{n-1}}\int_{-s}^{s}(x_1+s)\int_{-s}^{s}\|\nabla u(y,\hat{x})\|^2dydx_1d\hat{x}\\ &= \int_{-s}^{s}(x_1+s)\int_{-s}^{s}\int_{\mathbb{R}^{n-1}}\|\nabla u(y,\hat{x})\|^2d\hat{x}dydx_1\\ &= \frac{(2s)^2}{2}\|\nabla u\|^2_{L_2(S)} = 2s\|\nabla u\|^2_{L_2(\Omega)}.\end{aligned}$$

Da die Aussage damit für alle $u \in D(\Omega)$ gilt und $D(\Omega)$ dicht in $H^1_0(\Omega)$ liegt, folgt die Behauptung. □

Damit können wir nun die zweite Version des Hauptresultats beweisen.

Satz 12.6 Es sei $\mathcal{A} \in L_\infty(\Omega, \mathbb{R}^{n\times n})$ und $c \in L_\infty(\Omega)$. Zudem existiere $\tilde{\alpha} > 0$, so dass $\mathcal{A}(x) \geq \tilde{\alpha}\mathrm{Id}$ gilt für alle $x \in \Omega$ und es sei $c(x) \geq 0$ für alle $x \in \Omega$.

Dann ist a aus (12.4) elliptisch und stetig auf dem Raum $H^1_0(\Omega)$, d.h. die Voraussetzungen des Satzes von Lax-Milgram (Satz 12.1) sind erfüllt.

Beweis: Stetigkeit folgt wie im Beweis von Satz 12.4. Die Elliptizität folgt mit Lemma 12.5 (für passendes s, das existiert, weil Ω beschränkt ist) und (12.8) aus der Ungleichung

$$a(u,u) \geq \tilde{\alpha}\|\nabla u\|^2_{L_2} = \frac{\tilde{\alpha}}{2}\|\nabla u\|^2_{L_2} + \frac{\tilde{\alpha}}{2}\|\nabla u\|^2_{L_2} \geq \frac{\tilde{\alpha}}{2}\|\nabla u\|^2_{L_2} + \frac{\tilde{\alpha}}{4s^2}\|u\|^2_{L_2} \geq \alpha\|u\|^2_{H^1},$$

mit $\alpha = \min\{\tilde{\alpha}/2, \tilde{\alpha}/4s^2\}$. □

Aus Satz 12.6 folgt, dass die Poisson-Gleichung mit Randbedingung $u = 0$ eine eindeutige schwache Lösung $u_* \in H^1_0(\Omega)$ besitzt: Die schwache Form der Gleichung lautet

$$\int_\Omega \nabla u \cdot \nabla v - fvdx = 0 \text{ für alle } v \in H^1_0(\Omega)$$

also ist

$$a(u,v) = \int_\Omega \nabla u \cdot \nabla v dx$$

elliptisch mit $\alpha = 1$. Die Lösung erfüllt also $\|u\|_{H^1} \leq \|f\|_{(H_1)^*} \leq \|f\|_{L_2}$.

Die Vorgehensweise in diesem Abschnitt kann auf weitere Randbedingungen erweitert werden, worauf wir hier aus Zeitgründen aber verzichten.

12.3 Das Ritz-Galerkin Verfahren

Die Methode der finiten Elemente beruht entscheidend auf dem Konzept der schwachen Lösung und dem Minimierungsproblem (12.6). Die Grundidee ist, dieses Problem nur auf einem endlichdimensionalen Unterraum zu lösen.

Statt $\mathcal{E}(u)$ über $u \in V$ (also z.B. $V = H^1(\Omega)$ oder $V = H_0^1(\Omega)$) zu minimieren, lösen wir

$$\min_{u_h \in V_h} \mathcal{E}(u_h) \tag{12.9}$$

für einen endlichdimensionalen Teilraum $V_h \subset V$. Dies ist das sogenannte *Ritz-Verfahren* und die Funktionen in V_h werden *Ansatzfunktionen* genannt.

Alternativ kann man die notwendigen Optimalitätsbedingungen (12.5) auf einem Unterraum lösen. Statt ein $u \in V$ zu suchen, das $D\mathcal{E}(u)v = a(u,v) - l(v) = 0$ für alle $v \in V$ erfüllt, löst man das Problem

$$\text{Finde } u_h \in V_h \text{ mit } a(u_h, v_h) - l(v_h) = 0 \text{ für alle } v_h \in V_h. \tag{12.10}$$

Diesen Ansatz nennt man *Galerkin-Verfahren*. Beachte, dass hier sowohl das u_h als auch die Testfunktionen v_h aus V_h gewählt werden.

Für elliptische PDGen führen beide Ansätze auf die selbe Lösung u_h, da der Zusammenhang zwischen der Minimierung von $\mathcal{E}$ und den Optimalitätsbedingungen auf V_h mit den gleichen Rechnungen wie für V gezeigt werden kann. Die Probleme (12.9) und (12.10) sind also äquivalent. Daher spricht man vom *Ritz-Galerkin-Verfahren*.

Wie klären nun, wie man die äquivalenten Probleme (12.9) und (12.10) in eine Form bringen kann, so dass man sie mit dem Computer lösen kann. Dazu sei $\phi_1, \dots, \phi_N$ eine Basis von V_h (wie diese in der Praxis aussieht, klären wir im nächste Abschnitt). Die einzelnen ϕ_j heißen dann Basisfunktionen.

Die Gleichung (12.10) ist nun genau dann für alle $v_h \in V_h$ erfüllt, wenn sie für alle Basiselemente von V_h erfüllt ist, also wenn

$$a(u_h, \phi_j) - l(\phi_j) = 0 \text{ für alle } j = 1, \dots, N$$

gilt. Schreiben wir die gesuchte Funktion u_h nun als

$$u_h(x) = \sum_{i=1}^{N} \underline{u}_i \phi_i(x),$$

wo ergibt sich dies zu

$$\sum_{i=1}^{N} \underline{u}_i a(\phi_i, \phi_j) - l(\phi_j) = 0 \text{ für alle } j = 1, \dots, N.$$

Dies können wir mit $\underline{u} = (\underline{u}_1, \dots, \underline{u}_N)^T$, $A = (a(\phi_i, \phi_j))_{i,j=1,\dots,N}$ und $b = (l(\phi_1), \dots, l(\phi_N))^T$ kurz als

$$A\underline{u} = b \tag{12.11}$$

schreiben. Wir erhalten also ein lineares Gleichungssystem. Die Matrix A darin wird oft als *Steifigkeitsmatrix* bezeichnet. Dabei ist $A \in \mathbb{R}^{N \times N}$ symmetrisch, weil a symmetrisch ist und $A > 0$ weil a elliptisch ist. A ist also eine symmetrische und positiv definite (spd) Matrix. Wir können A^{-1} als Abbildung von l nach u_h auffassen, also von $(H^1)^*$ nach

H^1. Aus der Abschätzung $\alpha\|u_h\|_{H^1}^2 \leq a(u_h,u_h) = l(u_h) \leq \|l\|_{(H^1)^*}\|u_h\|_{H^1}$ folgt dann $\alpha\|u_h\|_{H^1} \leq \|l\|_{(H^1)^*}$ und damit[4]

$$\|A^{-1}\|_{(H^1)^*\to H^1} \leq \frac{1}{\alpha}.$$

Explizit ausgeschrieben gilt

$$a(\phi_i,\phi_j) = \int_\Omega (\mathcal{A}(x)\nabla\phi_i(x))\cdot\nabla\phi_j(x) + c(x)\phi_i(x)\phi_j(x)dx.$$

Wählt man die Basisfunktionen ϕ_i so, dass sie kleine Träger haben, die sich nur für wenige i,j-Kombinationen überlappen, so wird das Integral für viele i,j-Kombinationen zu Null. Die Matrix A ist in diesem Fall dünn besetzt, was die numerische Lösung von $A\underline{u} = b$ deutlich vereinfacht.

Das folgende Lemma klärt, wie gut u_h die exakte Lösung u im Vergleich zur bestmöglichen Approximation in V_h approximiert.

Lemma 12.7 (Céa) Es sei a stetig und elliptisch mit Konstanten C und α. Dann gilt für die Lösung u_h von (12.9) und (12.10)

$$\|u-u_h\|_{H^1} \leq \sqrt{\frac{C}{\alpha}} \inf_{\hat{u}_h\in V_h} \|u-\hat{u}_h\|_{H^1}.$$

Beweis: Wegen $V_h \subset V$ folgt $a(u,v_h) = l(v_h)$ für alle $v_h \in V_h$, also auch für u_h. Zudem minimiert u_h nach Definition $\mathcal{E}$ über V_h. Damit und mit der Symmetrie von a folgt für alle $v_h \in V_h$

$$\begin{aligned}
\alpha\|u-u_h\|_{H^1}^2 &\leq a(u-u_h,u-u_h)\\
&= a(u,u) - 2\underbrace{a(u,u_h)}_{=l(u_h)} + a(u_h,u_h)\\
&= a(u,u) + 2\mathcal{E}(u_h)\\
&\leq a(u,u) + 2\mathcal{E}(v_h)\\
&= a(u,u) - 2a(u,v_h) + a(v_h,v_h)\\
&= a(u-v_h,u-v_h) \leq C\|u-v_h\|_{H^1}^2
\end{aligned}$$

Division durch α, Wurzelziehen und Bilden des Infimums über v_h liefert die Behauptung. □

Für eine Folge von Unterräumen V_{h_i}, $i\in\mathbb{N}$, für die $\bigcup_{i\in\mathbb{N}} V_{h_i}$ dicht in V liegt, folgt also

$$u_{h_i} \to u$$

in H^1 für $i\to\infty$.

[4]Die Operatornorm für einen Operator $A: V_1 \to V_2$ ist hierbei wie üblich definiert als $\|A\|_{V_1\to V_2} := \sup_{v\in V_1,\|v\|_{V_1}=1}\|Av\|_{V_2}$.

12.4 Wahl der Ansatz- und Basisfunktionen

Es bleibt zu klären, wie die Ansatzfunktkionen V_h und die zugehörigen Basisfunktionen ϕ_i gewählt werden können. Dies geschieht in der Finite-Elemente-Methode in der Regel auf Basis einer Triangulierung, also einer Zerlegung von Ω in kleine Teilgebiete, die im zweidimensionalen Fall oft als Dreiecke gewählt werden — daher der Name Triangulierung.

Definition 12.8 Eine Triangulierung $\mathcal{T}$ ist eine Zerlegung von Ω in Teilgebiete T_i, $i = 1, \ldots, M$, mit den folgenden Eigenschaften

- Jedes $T_i \in \mathcal{T}$ ist ein abgeschlossener, zusammenhängender Polyeder (z.B. Dreieck, Rechteck, Tetraeder, Quader, ...)
- $\overline{\Omega} = \bigcup_{i=1}^{M} T_i$
- $\operatorname{int} T_i \cap \operatorname{int} T_j = \emptyset$ für alle $i \neq j$
- $T_i \cap T_j$ ist entweder leer oder ein Polyeder von niedrigerer Dimension als Ω.

Den maximalen Abstand zweier Punkte in T_i bezeichnen wir mit

$$\operatorname{diam} T_i := \max_{x,y \in T_i} \|x - y\|$$

(gesprochen: Durchmesser von T_i). Den maximalen Durchmesser der Triangulierung bezeichnen wir mit

$$h := \max_{i=1,\ldots,M} \operatorname{diam} T_i.$$

Wir schreiben $\mathcal{T}_h$, wenn wir den Durchmesser der Triangulierung in der Notation hervorheben wollen und h_T, wenn wir den Zusammenhang von h mit der Triangulierung explizit betonen wollen. □

Sind die Teilgebiete T_i in etwa gleich groß, so ist der Durchmesser h proportional zu $M^{-1/n}$. Je größer die Dimension, desto mehr Teilgebiete braucht mal also, um eine vorgegebene ober Schranke des Durchmessers zu realisieren.

Das Erstellen einer Triangulierung ist ein Problem für sich, das wir hier nicht behandeln wollen. Es gibt für viele Probleme heutzutage Standardsoftware, mit der man Triangulierungen konstruieren kann.

Wir erläutern nun am Beispiel des Raums $V_h \subset V = H^1(\Omega)$, wie man mit Hilfe der Triangulierung die Basisfunktionen ϕ_i konstruiert. Dazu verwenden wir den folgenden Satz, der aus [2, Satz 5.2] folgt.

Satz 12.9 Gegeben sei eine Funktion $v \in C(\overline{\Omega})$, so dass $v|_{T_i}$ für jedes Teilgebiet T_i stetig differenzierbar ist. Dann ist $v \in H^1(\Omega)$.

Aus diesem Satz folgt, dass für eine gegebene Triangulierung $\mathcal{T}$ z.B. die folgenden Räume endlichdimensionale Teilräume von $H^1(\Omega)$ bilden:

Polynome vom Grad l auf Dreiecken oder Tetraedern

Die T_i sind Simplizes (also Dreiecke in $\mathbb{R}^2$, Tetraeder in $\mathbb{R}^3$ etc.) und

$$V_h := \{v \in C(\overline{\Omega}) \,|\, v|_{T_i} \text{ ist ein Polynom vom Grad } \leq l \text{ für jedes } T_i \in \mathcal{T}\}$$

Für $l = 1, 2$ und 3 spricht man von linearen, quadratischen oder kubischen Elementen. Falls die Polynome vorgegebene Werte in Stützstellen (man spricht auch von Knoten) interpolieren, so spricht man von Lagrange-Elementen. Werden Werte und Ableitungen in vorgegebenen Knoten interpoliert, spricht man von Hermite-Elementen.

Multilineare Finite Elemente auf Quadern

Die T_i sind Quader (also Rechtecke in $\mathbb{R}^2$, "übliche" Quader im $\mathbb{R}^3$ etc.) und

$$V_h := \{v \in C(\overline{\Omega}) \,|\, v|_{T_i} \text{ ist ein Polynom und linear auf den Kanten für jedes } T_i \in \mathcal{T}\}$$

Diese Elemente haben den Vorteil, dass sie sich leicht in beliebigen Dimensionen implementieren lassen.

