Numerische Mathematik – Teil II

nach einer Vorlesung von

Prof. Dr. Angela Kunoth

im Sommersemester 2018
an der
Universität zu Köln

Version vom 20. Juni 2018

Inhaltsverzeichnis

Teil 1: Basiskonzepte der Numerik **4**

0 Einführung **4**

9 Iterative Verfahren zur Lösung nichtlinearer Gleichungen **7**
9.1 Einführung . . . 7
9.2 Kondition des skalaren Nullstellenproblems . . . 7
9.3 Fixpunktiteration . . . 8
9.4 Konvergenzordnung und Fehlerschätzung . . . 11
9.5 Iterationsmethoden für eine skalare Gleichung . . . 13
9.6 Das Newton–Verfahren für Systeme . . . 17

10 Nichtlineare Ausgleichsrechnung **25**

11 Iterative Lösung linearer Gleichungssysteme **27**
11.1 Motivation und Einführung . . . 27
11.2 Klassische Iterationsverfahren . . . 27
11.3 Gradientenverfahren . . . 31
11.4 cg–Verfahren . . . 36

Teil 2: Numerik gewöhnlicher Differentialgleichungen **44**

1 Modellierung **44**

2 Theoretische Grundlagen **49**

3 Einschrittverfahren zur Lösung von Anfangswertaufgaben **53**
3.1 Die Euler'sche Polygonzugmethode . . . 53
3.2 Einschrittverfahren . . . 53
3.3 Konvergenz von Einschrittverfahren . . . 55
3.4 Die Methode der Taylor–Entwicklung . . . 59
3.5 Explizite Runge–Kutta–Verfahren . . . 60
3.6 Implizite Runge–Kutta–Verfahren . . . 64
3.7 Asymptotische Entwicklung, Extrapolation und Schrittweitensteuerung . . . 70

4 Mehrschrittverfahren zur Lösung von Anfangswertaufgaben **72**
4.1 Definition von k–Schritt–Verfahren . . . 72
4.2 Lokaler Abbruchfehler und Konsistenz . . . 72
4.3 Allgemeine Konsistenzkriterien . . . 73
4.4 Konstruktion von Mehrschrittverfahren mittels numerischer Differentiation 78
4.5 Konstruktion von Mehrschrittverfahren über numerische Integration . . . 79
4.6 Prädiktor–Korrektor–Verfahren . . . 83
4.7 Konvergenz und Stabilität von Mehrschrittverfahren . . . 84

5 Steife Probleme **102**

6 Randwertprobleme **108**
6.1 Einleitung . . . 108

6.2 Das einfache Schießverfahren 109
6.3 Die Mehrziel(–schieß–)methode 110
6.4 Differenzenverfahren 110

Literaturverzeichnis **114**

Teil 1: Basiskonzepte der Numerik

0 Einführung

Wissenschaftliches Rechnen (Scientific Computing) ist eine angewandte Disziplin, zu der auch Computeralgebra, Computational Number Theory, Computational Chemistry o.ä. gehören. Dieser interdisziplinäre Wissenschaftszweig entwickelt sich seit ca. 10-25 Jahren.
Wissenschaftliches Rechnen umfasst Methoden, Techniken und Verfahren zur rechnergestützten Simulation mathematisch-technisch-naturwissenschaftlicher Phänomene. Dazu gehört unter anderem die Numerik von gewöhnlichen und partiellen Differentialgleichungen.

Es gibt verschiedene Möglichkeiten der Erkenntnisgewinnung: zunächst die klassischen Ansätze wie theoretisch mit Papier und Bleistift (deduktiver Weg) oder praktisch mittels physikalischer Experimente (induktiver Weg).
Die Nachteile und Schwächen dieser beiden Ansätze liegen klar auf der Hand: zum Einen lassen sich nicht alle Problemklassen behandeln. So ist es beispielsweise schwierig, experimentelle Versuche in der Teilchendynamik, der Virusdynamik oder bei turbulenten Strömungen durchzuführen. Zum Anderen sind deutlich Grenzen durch z.B. Größe (in der Astronomie) oder Geschwindigkeit (bei der Plattentektonik) gesetzt. Außerdem können schnell Gefahren für Gesundheit und Sicherheit entstehen. Hier sei als Beispiel an die Atomkraft erinnert. Zusätzlich enstehen sehr schnell sehr hohe Kosten.

An dieser Stelle kommt nun das Wissenschaftliche Rechnen ins Spiel, das uns einen dritten Ansatz zur Erlangung neuer Erkenntnisse bietet, nämlich über computergestützte Simulationen.
Die Vorteile hierbei sind, dass jederzeit relativ kostengünstig Experimente durchgeführt werden können, und bei diesen die Einflussparameter schnell modifiziert werden können. Hinzu kommt, dass man so viel mehr Daten verarbeiten kann, aber auch wesentlich mehr Daten erhält. Hierin liegt aber auch gleichzeitig ein Nachteil, da die große Datenmenge, die man als Ergebnis erhält, augewertet und visualisiert werden muss. Dazu muss häufig eine aufwendige Organisationsstruktur erarbeitet und realisiert werden.