Beispiel 12.10 Wir betrachten den Raum V_h der linearen Elemente auf Dreiecken. Dann ist ein $v_h \in V_h$ auf jedem Element T_i eine lineare Funktion, zudem ist sie auf ganz Ω stetig. Gibt man sich Werte w_j in den Eckpunkten p_j, $j = 1, \ldots, P$ der Dreiecke T_i vor, so sieht man, dass es genau eine Funktion $v_h \in V_h$ gibt, für die $v_h(p_j) = w_j$ für alle Knoten p_j gilt:

Dass es höchstens eine solche Funktion geben kann, sieht man daran, dass die Differenz $v_h^1 - v_h^2$ zweier solcher Funktionen auf jedem Dreieck eine lineare Funktion ist, die an den Eckpunkten den Wert 0 annimmt. Das kann aber nur die Nullfunktion sein, also folgt $v_h^1 = v_h^2$ auf jedem Dreieck, also auf ganz Ω.

Dass es eine solche Funktion gibt sieht man, indem man für jeden Knotenpunkt p_j die Funktion ϕ_j betrachtet, die die Bedingungen

- $\phi_j(p_j) = 1$
- $\phi_j(p_k) = 0$ für alle $p_k \neq p_j$
- $\phi_j|_{T_i}$ ist linear auf jedem Dreieck T_i
- ϕ_j ist stetig auf $\overline{\Omega}$

erfüllt (ϕ_j ist eine sogenannte "Hütchenfunktion"; wenn man versucht, die Funktion durch eine Skizze zu veranschaulichen, sieht man, warum. Wir werden gleich zeigen, dass eine solche Funktion tatsächlich existiert).

Definieren wir nun die Funktion $v_h(x) := \sum_{k=1}^{P} w_k \phi_k(x)$, so ist diese ebenfalls linear auf jedem Dreieck, stetig auf $\overline{\Omega}$ und erfüllt die Interpolationsbedingung $v_h(p_j) = w_j$. □

12.5 Implementierung

Um die Basisfunktionen im Rechner darzustellen, werden diese aus sogenannten Formfunktionen zusammengesetzt, die auf den einzelnen Teilgebieten definiert sind. Diese Formfunktionen wiederum werden zunächst auf einem Standardteilgebiet $\widehat{T}$ definiert. Diese Definition wird dann mittels einer affin linearen Transformation auf alle T_i übertragen.

Wir beschreiben dieses Vorgehen auf Simplexgittern. Der Standardsimplex in $\mathbb{R}^n$ ist dabei gegeben durch

$$\widehat{T} := \{x \in \mathbb{R}^n \,|\, x_i \geq 0 \text{ für alle } 1 = 1, \ldots, n \text{ und } \sum_{i=1}^{n} x_1 \leq 1\}.$$

In $\mathbb{R}^2$ ist dies gerade das Dreieck mit den Eckpunkten $(0,0)$, $(1,0)$ und $(0,1)$. Im $\mathbb{R}^3$ ist $\widehat{T}$ das Tetraeder mit den Eckpunkten $(0,0,0)$, $(1,0,0)$, $(0,1,0)$ und $(0,0,1)$. Wir bezeichnen die Eckpunkte des Simplex mit $V_0, \ldots, V_n$; V_0 ist dabei der Nullpunkt und V_k für $k \neq 0$ der Eckpunkt mit der 1 an der i-ten Stelle.

Auf $\widehat{T}$ definieren wir nun die Formfunktionen $\beta_{\widehat{T},\eta}$, wobei η für einen oder mehrere Indizes steht. Wir beginnen zunächst mit linearen Formfunktionen. Diese sind gegeben durch die baryzentrischen Koordinaten

$$\lambda_0(x) = 1 - \sum_{i=1}^{n} x_i, \quad \lambda_k = x_k \text{ für } k = 1, \ldots, n.$$

Setzen wir die Formfunktionen nun als

$$\beta_{\widehat{T},k} := \lambda_k, \;\; k = 0, \ldots, \tag{12.12}$$

so rechnet man leicht nach, dass die Beziehung

$$\beta_{\widehat{T},k}(x) = \begin{cases} 1, x = V_k \\ 0, x = V_j, j \neq k \end{cases}$$

gilt. Für vorgegebene Werte w_k in den Ecken V_k ist also $u(x) = \sum_{k=0}^{n} w_k \beta_{\widehat{T},k}(x)$ eine lineare Funktion auf $\widehat{T}$ mit $u(V_k) = p_k$ für alle $k = 0, \ldots, n$.

Quadratische Formfunktionen können nun aus den linearen Formfunktionen konstruiert werden. Eine quadratische Funktion auf $\widehat{T}$ ist eindeutig bestimmt durch ihre Werte in den Eckpunkten V_i und in den Mittelpunkten M_{ij} zwischen je zwei Eckpunkten V_i und V_j (das müsste man natürlich beweisen, was aber nicht sehr schwer ist). Definieren wir nun die Formfunktionen

$$\beta_{\widehat{T},k}(x) := \lambda_i(x)(2\lambda_k(x) - 1), k = 0, \ldots, n, \quad \beta_{\widehat{T},kl}(x) := 4\lambda_k(x)\lambda_l(x), k, l = 0, \ldots, n, k < l$$

so rechnet man nach, dass

$$\beta_{\widehat{T},k}(x) = \begin{cases} 1, x = V_k \\ 0, x = V_j, j \neq k \\ 0, x = M_{kl} \end{cases} \text{ und } \beta_{\widehat{T},kl}(x) = \begin{cases} 1, x = M_{kl} \\ 0, x = M_{ij}, (i,j) \neq (k,l) \\ 0, x = V_k \end{cases} .$$

Geben wir also Werte w_k und w_{kl} in den Punkten V_k und M_{kl} vor, so erhalten wir die eindeutige quadratische Funktion auf $\widehat{T}$, die diese Werte interpoliert, mittels

$$u(x) = \sum_{k=0}^{n} w_k \beta_{\widehat{T},k}(x) + \sum_{\substack{k,l=0 \\ k<l}}^{n} w_{kl} \beta_{\widehat{T},kl}(x).$$

Ähnlich kann man kubische oder noch höhergradige Formfunktionen definieren.

Um die Formfunktionen nun auf die Teilgebiete T_i der Triangulierung zu übertragen, nehmen wir an, dass wir den Standardsimplex durch Transformationen auf die T_i abbilden lassen. In der Praxis werden oft affin lineare Transformationen verwendet, auf die wir uns hier beschränken wollen. Wir nehmen also an, dass invertierbare Matrizen $B_i \in \mathbb{R}^{n\times n}$ und Vektoren $b_i \in \mathbb{R}^n$ existieren mit

$$T_i = B_i \widehat{T} + b_i =: F_i(\widehat{T}),$$

wobei $B_i\widehat{T} := \{B_i x \,|\, x \in \widehat{T}\}$ ist. Dann gilt $F_i^{-1}(x) = B_i^{-1}x - B_i^{-1}b_i$. Dann definieren wir die Formfunktionen auf T_i mittels

$$\beta_{T_i,\eta}(x) = \beta_{\widehat{T},\eta}(F_i^{-1}(x)). \tag{12.13}$$

Jede Basisfunktion ϕ_j ist nun eine Kombination von verschiedenen Formfunktionen. Welche Formfunktionen genau eine Basisfunktion ergeben, hängt von der Art der Formfunktionen, der Form der Teilgebiete etc. ab. Wir illustrieren diese Konstruktion für die linearen Basisfunktionen ϕ_j auf Dreiecken im $\mathbb{R}^2$ aus Beispiel 12.10.

Beispiel 12.11 Wir betrachten wieder den Fall aus Beispiel 12.10. Wir zeigen, wie die dort verwendeten Basisfunktionen ϕ_j aus den Formfunktionen $\beta_{T_i,k}$ aus (12.12), (12.13) zusammengesetzt werden können. Sei also ein Knoten x_j der Triangulation gegeben. Dann gibt es eine gewisse Anzahl von Dreiecken $T_{i_1}, \ldots, T_{i_q}$, so dass der Knoten p_j mit einem Eckpunkt V_{j_r} von T_{i_r}, $r = 1, \ldots, q$ übereinstimmt. Wir setzen nun die Basisfunktion wie folgt aus den Formfunktionen zusammen:

$$\phi_j(x) := \begin{cases} \beta_{T_{i_r},j_r}(x), & \text{falls } x \in T_{i_r} \text{ für ein } r = 1, \ldots, q \\ 0, & \text{sonst} \end{cases}$$

Die Funktion ist wohldefiniert, obwohl der Wert auf der Rechten Seite für Punkte x, die in mehreren Dreiecken liegen, zunächst nicht eindeutig definiert ist. Wenn zwei Dreiecke aber eine nichtleere Schnittmenge haben, so stimmen die Werte der betreffenden Formfunktionen per Konstruktion in allen Eckpunkten in der Schnittmenge überein. Da die Formfunktionen linear sind, stimmen die Werte also auch auf eventuellen gemeinsamen Kanten überein. Daher ist die Funktion ϕ_j wohldefiniert und stetig. Per Konstruktion ist sie zudem linear auf jedem Dreieck und erfüllt $\phi_j(p_k) = 1$ genau dann, wenn $k = j$ ist. Also ist sie die gesuchte Basisfunktion. □

Ähnliche Konstruktionen wie in diesem Beispiel lassen sich für alle gebräuchlichen Basisfunktionen angeben. Im Rechner werden dazu für jedes Teilgebiet T_i und jede Basisfunktion

ϕ_j Indizes $\eta(i,j)$ gespeichert, so dass $\phi_j|_{T_i}(x) = \beta_{\widehat{T},\eta(i,j)}(F_i^{-1}(x))$ gilt. Dafür muss formal noch eine Null-Formfunktion eingeführt werden, für den Fall dass $\phi_j|_{T_i} \equiv 0$ gilt; diese können wir bei allen Berechnungen im Folgenden aber wieder weglassen, weil sie sowieso keinen Beitrag leistet. Damit kann auf jedem Teilgebiet T_i die Basisfunktion und ihre Ableitung mittels

$$\phi_j(x) = \beta_{\widehat{T},k}(F_i^{-1}(x)) \quad \text{und} \quad D\phi_j(x) = D\beta_{\widehat{T},k}(F_i^{-1}(x))B_i^{-1}$$

ausgewertet werden.

Zum Aufstellen des lineare Gleichungssystems (12.11) — man nennt diesen Vorgang auch *Assemblierung* — benötigen wir nun die Größen

$$a_{ij} = a(\phi_i,\phi_j) = \int_\Omega (\mathcal{A}(x)\nabla\phi_i(x))\cdot\nabla\phi_j(x) + c(x)\phi_i(x)\phi_j(x)dx, \qquad b_j = \int_\Omega f(x)\phi_j(x)dx.$$

Diese werden gemäß dem folgenden Algorithmus durch Iteration über alle Zellen berechnet.

Algorithmus 12.12 (0) Setze $a_{ij} := 0$, $b_j := 0$ für alle $i,j = 1,\dots,N$.

(1) Für alle Teilgebiete T_s:

(2) Für alle Basisfunktionen ϕ_i:

(3) Berechne $\Delta b_i := \int_{T_s} f(x)\phi_i(x)dx$

(4) Setze $b_i := b_i + \Delta b_i$

(5) Für alle Basisfunktionen ϕ_j:

(6) Berechne $\Delta a_{ij} := \int_{T_s}(\mathcal{A}(x)\nabla\phi_i(x))\cdot\nabla\phi_j(x) + c(x)\phi_i(x)\phi_j(x)dx$

(7) Setze $a_{ij} := a_{ij} + \Delta a_{ij}$

(8) Ende aller Schleifen □

Mithilfe der obigen Ausdrücke für ϕ_j und die Ableitungen $D\phi_j = (\nabla\phi_j)^T$ sowie der Transformationsregel können wir die Integrale im Algorithmus direkt mit den Formfunktionen über $\widehat{T}$ formulieren. Es gilt

$$\int_{T_s} f(x)\phi_i(x)dx = \int_{\widehat{T}} f(F_i(x))\beta_{\widehat{T},k(i,s)}(x)\det B_i dx$$

und

$$\begin{aligned}&\int_{T_s}(\mathcal{A}(x)\nabla\phi_i(x))\cdot\nabla\phi_j(x) + c(x)\phi_i(x)\phi_j(x)dx\\ &= \int_{\widehat{T}}(\mathcal{A}(F_i(x))(D\beta_{\widehat{T},l}(F_i^{-1}(x))B_i^{-1})^T\cdot(D\beta_{\widehat{T},k}(F_i^{-1}(x))B_i^{-1})^T\det B_i\\ &\quad + c(F_i(x))\beta_{\widehat{T},k(i,s)}(x)\beta_{\widehat{T},k(j,s)}(x)\det B_i dx.\end{aligned}$$

Alternativ kann man die Schleifen auch über die Formfunktionen laufen lassen. Dann muss man zu jedem Paar aus Formfunktion $\beta_{\widehat{T},\eta}$ und Teilgebiet T_s den Index $j(s,\eta)$ der zugehörigen Basisfunktion $\phi_{j(s,\eta)}$ speichern.

Zum Lösen des linearen Gleichungssystems kann nun ein Verfahren zum numerischen Lösen von linearen Gleichungssystemen mit großen, dünn besetzten spd Matrizen verwendet werden. Beispielsweise eignet sich das CG-Verfahren aus der Vertiefung der Numerik. Es gibt aber auch Varianten des Choleski-Verfahrens für dünn besetzte Matrizen.

Wir erinnern daran, dass die Konvergenz des CG-Verfahrens durch die Konditionszahl $\kappa_{A.M} = \Gamma/\gamma$ bestimmt ist, wobei $0 < \gamma < \Gamma$ untere und obere Schranken der Eigenwerte von $M^{-1}A$ sind. Je kleiner $\kappa_{A,M}$ ist, desto schneller konvergiert das Verfahren. Ohne Vorkonditionierer — also mit $M = \mathrm{Id}$ — ist $\kappa_{A,\mathrm{Id}} = O(h^{-2}) = O(M^{-2/n})$ (die letzte Gleichung gilt, wenn die Teilgebiete in etwa geich groß sind). → `Übung` Je feiner man also diskretisiert, desto mehr Iterationen benötigt das CG-Verfahren. Da bei abnehmendem h zugleich die Matrizen größer werden, nimmt die Rechenzeit also stark zu. Daher ist es wichtig, einen guten Vorkonditionierer zu finden, mit dem die Kondition von $M^{-1}A$ verringert wird. Wir kommen in Kapitel 14 darauf zurück.

Kapitel 13

Fehleranalyse

Wir analysieren nun den Fehler der im letzten Kapitel vorgestellten Methode. Die Analyse beruht auf dem Lemma von Céa (Lemma 12.7) und der folgenden Analyse des Fehlers für Interpolationspolynome, mit der wir die rechte Seite der Ungleichung im Céa-Lemma abschätzen können.

13.1 Interpolationsfehler

Wir analysieren nun zunächst den Interpolationsfehler auf dem Standardgebiet $\widehat{T}$. Für die Analyse benötigen wir die H^m-Norm

$$\|v\|_m := \sum_{|\alpha| \leq m} \|\partial_\alpha v\|_{L_2}$$

und die H^m-Halbnorm

$$|v|_m := \sum_{|\alpha| = m} \|\partial_\alpha v\|_{L_2}.$$

Beachte dass aus diesen Definitionen $\|v\|_m = \|v\|_{m-1} + |v|_m$ folgt. Wenn es wichtig ist, das zu Grunde liegenden Gebiet Ω zu betonen, schreiben wir auch $\|v\|_{H^m(\Omega)}$ statt $\|v\|_m$ und analog $|v|_{H^m(\Omega)}$ statt $|v|_m$.

Wir betrachten im Folgenden den Raum $\mathcal{P}_{m-1}$ der Interpolationspolynome[1] auf $\mathbb{R}^n$ vom Grad $\leq m-1$ für ein $m \in \mathbb{N}$, $m \geq 2$. Wir nennen $x_i \in \widehat{T}$, $i = 0, \ldots, q$, *gültige Interpolationsknoten*, wenn für alle $v_i \in \mathbb{R}$, $i = 0, \ldots, q$ genau ein $p \in \mathcal{P}_{m-1}$ existiert mit $p(x_i) = v_i$ für alle $i = 0, \ldots, q$.

Lemma 13.1 (Bramble-Hilbert-Lemma) Seien $x_i \in \widehat{T}$, $i = 0, \ldots, q$, gültige Interpolationsknoten für $\mathcal{P}_{m-1}$. Sei außerdem $\Omega \subset \mathbb{R}^n$ ein Gebiet mit Lipschitzrand und $\widehat{T} \subseteq \Omega$.

[1] Einige der folgenden Aussagen lassen sich auf den Fall $m = 1$ verallgemeinern, was wir hier aber nicht durchführen wollen.

Zudem gelte $m \geq n/2$. Dann gibt es eine Konstante $c > 0$, so dass für alle $v \in C(\overline{\Omega}) \cap H^m(\Omega)$ die Ungleichung

$$\|v\|_m \leq c \left(|v|_m + \sum_{i=0}^{q} |v(x_i)| \right)$$

gilt.

Beweis: Aus den Annahmen an Ω folgt die kompakte Einbettung $H^m(\Omega) \hookrightarrow H^{m-1}$ und wegen $m > n/2$ gilt zudem die Einbettung $H^m(\Omega) \hookrightarrow C(\overline{\Omega})$.

Angenommen, solch eine Konstante c existiert nicht. Dann gibt es für jedes $c > 0$ ein $\tilde{v}_c$, so dass

$$\|\tilde{v}_c\|_m > c \left(|\tilde{v}_c|_m + \sum_{i=0}^{q} |\tilde{v}_c(x_i)| \right).$$

Für $c = k$, $k \in \mathbb{N}$, setzen wir nun $v_k := \tilde{v}_c / \|\tilde{v}_c\|_m$. Dann gilt $\|v_k\|_m = 1$ und

$$|v_k|_m + \sum_{i=0}^{q} |v_k(x_i)| \leq \frac{1}{k} \|v_k\|_m = \frac{1}{k} \to 0$$

für $k \to \infty$. Da H^m in H^{m-1} kompakt eingebettet ist, besitzt v_k eine in H^{m-1} konvergente Teilfolge, die wir wieder mit v_k bezeichnen, und für die wie bisher $|v_k|_m \to 0$ gilt. Den Grenzwert bezeichnen wir mit v^*.

In H^m gilt nun

$$\|v_k - v_l\|_m = \|v_k - v_l\|_{m-1} + |v_k - v_l|_m \leq \|v_k - v_l\|_{m-1} + |v_k|_m + |v_l|_m \to 0$$

für $k, l \to \infty$. Daraus folgt, dass v_k eine Cauchyfolge in H^m ist und daher auch in H^m konvergiert. Für den Grenzwert gilt dabei $|v^*|_m = 0$, also ist v^* ein Polynom vom Grad $\leq m-1$. Da v_k gleichmäßig gegen die stetige Funktion v_k konvergiert, konvergiert sie auch punktweise und es folgt $v^*(x_i) = 0$ für alle x_i. Nach Annahme an die Stützstellen gilt also $v^* \equiv 0$ und damit $\|v^*\|_m = 0$. Dies ist aber ein Widerspruch, da eine Folge v_k mit $\|v_k\|_m = 1$ für alle $k \in \mathbb{N}$ kann nicht gegen einen Grenzwert mit $\|v^*\|_m = 0$ konvergieren. □

Korollar 13.2 Betrachte $\widehat{T}$, Ω und gültige Interpolationsknoten $x_i \in \widehat{T}$ wie in Lemma 13.1. Dann ist die Polynominterpolation $I : C(\overline{\Omega}) \cap H^m \to H^m$, die jedem $v \in C(\overline{\Omega}) \cap H^m$ das Interpolationspolynom $p \in \mathcal{P}_{m-1} \subset H^m$ mit $p(x_i) = v(x_i)$ für alle x_i zuordnet, eine stetige lineare Abbildung.

Beweis: Die Linearität folgt sofort aus der leicht verifizierbaren Tatsache, dass $\lambda_1 p_1 + \lambda_2 p_2$ gerade das Interpolationspolynom zu $\lambda_1 v_1 + \lambda_2 v_2$ ist, wenn p_1 und p_2 die Funktionen v_1 und v_2 interpolieren.

Zum Nachweis der Stetigkeit verwenden wir, dass eine lineare Abbildung genau dann stetig ist, wenn eine Konstante $C > 0$ mit $\|Iv\|_m \leq C\|v\|_m$ für alle $v \in H^m$ existiert. Dies gilt

für die Interpolation aber, denn mit Lemma 13.1 folgt

$$\begin{aligned}\|Iv\|_m &\leq \|v\|_m + \|v - Iv\|_m \\ &\leq \|v\|_m + c\left(|v - Iv|_m + \sum_{i=0}^{q} |\underbrace{v(x_i) - I(v)(x_i)}_{=0}|\right) \\ &= \|v\|_m + c|v|_m \leq (1+c)\|v_m\|.\end{aligned}$$

□

Satz 13.3 Betrachte $\widehat{T}$, Ω und gültige Interpolationsknoten $x_i \in \widehat{T}$ wie in Lemma 13.1. Sei $L : H^m(\Omega) \to Y$ eine stetige lineare Abbildung, so dass $P_{m-1} \subset \ker L$. Dann existiert eine Konstante $c > 0$ mit

$$\|Lv\|_Y \leq c|v|_m$$

für alle $v \in H^m$.

Beweis: Sei $I : C(\overline{\Omega}) \cap H^m \to \mathcal{P}_{m-1}$ der Interpolationsoperator zu den Knoten x_i. Dann gelten für alle $v \in C(\overline{\Omega}) \cap H^m$ die Gleichungen $LIv = 0$ sowie $|v - Iv|_m = |v|_m$ und damit

$$\begin{aligned}\|Lv\|_Y &= \|L(v - Iv)\|_Y \leq \|L\|\,\|v - Iv\|_m \\ &\leq c\|L\|c\left(|v - Iv|_m + \sum_{i=0}^{q} |\underbrace{v(x_i) - I(v)(x_i)}_{=0}|\right) = c\|L\|\,|v - Iv|_m \\ &= c\|L\|\,|v|_m\end{aligned}$$

wobei $\|L\|$ kurz für $\|L\|_{H^m \to Y}$ steht. Für allgemeines $v \in H^m$ folgt die Aussage dann aus der Stetigkeit der Normen, weil $C(\overline{\Omega})$ dicht in H^m liegt. □

Mit Hilfe von Satz 13.3 und der Tatsache, dass die Teilgebiete T_i von der Form $T_i = F_i(\widehat{T}) = B_i\widehat{T} + b_i$ sind, können wir nun eine Abschätzung des Interpolationsfehlers auf $\mathcal{T}$ herleiten. Dazu brauchen wir die folgende Transformationsformel.

Lemma 13.4 Sei $x = F_i(\hat{x}) = B_i\hat{x} + b_i$ und $\hat{v}(\hat{x}) = v \circ F_i(\hat{x})$. Dann gilt

$$|\hat{v}|^2_{H^k(\widehat{T})} \leq \|B_i\|^{2k} |\det B_i|^{-1} |v|^2_{H^k(T_i)}.$$

Beweis: Mit der Kettenregel gilt

$$\hat{v}^{(k)}(\hat{x})(\hat{v}_1, \ldots, \hat{v}_k) = (v \circ F_i)^{(k)}(\hat{x})(\hat{v}_1, \ldots, \hat{v}_k) = v^{(k)}(F_i(\hat{x}))(\hat{B}_i^k v_1, \ldots, \hat{B}_i^k v_k).$$

Daraus folgt

$$\|\hat{v}^{(k)}(\hat{x})\| \leq \|B_i\|^k \|v^{(k)}(F_i(\hat{x}))\| = \|B_i\|^k \|v^{(k)}(x)\|.$$

Damit gilt mit der Transformationsformel für Integrale

$$\int_{\widehat{T}} \|\hat{v}^{(k)}(\hat{x})\|^2 d\hat{x} \leq \|B_i\|^{2k} \int_{\widehat{T}} |v^{(k)}(F_i(\hat{x}))\|^2 d\hat{x} = \|B_i\|^{2k} \int_T |v^{(k)}(x)\|^2 |\det B_i|^{-1} dx.$$

□

Damit können wir nun den Hauptsatz zum Interpolationsfehler beweisen. Dafür benötigen wir die folgende Annahme, die sicher stellt, dass die in der Abschätzung auftretenden Konstanten nicht zu groß werden.

Definition 13.5 Eine Folge von Triangulierungen $\mathcal{T}_{h_l}$, $l \in \mathbb{N}$, heißt *formregulär*, wenn es eine Konstante κ gibt, so dass

$$\kappa(B_i) = \|B_i\| \, \|B_i^{-1}\| \leq \kappa$$

für alle $T_i \in T_{h_l}$ und alle $l \in \mathbb{N}$. □

Anschaulich bedeutet diese Bedingung z.B. im Falle von Dreiecksgebieten, dass die Winkel in den Dreiecken nicht beliebig spitz werden können. Zudem verwenden wir im Folgenden, dass die Ungleichung

$$\|B_i\| \leq \gamma \operatorname{diam} T_i$$

für alle T_i gilt, vorausgesetzt dass $\widehat{T}$ einen Ball $B_{1/\gamma}(\bar{x})$ mit $\gamma > 0$ enthält (was für alle sinnvollen Standardgebiete der Fall ist). Dies folgt aus der Rechnung

$$\begin{aligned} \|B_i\| &= \sup_{x \in B_1(0)} \|B_i x\| = \gamma \sup_{x \in B_{1/\gamma}(\bar{x})} \|B_i(x - \bar{x})\| \\ &= \gamma \sup_{x \in B_{1/\gamma}(\bar{x})} \|B_i x - B_i \bar{x})\| \leq \gamma \sup_{x_1, x_2 \in T_i} \|x_1 - x_2\| = \gamma \operatorname{diam}(T_i). \end{aligned}$$

Satz 13.6 Sei $\Omega \subset \mathbb{R}^n$ ein Gebiet mit Lipschitzrand, $m > n/2$ und $\mathcal{T}_{h_l}$ eine formreguläre Folge von Triangulierungen mit Konstante $\kappa > 0$. Sei $I_l : C(\overline{\Omega}) \cap H^m(\Omega) \to H^m(\Omega)$ der stückweise Interpolationsoperator auf den Teilgebieten T_i von $\mathcal{T}_{h_l}$ mit Interpolationsknoten $x_{i,j} = F_i(x_j)$, wobei die x_j gültige Interpolationsknoten auf dem Standardgebiet $\widehat{T}$ sind.

Dann existiert eine Konstante $C > 0$, so dass für alle $l \in \mathbb{N}$, alle $k = 0, \ldots, m$ und alle $v \in C(\overline{\Omega}) \cap H^m(\Omega)$ die Ungleichung

$$\|v - I_l v\|_k \leq C h_l^{m-k} |v|_m$$

gilt.

Beweis: Aus der Definition von $\|\cdot\|_m = \|\cdot\|_{H^m(\Omega)}$ folgt, dass $\|\cdot\|^2_{H^m(\Omega)}$ als Summe der Normen $\|\cdot\|^2_{H^m(T_i)}$ geschrieben werden kann. Es genügt daher zu zeigen, dass

$$\|v - Iv\|^2_{H^k(T_i)} \leq C h_l^{2(m-k)} |v|^2_{H^m(T_i)}$$

für alle Teilgebiete T_i der Triangulierung $\mathcal{T}_{h_l}$ gilt. Aufsummieren und Wurzelziehen liefert dann die gewünschte Ungleichung.

Zum Beweis der Ungleichung für $\|v - Iv\|^2_{H^k(T_i)}$ beweisen wir die entsprechende Ungleichung für $|v - Iv|^2_{H^j(T_i)}$. Aufsummieren über $j = 0, \ldots, k$ liefert dann die Behauptung. Um die Ungleichung für $|v - Iv|^2_{H^j(T_i)}$ zu beweisen, transformieren wir diese quadrierte Halbnorm

mit Lemma 13.4 auf das Standardgebiet $\widehat{T}$, wenden dort Satz 13.3 mit $L = \mathrm{Id} - \hat{I}$ an und transformieren die quadrierte Halbnorm $|\hat{v}|_{H^m(\widehat{T})}$ dann mit mit Lemma 13.4 mit $k = m$ zurück. Dabei ist $\hat{I}$ der Interpolationsoperator auf $\widehat{T}$. Für diesen gilt $\hat{I}\hat{v}(\hat{x}) = Iv \circ F_i(\hat{x})$ und damit

$$\begin{aligned} |v - Iv|^2_{H^k(T_i)} &\leq \|B_i^{-1}\|^{2j} |\det B_i^{-1}|^{-1} |\hat{v} - \hat{I}\hat{v}| \\ &\leq c\|B_i^{-1}\|^{2j} |\det B_i|\, |\hat{v}|^2_{H^m(\widehat{T})} \\ &\leq c\|B_i^{-1}\|^{2j} |\det B_i| c \|B_i\|^{2m} |\det B_i|^{-1} |v|^2_{H^m(T_i)} \\ &\leq c\kappa^{2j} \|B_i\|^{2(m-j)} |v|^2_{H^m(T_i)} \quad \leq \quad c\kappa^{2j}\gamma^{2(m-j)} h_l^{2(m-j)} |v|^2_{H^m(T_i)}. \end{aligned}$$

Die Behauptung folgt nun mit $C = \sum_{j=0}^{k} c\kappa^{2j}\gamma^{2(m-j)}$. □

13.2 Fehler der Finite-Elemente-Methode

Mit Hilfe des Interpolationsfehlers und des Lemmas von Céa 12.7 können wir nun den Fehler der Finite-Elemente-Methode zur Lösung des Problems

$$a(u, v) - l(v) = 0 \qquad \text{für alle } v \in H^1(\Omega)$$

abschätzen.

Satz 13.7 Sei $\Omega \subset \mathbb{R}^n$ ein Gebiet mit Lipschitzrand, $m > n/2$ und $\mathcal{T}_{h_l}$ eine formreguläre Folge von Triangulierungen. Es seien die Ansatzfunktionen $v_h \in V_{h_l}$ die stetigen, stückweisen Polynome vom Grad $m - 1$. Zudem sei a stetig und elliptisch und $u \in H^m(\Omega)$. Dann existiert eine Konstante $C > 0$, so dass für den Finite-Elemente-Fehler gilt

$$\|u - u_{h_l}\|_1 \leq C h_l^{m-1} |u|_m.$$

Beweis: Die Ungleichung folgt direkt aus dem Céa-Lemma 12.7 und Satz 13.6:

$$\|u - u_{h_l}\|_1 \leq c_1 \|u - Iu\|_1 \leq C h_l^{m-1} |u|_m.$$

□

Unter geeigneten Regularitätsannahmen an die Lösung u kann man die Ungleichung

$$\|u - u_h\|_{L_2(\Omega)} \leq Kh \|u - u_h\|_1$$

zeigen (siehe z.B. [2, Folgerung 7.7]). Damit ergibt sich unter den Annahmen von Satz 13.7 die Ungleichung

$$\|u - u_{h_l}\|_{L_2(\Omega)} \leq c h_l^m |u|_m.$$

Die Abschätzung wird also um den Faktor h_l besser, wenn wir von der H^1-Norm zur L_2-Norm übergehen.

13.3 Fehlerschätzer und Adaptivität

Adaptivität ist bei der Finite Elemente Methode das Gegenstück zur Schrittweitensteuerung bei den gewöhnlichen Differentialgleichungen. Wie dort benötigen wir auch hier Fehlerschätzer, um feststellen zu können, ob der Fehler in einem Teilgebiet zufriedenstellend klein ist oder nicht. Der wesentliche Unterschied zu den gewöhnlichen Differentialgeichungen besteht darin, dass wir das Gitter hier als ganzes betrachten müssen, da es nicht mit der Lösung Schritt für Schritt aufgebaut wird sondern vor der Durchführung der Rechnung vollständig vorhanden sein muss.

Der Grundalgorithmus dabei lautet wie folgt.

Algorithmus 13.8 (Grundalgorithmus zur Adaptiven Triangulierung)

Eingabe: Toleranz TOL, Finite Elemente Raum V_0, Lösung u_0 auf V_0, Fehlerschätzer ε_0 für die Gesamtlösung, Fehlerindikator η_i auf den Teilgebieten T_i

(0) setze $j := 1$
(1) solange $\varepsilon_{j-1} > TOL$
(2) verfeinere Teilgebiete mit großen $\eta_i \quad \rightsquigarrow \quad V_j$
(3) berechne die Lösung u_j auf V_j, also u_j mit $a(u_j, v_j) - l(v_j) = 0$ für alle $v_j \in V_j$
(4) schätze den Fehler ε_j und berechne die Fehlerindikatoren η_i
(5) Ende der Schleife □

Der Algorithmus könnte in verschiedener Hinsicht noch erweitert werden. Z.B. könnten auch Vergröberungen von Teilgebieten bei sehr kleinem Feherindikator durchgeführt werden.

In den folgenden Abschnitten erläutern wir die einzelnen Schritte und Komponenten des Algorithmus genauer.

13.4 Verfeinerungsmethoden

Hier muss man zunächst entscheiden, welche Teilgebiete verfeinert werden. Hierzu gibt es verschiedene Möglichkeiten. Entweder man bestimmt einen Schwellenwert $\bar{\eta}$ und verfeinert alle Teilgebiete mit $\eta_i > \bar{\eta}$. Der Wert $\bar{\eta}$ kann dabei entweder absolut (z.B. gleich oder abhängig von der der Toleranz TOL) oder relativ (z.b. $0.8 \cdot \max_i \eta_i$) gewählt werden.

Eine andere Methode ist, eine Prozentzahl d festzulegen und immer die $d\%$ Teilgebiete mit den größten η_i zu verfeinern. Es hängt typischerweise vom Problem ab, welche Methode am schnellsten und effizientesten zu einer Triangulierung führt, auf der die gewünschte Toleranz eingehalten wird.

Die Frage, wie ein zur Verfeinerung vorgesehenes Teilgebiete tatsächlich verfeinert wird, hängt stark von der Geometrie der Gebiete ab. Üblicherweise gibt es dabei zwei Dinge zu beachten:

- Die Folge der erzeugten Triangulierungen muss formregulär sein. Z.B. bei einem Dreicksgitter ist also zu vermeiden, dass bei der Verfeinerung immer spitzere Winkel entstehen.

- Die Triangulierungen müssen konform sein. Das bedeutet, dass ein Eckpunkt p_j eines Teilgebiets immer auch ein Eckpunkt jedes Teilgebiets ist, das p_j enthält. Nichtkonformität kann zwar in speziellen Anwendungen erlaubt sein, führt aber dazu, dass die Konstruktion der Basisfunktionen aus den Formfunktionen sehr kompliziert wird und wird daher in der Regel vermieden.

Für Dreiecks- und Tetraedergitter erfüllt die folgende Art der Verfeinerung beide Bedingungen:

- Jedes zu unterteilende Dreieck wird in vier ähnliche Dreiecke mit halber Kantenlänge zerlegt. Bei Tetraedern führt die Halbierung der Kanten zu vier ähnlichen Tetraedern und einem Oktaeder, das wiederum in vier Tetraeder zerlegt werden kann. Diese Art der Verfeinerung durch Halbierung aller Kanten wird *rote Verfeinerung* genannt.
- Jedes Dreieck, das gemeinsame Kanten mit mehreren rot verfeinerten Dreieck besitzt, wird selbst wieder rot verfeinert. Analog wird bei Tetraedern verfahren, allerdings sind hier die Nachbarn mit gemeinsamen Seiten oder gemeinsamen Kanten ausschlaggebend.
- Jedes Dreieck, das nur eine gemeinsame Kante mit einem verfeinerten Dreieck besitzt, wird halbiert. Tetraeder werden halbiert oder geviertelt, je nach dem, ob eine gemeinsame Kante oder eine gemeinsame Seite mit einem verfeinerten Dreieck vorliegt. Diese Art der Verfeinerung nennt man *grüne Verfeinerung*. Eine grüne Verfeinerung wird rückgängig gemacht, falls eines seines Teilgebiete rot verfeinert werden soll.

Zusätzlich kann man verschiedene Regularitätsbedingungen einführen, z.B. dass nur eine gewisse Anzahl grüner Verfeinerungen eines Teilgebiets erlaubt ist.

13.5 Residuenbasierte Fehlerschätzer

Ein Fehlerschätzer ε sollte eine Abschätzung für den tatsächlichen Fehler $\|u_h - u\|_V$ liefern. Dabei heißt der Fehlerschätzer *zuverlässig*, falls ein $C_z > 0$ existiert mit

$$\|u_h - u\|_V \leq C_z \varepsilon$$

und er heißt *effizient*, wenn ein $C_e > 0$ existiert mit

$$C_e \varepsilon \leq \|u_h - u\|_V.$$

Er heißt asymptotisch exakt, falls $\varepsilon / \|u_h - u\|_V \to 1$ für $h \to 0$. Oftmals bekommt man die obigen Ungleichungen nicht für beliebige $h > 0$ sondern nur für hinreichend kleine; das haben wir schon bei den konzeptionell identischen Fehlerschätzern für gewöhnliche Differentialgleichungen aus Definition 7.1 gesehen, vgl. Satz 7.2.

Eine Möglichkeit, einen Fehlerschätzer zu definieren, ist die Verwendung des Residuums. Um das Vorgehen zu erläutern, betrachten wir das Modellproblem $a(u, v) - l(v) = 0$ für alle $v \in V$ mit

$$a(u, v) = \int_\Omega \nabla u \cdot \nabla v dx \quad \text{und} \quad l(v) = \int_\Omega f v dx$$

und $V = H_0^1(\Omega)$. Für eine Finite-Elemente-Lösung $u_h \in V_h$ des Problems und eine Testfunktion $v \in V$ definieren wir dann das Residuum als

$$r(v) := a(u_h, v) - l(v).$$

Das Residuum gibt also an, wie stark die gewünschte Identität $a(u, v) - l(v) = 0$ verletzt ist. Beachte, dass für alle $v_h \in V_h$ gerade $r(v_h) = 0$ gilt, denn u_h ist ja gerade durch $a(u_h, v_h) - l(v_h) = 0$ für alle $v_h \in V_h$ bestimmt.

Wir zeigen nun, dass $\varepsilon = \|r\|_{V^*}$ ein zuverlässiger und effizienter Fehlerschätzer ist. Definieren wir

$$\|r\|_{V^*} := \sup_{\substack{v \in V \\ v \neq 0}} \frac{|r(v)|}{\|v\|_V},$$

so folgt aus der Elliptizität von a

$$\begin{aligned} \alpha \|u_h - u\|_V^2 &\leq a(u_h - u, u_h - u) = a(u, u_h - u) - a(u_h, u_h - u) \\ &= a(u_h, u_h - u) - l(u_h - u) = r(u_h - u) \leq \|r\|_{V^*} \|u_h - u\|_V \end{aligned}$$

und damit

$$\|u_h - u\|_V \leq \frac{1}{\alpha} \|r\|_{V^*}.$$

Dies zeigt die Zuverlässigkeit.

Zum Beweis der Effizient definieren wir die a-Norm $\|v\|_a := \sqrt{a(v, v)}$. Dann folgt aus der Stetigkeit von a die Ungleichung

$$\|u_h - u\|_a \leq \sqrt{C} \|u_h - u\|_V.$$

Zudem gilt wegen der Bilinearität und Symmetrie von a, die Cauchy-Schwarz-Ungleichung

$$|a(u_h - u, v)| \leq \|u_h - u\|_a \|v\|_a,$$

wobei Gleichheit gerade für $v = u_h - u$ gilt. Also folgt

$$\|u_h - u\|_a = \sup_{\substack{v \in V \\ v \neq 0}} \frac{|a(u_h - u, v)|}{\|v\|_a}.$$

Für den Zähler dieses Bruchs gilt wegen $a(u, v) - l(v) = 0$

$$|a(u_h - u, v)| = |a(u_h, v) - a(u, v)| = |a(u_h, v) - l(v)| = |r(v)|.$$

Wir erhalten also die Gleichung

$$\|u_h - u\|_a = \sup_{\substack{v \in V \\ v \neq 0}} \frac{|r(v)|}{\|v\|_a} \geq \sup_{\substack{v \in V \\ v \neq 0}} \frac{1}{\sqrt{C}} \frac{|r(v)|}{\|v\|_V} = \frac{1}{\sqrt{C}} \|r\|_{V^*},$$

wobei wir bei der Ungleichung die aus der Stetigkeit folgende Ungleichung $\|v\|_a \leq \sqrt{C} \|v\|_V$ im Nenner verwendet haben. Insgesamt gilt also die Ungleichung

$$\|u_h - u\|_V \geq \frac{1}{\sqrt{C}} \|u_h - u\|_a \geq \frac{1}{C} \|r\|_{V^*},$$

also die Effizienz. Folglich ist $\varepsilon = \|r\|_{V^*}$ ein zuverlässiger und effizienter Fehlerschätzer.

Es bleibt zu klären, wie $\|r\|_{V^*}$ numerisch (näherungsweise) berechnet werden kann und wie die zugehörigen Fehlerindikatoren η_i berechnet werden können.

Mit mehrdimensionaler partieller Integration über die Teilgebiete erhalten wir

$$\begin{aligned} r(v) &= \int_\Omega \nabla u_h \cdot \nabla v - fv dx \\ &= \sum_{T\in\mathcal{T}} \int_T \nabla u_h \cdot \nabla v - fv dx \\ &= \sum_{T\in\mathcal{T}} \int_T (-\Delta u_h - f) v dx + \sum_{T\in\mathcal{T}} \int_{\partial T} \nabla(u_h|_T)\cdot \nu v ds. \end{aligned}$$

Beachte, dass ∇u_h am Rand der Teilgebiete T_i i.d.R. unstetig ist. Die Ableitungen $\nabla(u_h|_{T_i})(x)$ und $\nabla(u_h|_{T_j})(x)$ stimmen also für $x \in \partial T_i \cap \partial T_j$ i.A. nicht überein.

Der Rand ∂T eines Teilgebiets besteht nun aus verschiedenen $n-1$-dimensionalen Objekten, den *Facetten* (beim Dreieck sind das gerade die Kanten, beim Tetraeder die Seitenflächen). Jede Facette F im Inneren von Ω kommt dabei in genau zwei (benachbarten) Teilgebieten vor; wir bezeichnen diese beiden Teilgebiete mit $T_{F,1}$ und $T_{F,2}$. Auf jeder Randfacette $F \subset \Gamma = \partial\Omega$ ist $v|_F \equiv 0$, da $v|_\Gamma \equiv 0$ wegen $v \in H_0^1(\Omega)$. Bezeichnen wir die Menge der Facetten in $\mathcal{T}$ mit $\mathcal{F}$, so folgt

$$r(v) = \sum_{T\in\mathcal{T}} \int_T \underbrace{(-\Delta u_h - f)}_{=:R(u_h)} v dx + \sum_{\substack{F\in\mathcal{F}\\ F\not\subset\Gamma}} \int_F \underbrace{(\nabla(u_h|_{T_{F,1}}) - \nabla(u_h|_{T_{F,2}}))\cdot\nu}_{=:J(u_h)} v ds.$$

Setzen wir J auf dem Rand Γ gleich 0, so können wir dies kürzer schreiben als

$$r(v) = \sum_{T\in\mathcal{T}} \int_T R(u_h) v dx + \sum_{F\in\mathcal{F}} \int_F J(u_h) v ds.$$

Wegen $r(v_h) = 0$ erhalten wir für $v_h \in V_h$

$$0 = r(v_h) = \sum_{T\in\mathcal{T}} \int_T R(u_h) v_h dx + \sum_{F\in\mathcal{F}} \int_F J(u_h) v_h ds.$$

Wegen der Linearität von r folgt daraus $r(v - v_h) = r(v) - r(v_h) = r(v)$ und damit für $S = \bigcup_{F\in\mathcal{F}} F$

$$r(v) = r(v - v_h) = \sum_{T\in\mathcal{T}} \int_T R(u_h)(v - v_h) dx + \sum_{F\in\mathcal{F}} \int_F J(u_h)(v - v_h) ds$$

für alle $v_h \in V_h$.

Mit Hilfe der beiden Integrale auf der rechten Seite können wir nun den Fehlerschätzer $\varepsilon = \|r\|_{V^*}$ durch numerisch berechenbare Größen nach oben und unten abschätzen. Für die obere Schranke verwenden wir, dass wir alle $v \in V$ durch $v_h \in V_h$ so approximieren können, dass die Ungleichungen

$$\sum_{T\in\mathcal{T}} h_T^{-2} \int_T |v - v_h|^2 dx \le c^2 |v|^2_{H^1(\Omega)} \text{ und } \sum_{F\in\mathcal{F}} h_F^{-1} \int_F |v - v_h|^2 ds \le c^2 |v|^2_{H^1(\Omega)}$$

gelten (vgl. [4, Formel (6.11)]), wobei die (i.A. unbekannte) Konstante nur von der Geometrie der Triangulierung abhängt. Damit ergibt sich

$$|r(v)| \leq C(\|R(u_h)\|_{L_2(\Omega)} h_{\mathcal{T}} + \|J(u_h)\|_{L_2(S)} \sqrt{h_{\mathcal{F}}}) |v|_{H^1(\Omega)}$$

und folglich

$$\|r\|_{V^*} \leq C(\|R(u_h)\|_{L_2(\Omega)} h_{\mathcal{T}} + \|J(u_h)\|_{L_2(S)} \sqrt{h_{\mathcal{F}}}),$$

wobei $h_{\mathcal{F}}$ den maximalen Durchmesser der Facetten in $\mathcal{F}$ bezeichnet. Die rechte Seite kann nun für lineare Finite Elemente leicht numerisch ausgewertet werden: Da Δu_h auf den Teilgebieten T konstant ist, muss zur Auswertung von $|R(u_h)\|_{L_2(T)}$ auf jedem T nur über f integriert werden, was üblicherweise durch eine Quadraturformel geschieht. Zur Berechnung von $\|J(u_h)\|_{L_2(F)}$ muss auf jeder Facette F nur ein Skalarprodukt ausgewertet werden. Für finite Elemente höherer Ordnung lässt sich dies verallgemeinern; hier muss dann auch über Δu_h und ∇u_h integriert werden.

Etwas schwieriger ist es zu beweisen, dass auch ein $c > 0$ existiert, für das die umgekehrte Ungleichung gilt, also

$$\|r\|_{V^*} \geq c(\|R(u_h)\|_{L_2(\Omega)} h_{\mathcal{T}} + \|J(u_h)\|_{L_2(S)} \sqrt{h_{\mathcal{F}}}).$$

Hierzu konstruiert man einen Unterraum geeigneter Funktionen $\widetilde{V} \subset V$ — die sogenannten "Bubbles" —, für die ein $c > 0$ mit

$$\sup_{\substack{v \in \widetilde{V} \\ v \neq 0}} \frac{r(v)}{\|v\|_V} \geq c(\|R(u_h)\|_{L_2(\Omega)} h_{\mathcal{T}} + \|J(u_h)\|_{L_2(S)} \sqrt{h_{\mathcal{F}}}) + \text{ Terme höherer Ordnung}$$

gefunden werden kann. Aus der offensichtlichen Ungleichung

$$\|r\|_{V^*} = \sup_{\substack{v \in V \\ v \neq 0}} \frac{|r(v)|}{\|v\|_V} \geq \sup_{\substack{v \in \widetilde{V} \\ v \neq 0}} \frac{|r(v)|}{\|v\|_V}$$

folgt dann die gesuchte Beziehung für alle hinreichend kleinen $h_{\mathcal{T}}$ und $h_{\mathcal{F}}$. Der "theoretische" Fehlerschätzer $\varepsilon = \|r\|_{V^*}$ kann also durch den numerisch berechenbaren Fehlerschätzer

$$\hat{\varepsilon} = \|R(u_h)\|_{L_2(\Omega)} h_{\mathcal{T}} + \|J(u_h)\|_{L_2(S)} \sqrt{h_{\mathcal{F}}}$$

abgeschätzt werden, der damit selbst ein effizienter und zuverlässiger Fehlerschätzer ist.

Durch Einschränkung auf die Teilgebiete T_i lassen sich leicht lokale Fehlerindikatoren

$$\eta_i := \sqrt{\int_{T_i} R(u_h)^2 dx} h_{T_i} + \sum_{F \subset T_i} \sqrt{\int_F J(u_h)^2 ds} \sqrt{h_F}$$

definieren, die darüber Auskunft geben, wie stark das Teilgebiet T_i zum Gesamtfehler beiträgt.

13.6 Hierarchische Fehlerschätzer

Die residuenbasierten Fehlerschätzer beruhen darauf, wie gut die numerische Lösung die exakte Gleichung erfüllen. Ein alternatives Konzept bilden die hierarchischen Fehlerschätzer, die — ähnlich wie bei den gewöhnlichen Differentialgleichungen — darauf beruhen, wie sehr sich Verfahren unterschiedlicher Ordnung voneinander unterscheiden. Wir erläutern dies am Beispiel linearer und quadratischer finiter Elemente. Die zugehörigen Räume bezeichnen wir mit V_h^1 und V_h^2. Berechnen wir mit beiden Räumen die numerischen Lösungen u_h^1 und u_h^2, so gilt

$$\begin{aligned} a(u_h^1, v_h) - l(v_h) &= 0 \text{ für alle } v_h \in V_h^1 \\ a(u_h^2, v_h) - l(v_h) &= 0 \text{ für alle } v_h \in V_h^2. \end{aligned}$$

Ganz analog zu den gewöhnlichen Differentialgleichungen kann man im Fall, dass u_h^2 tatsächlich genauer ist als u_h^1, abschätzen

$$\|u_h^1 - u\|_V = \|u_h^1 - u_h^2 + u_h^2 - u\|_V \leq \|u_h^1 - u_h^2\|_V + \|u_h^2 - u\|_V \approx \|u_h^1 - u_h^2\|_V.$$

Im Gegensatz zu den gewöhnlichen Differentialgleichungen gibt es hier aber i.d.R. kein Analogon zu den eingebetteten Verfahren, mit dem man u_h^1 und u_h^2 mit wenige Mehraufwand gemeinsam berechnen könnte.

Daher berechnet man u_h^2 hier in der Regel nicht exakt sondern nur näherungsweise. In vielen Fällen gilt die Beziehung $V_h^2 = V_h^1 \oplus V_h^B$, woraus

$$u_h^2 = u_L + u_B$$

mit $u_L \in V_h^1$ und $u_B \in V_h^B$ folgt. Das lineare Gleichungssyystem zur Berechnung von u_h^2 ist dann von der Form

$$\begin{pmatrix} A_{LL} & A_{LB} \\ A_{BL} & A_{BB} \end{pmatrix} \begin{pmatrix} u_L \\ u_B \end{pmatrix} = \begin{pmatrix} b_L \\ b_B \end{pmatrix}.$$

A_{LL} und b_L bilden dabei genau das Gleichungssystem auf V_h^1, also das zur Bestimmung von u_h^1. Daher gilt $A_{LL}u_h^1 = b_L$ und damit

$$\begin{pmatrix} A_{LL} & A_{LB} \\ A_{BL} & A_{BB} \end{pmatrix} \begin{pmatrix} u_L - u_h^1 \\ u_B \end{pmatrix} = \begin{pmatrix} b_L - A_{LL}u_h^1 \\ b_B - A_{BL}u_h^1 \end{pmatrix} = \begin{pmatrix} 0 \\ b_B - A_{BL}u_h^1 \end{pmatrix}.$$

Beachte, dass u_B in der Regel nicht mit u_h^1 übereinstimmt, da bei der Berechnung von u_B die Kopplungsterme A_{BL} und A_{LB} berücksichtigt werden müssen, die bei der Berechnung von u_h^2 nicht vorhanden sind. Es ist aber oft der Fall, dass die beiden Funktionen näherungsweise übereinstimmen, weswegen man die Differenz vernachlässigt und $u_L - u_h^1 = 0$ setzt. Damit erhält man $u_h^1 - u_h^2 = u_B$ und

$$A_{BB}u_B = b_B - A_{BL}u_h^1. \tag{13.1}$$

Eine näherungsweise Lösung dieser Gleichung ist gegeben durch

$$\hat{u}_B = D_{BB}^{-1}(b_B - A_{BL}u_h^1),$$

wobei D_{BB} den Diagonalanteil von A_{BB} bezeichnet. Diese Größe $\hat{u}_B$ wird nun als Fehlerschätzer verwendet. Man kann $\hat{u}_B$ als erste Iteration des Jacobi-Verfahrens zur Lösung von (13.1) betrachten. Dies erklärt auch, warum $\hat{u}_B$ (unter geeigneten Voraussetzungen) tatsächlich eine Näherung von u_B darstellt.

Kapitel 14

Vorkonditionierung und hierarchische Gitter

Wir haben am Ende von Abschnitt 12.5 gesehen, dass die Matrix A aus dem linearen Gleichungssystem (12.11) schlecht konditioniert ist, wodurch iterative Löser wie z.B. das CG-Verfahren langsam konvergieren. Wir erinnern an die Vertiefung der Numerik, wo wir gesehen haben, dass dies durch die Definition eines Vorkonditionierers, also einer Matrix M, für die $M^{-1}A$ besser konditioniert ist, behoben werden kann. Dabei muss aber sichergestellt sein, dass das lineare Gleichungsystem $My = d$ mit wenig Aufwand gelöst werden kann, da dies in jeder Iteration des CG-Verfahrens auftritt. Entscheidend für die Konvergenz des CG-Verfahrens ist dabei die Kondition

$$\kappa_{A,M} = \frac{\Gamma}{\gamma},$$

wobei γ und Γ der kleinste bzw. der größte Eigenwert von $M^{-1}A$ sind. Für die Eigenwerte λ von $M^{-1}A$ und die zugehörigen Eigenvektoren v gilt

$$M^{-1}Av = \lambda v \;\Rightarrow\; Av = \lambda Mv \;\Rightarrow\; v^TAv = \lambda v^TMv \;\Rightarrow\; \lambda = \frac{v^TAv}{v^TMv}.$$

Mit etwas Überlegung sieht man, dass daraus

$$\gamma = \inf_{v\neq 0} \frac{v^TAv}{v^TMv} \quad \text{und} \quad \Gamma = \sup_{v\neq 0} \frac{v^TAv}{v^TMv} \tag{14.1}$$

folgt.

Man kann nun beweisen, dass $\kappa_{A,\mathrm{Id}} = ch^{-2}$ und $\kappa_{A,\mathrm{diag}(A)} = ch^{-2}$ gilt. Die Iterationszahl nimmt also ohne Präkonditionierer und mit Jacobi-Präkonditionierer quadratisch mit der Feinheit der Teilgebiete zu. Diesen Effekt würde man gerne vermeiden.

Wenn a stetig und elliptisch auf $V = H_0^1(\Omega)$ ist, existieren Konstanten $c_1, c_2 > 0$, so dass für das H^1-Skalarprodukt die Ungleichungen

$$c_1\langle v, v\rangle_{H^1} \leq a(v, v) \leq c_2\langle v, v\rangle_{H^1}$$

für alle $v \in V$ gelten. Wir schreiben in diesem Fall $\langle v, v\rangle_{H^1} \approx a(v, v)$.

Schreibt man für $v \in V_h$ die Koeffizienten der Basisfunktionen als $\underline{v} \in \mathbb{R}^N$, so folgt

$$\underline{v}^T A \underline{v} = a(v, v) \approx \langle v, v\rangle_{H^1},$$

wobei die Konstanten in "$\approx$" unabhängig von $h = h_{\mathcal{T}}$ sind. Wenn wir nun Vorkonditionierer M finden können, so dass für alle $v \in V_h$ die Beziehung

$$\underline{v}^T M \underline{v} \approx \langle v, v\rangle_{H^1} \tag{14.2}$$

ebenfalls mit von h unabhängigen Konstanten gilt, so folgt aus (14.1), dass γ und Γ und damit $\kappa_{A,M}$ unabhängig von h sind. Zusätzlich sollte die Lösung von $My = d$ den Rechenaufwand $O(N)$ mit $N = \dim V_h$ nicht überschreiten, da dieses Gleichungssystem in jedem CG-Schritt gelöst werden muss.

14.1 Hierarchische Zerlegung

Im Gegensatz zu den hierarchischen Fehlerschätzern aus dem vorhergehenden Kapitel ist die "Hierarchie" hier nicht durch Finite-Element-Räume verschiedener Ordnung sondern durch Räume mit unterschiedlicher Feinheit gegeben.

Es seien dazu V_k geschachtelte (also $V_{k+1} \subset V_k$) Finite-Element-Räume mit unterschiedlichen Gitterfeinheiten $h_k > 0$, so dass $h_{k+1} = h_k/2$ gilt. Wir betrachten für $k \geq 1$ den L_2-Projektionsoperator

$$Q_k : L_2 \to V_k \quad u \mapsto \operatorname*{argmin}_{u_k \in V_k} \frac{1}{2}\|u - u_k\|_{L_2}^2.$$

Für $g(u_k) = \frac{1}{2}\|u - u_k\|_{L_2}^2$ gilt $Dg(u_k)v = \langle u - u_k, v\rangle_{L_2}$. Da g von $Q_k u$ minimiert wird, folgt $\langle u - Q_k u, v_k\rangle_{L_2} = 0$ für alle $v_k \in V_k$. Zudem gilt offensichtlich $Q_k v_k = v_k$ für alle $v_k \in V_k$. Zerlegen wir ein beliebiges $v \in V$ nun in $v = v_k + \tilde{v}$ mit $v_k = Q_k v \in V_k$ und $\tilde{v} \in V_k^{\perp}$, so folgt

$$\langle Q_k u, v\rangle_{L_2} = \langle Q_k u, v_k + \tilde{v}\rangle_{L_2} = \langle Q_k u, v_k\rangle_{L_2} = \langle u, v_k\rangle_{L_2} = \langle u, Q_k v\rangle_{L_2}. \tag{14.3}$$

Der Operator Q_k ist also selbstadjungiert bzgl. des L_2-Skalarprodukts.

Auf Grund von Satz 13.6 gilt zudem $Q_k u \to u$ für $k \to \infty$. Setzen wir also $Q_0 u := 0$ und definieren für alle $j = 1, 2, 3, \ldots$ die Differenzen $\Delta Q_j := Q_j - Q_{j-1}$, so folgt

$$\sum_{j=1}^{\infty} \Delta Q_j u = \lim_{k\to\infty} \sum_{j=1}^{k} \Delta Q_j u = \lim_{k\to\infty} Q_k u = u. \tag{14.4}$$

Da die Gitter immer feiner werden, zerlegt man die Lösung u dadurch in Anteile $\Delta Q_j u$ mit unterschiedlichen "Frequenzen". Dass dabei wirklich eine Zerlegung stattfindet, zeigt das folgende Lemma.

Lemma 14.1 Für alle $k \neq j$ gilt $\Delta Q_k \Delta Q_j = 0$ und es gilt $\Delta Q_j \Delta Q_j = \Delta Q_j$.

Beweis: Wegen der Schachtelung der Finite-Element-Räume gilt $Q_k Q_j = Q_{\min\{k,j\}}$. Daraus folgt:

$$\begin{array}{lllllll} k<j: & \Delta Q_k \Delta Q_j & = & (Q_k - Q_{k-1})(Q_j - Q_{j-1}) & = & Q_k - Q_{k-1} - Q_k + Q_{k-1} & = & 0 \\ k>j: & \Delta Q_k \Delta Q_j & = & (Q_k - Q_{k-1})(Q_j - Q_{j-1}) & = & Q_j - Q_{j-1} - Q_j + Q_{j-1} & = & 0 \\ k=j: & \Delta Q_k \Delta Q_k & = & (Q_k - Q_{k-1})(Q_k - Q_{k-1}) & = & Q_k - Q_{k-1} - Q_{k-1} + Q_{k-1} & = & \Delta Q_k \end{array}$$

□

Wegen (14.3) ist auch ΔQ_k selbstadjungiert. Zusammen mit Lemma 14.1 folgt, dass ΔQ_j eine L_2-orthogonale Zerlegung definiert, denn für $j \neq k$ gilt

$$\langle \Delta Q_j u, \Delta Q_k u \rangle_{L_2} = \langle \Delta Q_k \Delta Q_j u, u \rangle_{L_2} = 0.$$

Aus (14.4) folgt sofort

$$\langle u, u \rangle_{L_2} = \sum_{j=1}^{\infty} \langle \Delta Q_j u, \Delta Q_j u \rangle_{L_2}.$$

Für $u \in H^1$ können wir mehr zeigen. Aus Satz 13.6 folgt

$$\|u - Q_k u\|_{L_2} \leq C h_k \|u\|_{H^1}.$$

Damit gilt

$$\|\Delta Q_j u\|_{L_2}^2 \leq C h_j^2 \|u\|_{H^1}^2.$$

Man kann aber noch mehr zeigen, nämlich (für einen Beweis siehe Oswald [11])

$$\langle u, u \rangle_{H^1} \approx \sum_{j=1}^{\infty} \frac{1}{h_j^2} \langle \Delta Q_j u, \Delta Q_j u \rangle_{L_2}.$$

Insbesondere folgt für H^1-Funktionen u die Konvergenz $\|\Delta Q_j u\|_{L_2}^2 / h_j^2 \to 0$ für $j \to \infty$. Zudem folgt für alle $u \in \operatorname{Im} \Delta Q_j$ wegen $\Delta Q_k u = 0$ für $k \neq j$ die Relation

$$\|u\|_{H^1}^2 \approx \frac{1}{h_j^2} \|u\|_{L_2}^2.$$

14.2 Der BPX-Vorkonditionierer

Definieren wir nun für $J \in \mathbb{N}$, $u \in V_J$ eine lineare Abbildung M_0 mittels

$$M_0 u := \sum_{j=1}^{J} \frac{1}{h_j^2} \Delta Q_j u$$

so gilt wegen

$$\langle \Delta Q_j u, \Delta Q_j u \rangle_{L_2} = \langle \Delta Q_j \Delta Q_j u, u \rangle_{L_2} = \langle \Delta Q_j u, u \rangle_{L_2}$$

die Beziehung

$$\langle M_0 u, u \rangle_{L_2} = \sum_{j=1}^{J} \frac{1}{h_j^2} \langle \Delta Q_j u, u \rangle_{L_2} = \sum_{j=1}^{J} \frac{1}{h_j^2} \langle \Delta Q_j u, \Delta Q_j u \rangle_{L_2} \approx \langle u, u \rangle_{H^1}.$$

Die lineare Abbildung M_0 erfüllt also genau die Eigenschaften, die der Vorkonditionierer gemäß (14.2) haben sollte. Aus Effizienzgründen werden wir M_0 im Folgenden durch eine etwas andere Abbildung $M \approx M_0$ annähern. Der folgende Satz bildet die Grundlage dafür.

Satz 14.2 Für die durch

$$M_1^{-1}v := \sum_{j=1}^{J} h_j^2 Q_j v$$

definierte lineare Abbildung M_1 gilt

$$\langle M_0^{-1}v, v\rangle \approx \langle M_1^{-1}v, v\rangle.$$

Beweis: Es gilt

$$\Delta Q_k M_0 v = \Delta Q_k \sum_{j=1}^{J} \frac{1}{h_j^2}\Delta Q_j u = \frac{1}{h_j^2}\Delta Q_k v.$$

Daraus folgt für alle $v \in V_J$ wegen $Q_J|_{V_J} = \mathrm{Id}_{V_J}$

$$\sum_{k=1}^{J} h_k^2 \Delta Q_k M_0 v = \sum_{k=1}^{J} h_k^2 \Delta Q_k \left(\sum_{j=1}^{J} \frac{1}{h_j^2}\Delta Q_j v \right) = \sum_{k=1}^{J} \Delta Q_k v = Q_J v = v,$$

also

$$M_0^{-1}v = \sum_{k=1}^{J} h_k^2 \Delta Q_k v.$$

Für diesen Ausdruck gilt für alle $v \in V_J$

$$\sum_{j=1}^{J} h_j^2 \Delta Q_j v = \sum_{j=1}^{J-1} (h_j^2 - h_{j+1}^2) Q_j v + h_J^2 v = \sum_{j=1}^{J-1} \frac{3}{4} h_j^2 Q_j v + h_J^2 Q_J v,$$

woraus die Behauptung durch Vergleich der Skalarprodukte folgt. □

Um $M_1^{-1}v$ auszuwerten, müssen wir $Q_j v$ für $j = 1, \ldots, J$ berechnen, d.h. wir müssen $u^* := Q_j v = \mathrm{argmin}_{u_k \in V_k} \frac{1}{2}\|v - u_k\|_{L_2}^2$ berechnen. Ableiten und Nullsetzen der Ableitung ergibt (wie oben bereits schon einmal verwendet) die Bedingung $\langle v - u^*, v_j\rangle_{L_2} = 0$ für alle $v_j \in V_j$, die wir auch als

$$\langle v, v_j\rangle_{L_2} = \langle u^*, v_j\rangle_{L_2}$$

schreiben können. Diese Gleichung ist genau dann für alle $v_j \in V_j$ erfüllt, wenn sie für eine Basis $(\psi_k^j)_{k=1,\ldots,d_j}$ von V_j erfüllt ist, wenn also

$$\langle v, \psi_k\rangle_{L_2} = \langle u^*, \psi_k^j\rangle_{L_2}$$

gilt. Schreiben wir $u^* = \sum_{k=1}^{d_j} \underline{u}_k^* \psi_k^j$, so ist dies äquivalent zu dem Gleichungssystem

$$B^j \underline{u}^* = b^j \qquad \text{mit } B_{ik}^j = \langle \psi_i^j, \psi_k^j\rangle_{L_2} \text{ und } b_i^j = \langle v, \psi_k\rangle_{L_2}. \tag{14.5}$$

Die Matrix B^j wird *Massematrix* genannt und man kann beweisen, dass ihre Kondition unabhängig von j ist, siehe Deuflhard/Weiser [4], Aufgabe 4.14.

Da das Lösen des Gleichungssystems (14.5) aufwändig ist, ersetzt man B^j durch die Diagonalmatrix $\mathrm{diag}(B^j)$, die nur die Diagonaleinträge $B^j_{ii} = \langle \psi_i, \psi_i \rangle_{L_2}$ von B enthält. Damit wird $\tilde{u} \approx M_1^{-1} u$ berechnet als

$$\tilde{u} = \sum_{j=1}^{J} h_j^2 \sum_{i=1}^{d_j} \frac{\langle v, \psi_k \rangle_{L_2}}{B^j_{ii}} \psi_i^j.$$

Aus der Praxis weiß man, dass der Vorkonditionierer immer noch gut funktioniert, wenn wir $M_1^{-1} u$ durch $\tilde{u}$ ersetzen. Schließlich kann man auch die Berechnung von B^j_{ii} vermeiden, wenn man die Beziehung $h_j^2 a(\psi_i^j, \psi_i^j) \approx \langle \psi_i^j, \psi_i^j \rangle_{L_2} = B^j_{ii}$ ausnutzt. Auf Grund dieser Beziehung können wir B^j_{ii} durch $h_j^2 a(\psi_i^j, \psi_i^j) = h_j^2 A^j_{ii}$ ersetzen, wobei A^j die Steifigkeitsmatrix aus (12.11) ist. Dies führt letzendlich auf den Vorkonditionierer

$$M^{-1} v = \sum_{j=1}^{J} \sum_{i=1}^{d_j} \frac{\langle v, \psi_k \rangle_{L_2}}{A^j_{ii}} \psi_i^j = \sum_{j=1}^{J} \sum_{i=1}^{d_j} \frac{\langle v, \psi_i^j \rangle_{L_2}}{a(\psi_i^j, \psi_i^j)} \psi_i^j.$$

Dies ist der sogenannte *BPX*-Vorkonditionierer (nach Bramble, Pacsiak und Xu).

Auf dem gröbsten Raum V_1 ist A^1 eine relativ niedrigdimensionale Matrix, so dass man hier mit wenig Aufwand das volle Gleichungssystem (mit der Steifigkeitsmatrix A^1 an Stelle der Massenmatrix B^1) lösen kann, statt nur das Gleichungssystem mit $\mathrm{diag}(A^1)$ zu lösen. Man löst also $A^1 \underline{u}^{*,1} = b^1$ mit $b_i^1 = \langle v, \psi_i^1 \rangle_{L_2}$ und setzt

$$M^{-1} v = u^{*,1} + \sum_{j=2}^{J} \sum_{i=1}^{d_j} \frac{\langle v, \psi_i^j \rangle_{L_2}}{a(\psi_i^j, \psi_i^j)} \psi_i^j$$

mit $u^{*,1} = \sum_{k=1}^{d_1} \underline{u}_k^{*,1} \psi_k^1$ bzw.

$$M^{-1} \underline{v} = \underline{u}^{*,1} + \sum_{j=2}^{J} \sum_{i=1}^{d_j} \frac{\langle v, \psi_i^j \rangle_{L_2}}{a(\psi_i^j, \psi_i^j)}.$$

Kapitel 15

Finite Elemente für parabolische Gleichungen

In diesem Kapitel werden wir überblicksmäßig verschiedene Ansätze erläutern, um die Finite-Elemente-Methode auf parabolische Gleichungen anzuwenden. Schreiben wir die bisher betrachteteten elliptischen Gleichungen in der abstrakten Form $A(u) = 0$, so betrachten wir nun Gleichungen der Form

$$\dot{u} + A(u) = 0,$$

wobei die unbekannten Funktionen jetzt von $t \in \mathbb{R}$ und $x \in \mathbb{R}^n$ abhängen und $\dot{u}$ die Ableitung nach t bezeichnet. Alternativ zu $\dot{u}$ findet man auch oft die Bezeichungen u_t und u'. Mit ∇u und Δu bezeichnen wir weiterhin Ableitungen nach x.

Als konkretes Beispiel verwenden wir im Folgenden stets die zeitabhängige Wärmeleitungsgleichung

$$\dot{u}(t,x) - \Delta u(t,x) = f(t,x) \tag{15.1}$$

mit $x \in \Omega$, $\Omega \subset \mathbb{R}^n$ eine offene und beschränkte Teilmenge und $(x,t) \in Q = (0,\infty) \times \Omega$. Wir verwenden die Anfangsbedingung $u(0,x) = u_0(x)$ für alle $x \in \Omega$ und Dirichlet-0-Randbedingungen $u(t,x) = 0$ für alle $(t,x) \in (0,\infty) \times \partial\Omega$.

Eine klassische Lösung dieser Gleichung verlangt, dass u in x mindestens C^2 und in t mindestens C^1 ist. Wie bei den elliptischen Gleichungen gibt es aber auch hier die Möglichkeit, zu schwachen Lösungen überzugehen. Zusätzlich zu den bereits bekannten Räumen L_2, H^1 und H_0^1 definieren wir dazu die Räume

$$\begin{aligned}
L_2(0,T;L_2(\Omega)) &:= L_2(Q) \\
L_2(0,T;H^1(\Omega)) &:= \{u \in L_2(0,T;L_2(\Omega)) \,|\, \nabla u \in L_2(0,T;L_2(\Omega,\mathbb{R}^n))\} \\
L_2(0,T;H_0^1(\Omega)) &:= \{u \in L_2(0,T;H^1(\Omega)) \,|\, u(t,\cdot) \in H_0^1(\Omega) \text{ für fast alle } t \geq 0\} \\
L_2(0,T;H^{-1}(\Omega)) &:= L_2(0,T;H_0^1(\Omega))' \\
H^{-1}(\Omega) &:= H_0^1(\Omega)'
\end{aligned}$$

Dabei bezeichnet X' den Dualraum von X, also den Raum der stetigen linearen Abbindungen von X nach $\mathbb{R}$. Analog zur schwachen räumlichen Ableitung nennt man $v \in$

$L_2(0,T;H^{-1}(\Omega))$ die schwache Zeitableitung von $u \in L_2(0,T;L_2(\Omega))$, falls sie die Gleichung

$$\int_0^T \dot{\varphi}(t)u(t)dt = -\int_0^T \varphi(t)v(t)dt$$

für alle Testfunktionen $\varphi \in C_0^\infty((0,T);\mathbb{R})$ erfüllt. Wir schreiben dann $\partial_t u = v$.

Definition 15.1 Eine Funktion $u : Q \to \mathbb{R}$ heißt *schwache Lösung der Wärmeleitungsgleichung*, falls

- $u \in L_2(0,T;H_0^1(\Omega))$ und $\partial_t u \in L_2(0,T;H^{-1}(\Omega))$
- $\langle \partial_t u(t,\cdot), v\rangle_{L_2} + \langle \nabla u, \nabla v\rangle_{L_2} = \langle f(t,\cdot), v\rangle_{L_2}$ für alle $v \in H_0^1(\Omega)$ und fast alle $t \in (0,T)$
- $u(0,\cdot) = u_0$.

Dabei muss die rechte Seite $f \in L_2(0,T;H^{-1}(\Omega))$ erfüllen, damit $\langle f(t), v\rangle_{L_2} < \infty$ garantiert ist. □

15.1 Die Linienmethode

Die Idee der Linienmethode (manchmal auch vertikale Linienmethode genannt) besteht darin, den "räumlichen Teil" der Gleichung wie eine elliptische Gleichung zu diskretisieren. Wir erhalten dann eine gewöhnliche Differentialgeichung in dem endlich-dimensionalen Raum V_h, die wir als gewöhnliche Differentialgleichung im $\mathbb{R}^N$, $N = \dim V_h$, schreiben und lösen können. Diese Gleichung wird als Semidiskretisierung bezeichnet. Der Raum V_h wird dabei wie bei den elliptischen Gleichungen gewählt, also z.B. die stetigen stückweisen Polynome auf Dreiecks- oder Tetraidergittern.

In ihrer schwachen Form geschrieben sieht diese Semidiskretisierung für die Wärmeleitungsgleichung wie folgt aus:

$$\langle \partial_t u_h(t,\cdot), v_h\rangle_{L_2} + \langle \nabla u_h, \nabla v_h\rangle_{L_2} = \langle f(t,\cdot), v_h\rangle_{L_2} \text{ für alle } v_h \in V_h \text{ und alle } t \in (0,T)$$

Man kann nun unter geeigneten Regularitätsannahmen mit ähnlichen Methoden wie für elliptische Gleichungen beweisen, dass die Funktionen u_h für Gitter $\mathcal{T}_{h_k}$ mit $h = h_k \to 0$ gegen u konvergieren, siehe z.B. [10, Abschnitt 3.3].

Wählt man wie üblich eine Basis $(\phi_j)_{j=1,\dots,N}$ von V_h und schreibt $u_h = \sum_{j=1}^N \underline{u}_j \phi_j$ und $\underline{u} = (\underline{u}_1, \dots, \underline{u}_N)$, wobei die $\underline{u}_j$ nun zeitabhängig sind, so können wir die Gleichung umschreiben zu

$$B\dot{\underline{u}}(t) = -A\underline{u}(t) + b(t) \tag{15.2}$$

mit Steifigkeitsmatrix $A = (\langle \nabla\phi_i, \nabla\phi_j\rangle_{L_2})_{i,j=1,\dots,N}$, Massematrix $B = (\langle \phi_i, \phi_j\rangle_{L_2})_{i,j=1,\dots,N}$ und $b = (\langle f(t,\cdot), \phi_1\rangle_{L_2}, \dots, \langle f(t,\cdot), \phi_N\rangle_{L_2})^T$. Nach Multiplikation mit B^{-1} ergibt sich eine gewöhnliche Differentialgleichung

$$\dot{\underline{u}}(t) = -B^{-1}A\underline{u}(t) + B^{-1}b(t) \tag{15.3}$$

in Standardform im $\mathbb{R}^n$, die wir mit den üblichen Methoden numerisch lösen können, wozu in der Regel Einschrittverfahren verwendet werden. Zu beachten ist dabei allerdings, dass $B^{-1}A$ Eigenwerte der Größenordnung $-1/h_j^2$ besitzt. Dies sieht man wie folgt: Die Basisfunktionen nehmen stets den Wert 1 an einem Knoten eines Teilgebiets an und 0 auf den anderen. Die Steigung ist daher umgekehrt proportional zum Durchmesser h_T der Teilgebiete, also von der Größenordnung $1/h_j$. Die Skalarprodukte $\langle \nabla\phi_i, \nabla\phi_j\rangle_{L_2}$ sind daher von der Größenordnung (Größe der Teilgebiete)$/h_j^2$, oder gleich 0. Die Funktionswerte der ϕ_i hingegen sind unabhängig von h_j beschränkt. Daher sind die Einträge von B, also die $\langle \phi_i, \phi_j\rangle_{L_2}$, von der Größenordnung der Größe der Teilgebiete, die von B^{-1} also von der Größenordnung 1/(Größe der Teilgebiete). Folglich haben alle Einträge von $B^{-1}A$ betragsmäßig die Größenordnung $1/h_j^2$ und damit auch die Eigenwerte. Dass die Eigenwerte negativ sind, sieht man durch Betrachtung der Vorzeichen der Einträge von A und B, was wir hier nicht im Detail machen wollen. Es folgt, dass $B^{-1}A$ für feine Diskretisierungen betragsmäßig große negative Eigenwerte besitzt. Also ist die gewöhnliche Differentialgeichung (15.3) eine steife Gleichung und muss daher implizit diskretisiert werden.

Verwenden wir z.B. das implizite Euler-Verfahren mit Schrittweite $\tau > 0$ zur Lösung von (15.3), so erhalten wir

$$\underline{u}^{k+1} = \underline{u}^k - \tau B^{-1} A \underline{u}^{k+1} + \tau B^{-1} b(t_{k+1})$$

oder, äquivalent,

$$B\underline{u}^{k+1} = B\underline{u}^k - \tau A \underline{u}^{k+1} + \tau b(t_{k+1}).$$

Dabei ist $t_k = \tau k$, $\tau > 0$ der Zeitschritt und $\underline{u}^k$ die Näherung von $\underline{u}(t_k)$. Zur Bestimmung von u^k müssen wir also das lineare Gleichungssystem

$$(B + \tau A)\underline{u}^{k+1} = B\underline{u}^k + b(t_{k+1})$$

lösen.

Analog zu den finiten Differenzen können wir für $\theta \in (0,1)$ das θ-Verfahren

$$\underline{u}^{k+1} = \underline{u}^k + (1-\theta)\Big(- \tau B^{-1} A \underline{u}^k + \tau B^{-1} b(t_k)\Big) + \theta\Big(- \tau B^{-1} A \underline{u}^{k+1} + \tau B^{-1} b(t_{k+1})\Big)$$

definieren. Dies führt mittels

$$B\underline{u}^{k+1} = B\underline{u}^k + (1-\theta)\Big(- \tau A \underline{u}^k + \tau b(t_k)\Big) + \theta\Big(- \tau A \underline{u}^{k+1} + \tau b(t_{k+1})\Big)$$

auf das lineare Gleichungssystem

$$(B + \theta\tau A)\underline{u}^{k+1} = B\underline{u}^k + (1-\theta)\Big(\tau A \underline{u}^k + \tau b(t_k)\Big) + \theta\tau b(t_{k+1}).$$

Für $\theta = 1$ erhalten wir wieder das implizite Euler-Verfahren, für $\theta = 1/2$ das Crank-Nicolson-Verfahren bzw. die implizite Mittelpunktregel, die wie bei den finiten Differenzen die höhere Ordnung $O(\tau^2)$ in der Zeit liefert, allerdings auch hier mit dem Nachteil, dass sie nicht L-stabil ist.

Prinzipiell könnte man hier auch Verfahren für gewöhnliche Differentialgleichungen mit mehr Stufen (und entsprechend höherer Genauigkeit) anwenden. Das würde aber die Anzahl der in jedem Schritt zu lösenden linearen Gleichungsysteme erhöhen, weswegen dies eher selten gemacht wird.

Die Linienmethode verdankt ihren Namen der Tatsache, dass das Gitter feste Knotenpunkte im Ort definiert (die Eckpunkte der Dreiecke bzw. Tetraeder), die im (t,x)-Raum dann zu Linien werden. Verwenden wir z.B. wie bei den stückweise linearen Funktionen in Abschnitt (12.4) Basisfunktionen ϕ_j, die in genau einem Knotenpunkt des Gitters gleich 1 und in allen anderen gleich 0 sind, und bezeichnen wir den zu ϕ_j gehörigen Basispunkt mit x_j, so ist $\underline{u}_j(t) = u_h(t, x_j)$. Dies ist gerade der Wert von u_h entlang der Linie (t, x_j), $t \geq 0$ im (t,x)-Raum.

Der größte Vorteil der Linienmethode ist ihre einfache Implementierung. Der größte Nachteil ist, dass das räumliche Gitter zu allen Zeitpunkten gleich sein muss.

15.2 Die Schichten- oder Rothe-Methode

Die Schichten- oder Rothe-Methode geht gerade umgekehrt vor wie die Linienmethode. Wir diskretisieren erst in der Zeit mit einem Einschrittverfahren und dann im Raum. Dadurch wird ermöglicht, für jeden Zeitschritt ein unterschiedliches räumliches Gitter zu verwenden.

Dazu wenden wir die zeitliche Diskretisierung innerhalb der schwachen Formulierung der Gleichung gemäß Definition 15.1 an. Im Fall des impliziten Euler-Verfahrens[1] führt das auf die Gleichung

$$\left\langle \frac{u^{k+1} - u^k}{\tau}, v \right\rangle_{L_2} + \langle \nabla u^{k+1}, \nabla v \rangle_{L_2} = \langle f(t_{k+1}, \cdot), v \rangle_{L_2} \text{ für alle } v \in V \text{ und } k = 0, \ldots, K-1$$

wobei die u^k nun Funktionen aus $H_0^1(\Omega)$ sind und K die Anzahl der Zeitschritte bezeichnet. Diese Gleichung können wir umschreiben zu

$$\langle u^{k+1}, v \rangle_{L_2} + \tau \langle \nabla u^{k+1}, \nabla v \rangle_{L_2} = \langle \tau f(t_{k+1}, \cdot) + u^k, v \rangle_{L_2}. \tag{15.4}$$

Die gesuchte Funktion u^{k+1} erfüllt also wieder eine elliptische partielle Differentialgleichung, die im Vergleich zur Wärmeleitungsgleichung den zusätzlichen Term $\langle u^{k+1}, v \rangle_{L_2}$ enthält. Beachte, dass dieser dem Term $c(x)u$ in (12.1) entspricht. Wir haben also eine elliptische Gleichung im Sinne der vorherigen Kapitel vorliegen.

Dies zeigt, wie diese Gleichung im Raum diskretisiert werden kann, nämlich genau so wie die elliptischen Gleichungen in Abschnitt 12.3. Wir wählen also Finite-Element-Räume V_k für jeden Zeitpunkt t_k und lösen

$$\langle u_h^{k+1}, v_h \rangle_{L_2} + \tau \langle \nabla u_h^{k+1}, \nabla v_h \rangle_{L_2} = \langle \tau f(t_{k+1}, \cdot) + u_h^k, v_h \rangle_{L_2} \text{ für alle } v_h \in V_{k+1}.$$

Umgeschrieben in ein lineares Gleichungssystem mit Basis $(\phi_j^k)_{k=1,\ldots,d_k}$ von V_k und $u_h^k = \sum_{j=1}^{d_j} \underline{u}_j^k \phi_j^k$ liefert dies (nach Übergang von $k+1$ zu k, um die Notation zu vereinfachen)

$$(B_k + \tau A_k)\underline{u}^k = b_k$$

mit

$$A_k = (\tau \langle \nabla \phi_i^k, \nabla \phi_j^k \rangle_{L_2})_{i,j=1,\ldots,d_k}, B_k = (\langle \phi_i^k, \phi_j^k \rangle_{L_2})_{i,j=1,\ldots,d_k}$$

[1] Alternativen zum impliziten Euler-Verfahren betrachten wir im nächsten Abschnitt.

und

$$b_k = (\langle \tau f(t_k, \cdot) + u_h^{k-1}, \phi_j^k \rangle_{L_2})_{j=1,\dots,d_k}.$$

Falls $V_k = V_{k-1}$ ist, kann man b^k umschreiben zu

$$(\langle \tau f(t_k, \cdot) + u_h^{k-1}, \phi_j^k \rangle_{L_2})_{j=1,\dots,d_k} = (\langle \tau f(t_k, \cdot), \phi_j^k \rangle_{L_2})_{j=1,\dots,d_k} + B^k \underline{u}^{k-1}.$$

Ein Vergleich zeigt, dass in diesem Fall das Gleichungssystem identisch mit dem aus der Linienmethode ist.

Man kann beweisen, dass die Lösungen u^k für $\tau \to 0$ gegen $u(t_k, \cdot)$ konvergieren, siehe [10, Abschnitt 3.2]. Effizienter für die Fehlerabschätzung ist es hier aber, den Fehler nach erfolgter Ortsdiskretisierung gemeinsam für die Zeit- und Ortsdiskretisierung abzuschätzen, siehe z.B. [12, Satz 5.5] für quadratische finite Elemente. Für diese Elemente erhält man für den Fall, dass alle räumlichen Gitter identisch sind, einen Fehler der Ordnung $O(h^2 + \tau)$, während man im Fall unterschiedlicher Gitter einen Fehler der Ordnung $O(h^2/\sqrt{\tau} + \tau)$ erhält. Der zusätzliche Faktor ergibt sich aus der Tatsache, dass beim Wechsel der Gitters zusätzliche Fehler bei der Auswertung von $\langle \tau u_h^{k-1}, \phi_j^k \rangle_{L_2}$ entstehen, weil $u_h^{k-1} \in V_{k-1}$ ist, die ϕ_j^k aber eine Basis von $V_k \neq V_{k-1}$ bilden. Man muss also u_h^{k-1} vom Gitter, das V_{k-1} zu Grunde liegt, auf das zu V_k gehörige Gitter umrechnen.

Der Vorteil der Rothe-Methode liegt offenkundig darin, dass in jedem Zeitschritt ein unterschiedliches Gitter verwendet werden kann, was die adaptive Anpassung des Ortsgitters über die Zeit ermöglicht. Dafür ist aber die Implementierung und die Fehleranalyse komplizierter.

15.3 Das unstetige Galerkin-Verfahren

Als letztes lernen wir mit dem unstetigen Galerkin-Verfahren (engl. discontinuous Galerkin, oft abgekürzt als DG) eine Methode kennen, die eine Alternative zum impliziten Euler-Verfahren in der Rothe-Methode darstellt. Es handelt sich also um keinen vollständig neuen Ansatz, sondern um eine Variante der Rothe-Methode.

Die Idee beruht darauf, die zeitliche Diskretisierung nicht (oder jedenfalls nicht direkt) mit einem Einschrittverfahren, sondern wie die räumliche Diskretisierung mit der Finiten-Elemente-Methode und dem Galerkin-Ansatz durchzuführen. Dazu schreiben wir die zu Grunde liegende Gleichung — hier wieder die Wärmeleitungsgleichung (15.1) in der schwachen Form aus Definition 15.1 — zusätzlich in schwacher Form bezüglich der Zeit:

$$\int_0^T \langle \partial_t u(t, \cdot) - f(t, x), v(t, \cdot) \rangle_{L_2} + \langle \nabla u(t, \cdot), \nabla v(t, \cdot) \rangle_{L_2} dt = 0 \text{ für alle } v \in L_2(0, T; H_0^1(\Omega)).$$

Die Anfangsbedingung $u(0, x) = u_0(x)$ können wir ebenfalls in schwacher Form schreiben, nämlich als $\langle u(0, \cdot) - u_0, v_0 \rangle_{L_2} = 0$ für alle $v_0 \in L_2(\Omega)$. Lassen wir zur Vereinfachung der Notation die Argumente der Funktionen unter dem Integral weg, so gilt für die Lösung also

$$\int_0^T \langle \partial_t u - f, v \rangle_{L_2} + \langle \nabla u, \nabla v \rangle_{L_2} dt + \langle u(0, \cdot) - u_0, v_0 \rangle_{L_2} = 0$$

für alle $v \in L_2(0,T;H_0^1(\Omega))$ und $v_0 \in L_2(\Omega)$. Man kann beweisen, dass die schwache Lösung von (15.1) stetig in t ist (ggf. nach Abänderung auf einer Nullmenge). Wählen wir also ein Zeitgitter $0 = t_0 < t_1 < \ldots < t_K = T$, so gilt für jeden Zeitpunkt t_k mit den Bezeichnungen

$$u_k^- := \lim_{\substack{t\to t_k \\ t\le t_k}} u(t,\cdot), \quad u_k^+ := \lim_{\substack{t\to t_k \\ t\ge t_k}} u(t,\cdot) \quad \text{und} \quad [u]_k = \begin{cases} u_k^+ - u_k^-, & k = 1,\ldots,K-1 \\ u_0^+ - u_0, & k = 0 \end{cases}$$

die Gleichung $[u]_k = 0$, oder in schwacher Form $\langle [u]_k, v_k\rangle_{L_2} = 0$ für alle $v_k \in L_2(\Omega)$. Insbesondere gilt dies natürlich für $v_k = v_k^+ := \lim_{\substack{t\to t_k \\ t\ge t_k}} v(t,\cdot)$ für jedes $v \in L_2(Q)$, für das dieser Grenzwert existiert. Die schwache Lösung der Wärmeleitungsgleichung erfüllt also

$$\sum_{k=0}^{K-1} \left(\int_{t_k}^{t_{k+1}} \langle \partial_t u - f, v\rangle_{L_2} + \langle \nabla u, \nabla v\rangle_{L_2} dt + \left\langle [u]_k, v_k^+ \right\rangle_{L_2} \right) = 0 \tag{15.5}$$

für alle $v \in L_2(0,T;H_0^1(\Omega))$ mit rechtsseitigen Grenzwerten in den t_k und $v_0 \in L_2(\Omega)$. Dies ist nun die Form der Gleichung, auf die die unstetige Galerkin-Diskretisierung angewendet werden kann. Wir definieren dazu den Raum der Ansatzfunktionen

$$V_r = \left\{ v \in L_2(0,T;H_0^1(\Omega)) \;\middle|\; \begin{array}{l} v|_{(t_k,t_{k+1})} \text{ ist ein Polynom vom Grad} \le r \\ \text{in } t \text{ für alle } k = 0,\ldots,K-1 \end{array} \right\}.$$

Beachte, dass wir im Gegensatz zu den Definitionen der verschiedenen V_h in Abschnitt 12.4 hier keine Stetigkeit der Funktionen in V_r gefordert haben. Das begründet die Bezeichnung "unstetiges" Galerkin-Verfahren. Wie bisher erhält man die diskretisierte Form der Gleichung nun dadurch, dass man sowohl u als auch die Testfunktionen v in (15.5) aus V_r, also als $u = u_r \in V_r$ und $v = v_r \in V_r$ wählt.

Wir wollen den einfachsten Fall $r = 0$ genauer anschauen. In diesem Fall sind u_r, $v_r \in V_r$ stückweise konstante Funktionen. Bezeichnen wir den Wert von u_r und v_r auf dem Intervall (t_k, t_{k+1}) mit u_r^k bzw. v_r^k, so erhalten wir mit der Schrittweite $\tau_k := t_{k+1} - t_k$

$$\int_{t_k}^{t_{k+1}} \langle \partial_t u - f, v\rangle_{L_2} + \langle \nabla u, \nabla v\rangle_{L_2} dt = \tau_k \langle \nabla u_r^k, \nabla v_r^k\rangle_{L_2} - \int_{t_k}^{t_{k+1}} \langle f, v_r^k\rangle_{L_2} dt$$

sowie

$$\left\langle [u]_k, v_k^+ \right\rangle_{L_2} = \left\langle u_r^k - u_r^{k-1}, v_r^k \right\rangle_{L_2} \text{ für } k \ge 1 \quad \text{und} \quad \left\langle [u]_0, v_0^+ \right\rangle_{L_2} = \left\langle u_r^0 - u_0, v_r^0 \right\rangle_{L_2}.$$

Definieren wir $u_r^{-1} := u_0$, so lauten die Summanden in (15.5)

$$\tau_k \langle \nabla u_r^k, \nabla v_r^k\rangle_{L_2} - \int_{t_k}^{t_{k+1}} \langle f(t,\cdot), v_r^k\rangle_{L_2} dt + \left\langle u_r^k - u_r^{k-1}, v_r^k \right\rangle_{L_2}$$

Diese sind genau dann gleich 0 für alle v_r^k, wenn

$$\langle u_r^k, v_r^k\rangle_{L_2} + \tau_k \langle \nabla u_r^k, \nabla v_r^k\rangle_{L_2} = \left\langle \int_{t_k}^{t_{k+1}} f(t,\cdot) dt + u_r^{k-1}, v_r^k \right\rangle_{L_2}$$

gilt. Ein Vergleich mit (15.4) zeigt, dass dies genau die Diskretisierung mit dem impliziten Euler-Verfahren ergibt, wenn wir $\int_{t_k}^{t_k+1} f(t,\cdot) dt$ durch die Näherung $\tau_k f(t_{k+1},\cdot)$ ersetzen.

Das implizite Euler-Verfahren kann also als Spezialfall der unstetigen Galerkin-Methode angesehen werden. Der Vorteil dieser Methode ist aber natürlich, dass man durch Erhöhung von r Verfahren höherer Ordnung erhält. Wendet man die Methode auf gewöhnliche Differentialgleichungen an, erhält man — ähnlich wie bei der Kollokation — wieder (implizite) Runge-Kutta-Verfahren. Wir haben also wieder Einschrittverfahren erzeugt, allerdings mit einer besonderen Struktur.

Diese lässt sich insbesondere bei der Konvergenzanalyse ausnutzen, die so ähnlich funktioniert wie bei der räumlichen Diskretisierung. Allerdings ist sie etwas komplizierter, weil der Ansatzraum V_r hier kein Teilraum des Lösungsraums ist, weil die Lösungen ja stetig sind. Man spricht daher auch von *nichtkonformen* finiten Elementen. Näheres zur Konvergenztheorie findet sich z.B. in Abschnitt 5.2.2 von [12].

Literaturverzeichnis

[1] Aulbach, B.: *Gewöhnliche Differenzialgleichungen.* 2. Auflage. Elsevier–Spektrum Verlag, Heidelberg, 2004

[2] Braess, D.: *Finite Elemente.* Springer, 1992

[3] Deuflhard, P. ; Bornemann, F.: *Numerische Mathematik. II: Integration gewöhnlicher Differentialgleichungen.* 4. Auflage. de Gruyter, Berlin, 2013

[4] Deuflhard, P. ; Weiser, M.: *Numerische Mathematik III. Adaptive Lösung partieller Differentialgleichungen.* De Gruyter, 2011

[5] Evans, L. C.: *Partial Differential Equations.* American Mathematical Society, 1998

[6] Grüne, L. ; Junge, O.: *Gewöhnliche Differentialgleichungen. Eine Einführung aus der Perspektive der Dynamischen Systeme.* Vieweg + Teubner Verlag, 2009

[7] Hairer, E. ; Lubich, C. ; Wanner, G.: *Geometric numerical integration. Structure-preserving algorithms for ordinary differential equations.* 2nd edition. Springer-Verlag, Berlin, 2006

[8] Hairer, E. ; Wanner, G.: *Solving ordinary differential equations. II. Stiff and differential-algebraic problems.* 2nd edition. Springer-Verlag, Berlin, 1996

[9] Karafyllis, I. ; Grüne, L.: Feedback stabilization methods for the numerical solution of systems of ordinary differential equations. In: *Discrete Contin. Dyn. Syst. Ser. B* 16 (2011), Nr. 1, S. 283–317

[10] Ohlberger, M.: *Numerik partieller Differentialgleichungen I.* Vorlesungsskript, Universität Münster, 2013. – Available from https://www.uni-muenster.de/AMM/num/Vorlesungen/PDEI_WS1213/skriptum.pdf

[11] Oswald, P.: On discrete norm estimates related to multilevel preconditioners in the nite element method. In: *Proceedings of the International Conference on Constructive Theory of Functions.* Varna, Bulgaria, 1991, S. 203–214

[12] Rannacher, R.: *Numerische Mathematik 2 (Numerik partieller Differentialgleichungen).* Vorlesungsskript, Universität Heidelberg, 2008. – Available from https://ganymed.math.uni-heidelberg.de/∼lehre/notes/num2/numerik2.pdf

[13] Schober, M.: *Stabilitätsuntersuchung numerischer Einschrittverfahren mit Hilfe von Ljapunov-Funktionen*, Universität Bayreuth, Diplomarbeit, 2012

[14] Strauss, W. A.: *Partielle Differentialgleichungen.* Vieweg Verlag, 1995

Index