Die Durchführung von numerischen Simulationen lässt sich in vier Schritte unterteilen:

1. Modellierung: Man braucht eine Abbildung, die die Realität in einem Modell darstellt, so dass man das Problem auf die wesentlichen Daten reduzieren und mit ihnen arbeiten kann. Dazu benötigt man (Mess-)Daten und erhält dann z.B. ein mathematisch-physikalisches Modell für "Dynamik", und muss nun Gleichungen lösen, wie z.B. $F(y) = f$. Oft führt dies auf gewöhnliche oder partielle Differentialgleichungen.

2. Theorie: Um ein Problem erfolgreich lösen zu können, benötigt man theoretisch die Existenz und Eindeutigkeit einer Lösung, da dies in der Realität schon sicher gegeben ist. Idealerweise ist das Problem wohlgestellt, und es liegt eine stetige Abhängigkeit der Lösung von den Anfangsdaten vor.

3. Numerik: Nun erfolgt die Diskretisierung und numerische Lösung von $F(y) = f$:
   Man muss sich zunächst Gedanken über die Konsistenz, die Stabilität und die Konvergenz machen, bevor man dann zur Implementierung des Algorithmus' kommt. Aufgrund der hohen Komplexität vieler Probleme benötigt man effiziente Lösungsverfahren, wie z.B. Multiskalenmethoden, parallele oder adaptive Algorithmen. Schließlich ist eine Visualisierung der Lösung nötig.

4. Validierung: In diesem letzten Schritt erfolgt dann der Abgleich der approximativen Lösung mit den aus 1. gegebenen Daten und gegebenenfalls eine Korrektur des Modells, wobei dann wieder bei Punkt 1. begonnen wird.

Hierzu ein Diagramm, in dem auch die möglichen Fehlerquellen gekennzeichnet sind:

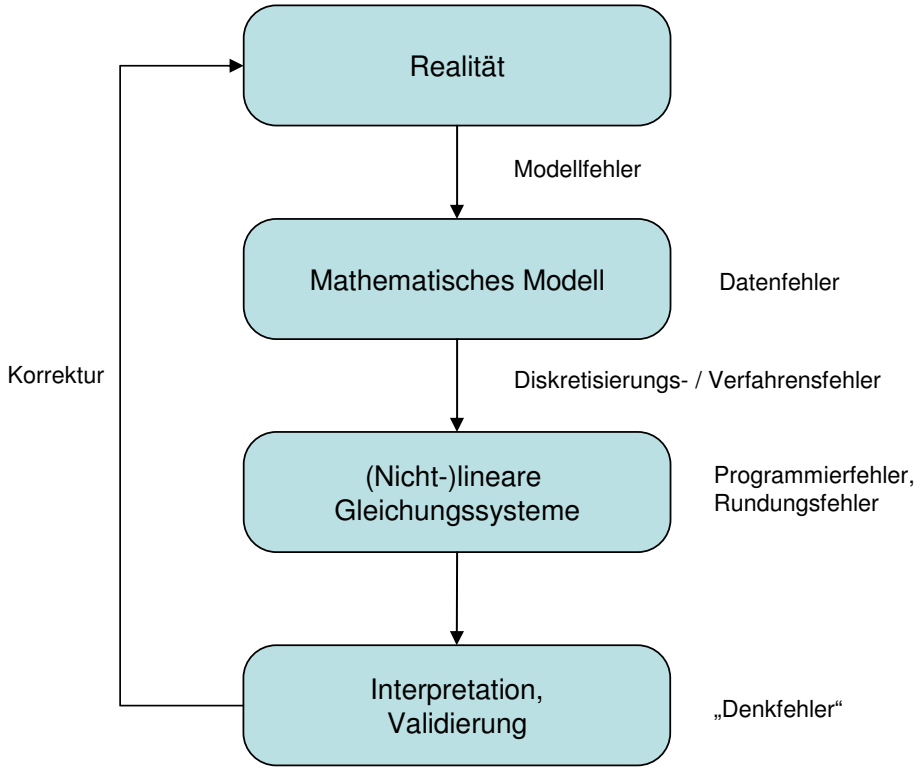


Abbildung 1: Durchführung einer numerischen Simulation

In dieser Vorlesung – als Weiterführung der Algorithmischen Mathematik und dem ersten Teil der Numerischen Mathematik – werden wir uns mit folgenden Inhalten beschäftigen:

- **Iterative Verfahren zur Lösung nichtlinearer Gleichungen:** Dieser Abschnitt behandelt die Bestimmung von Nullstellen von nichtlinearen Gleichungssystemen. Dies führen wir zunächst für den skalaren und später für den vektorwertigen Fall durch. Die theoretische Grundlage für die Existenz und Eindeutigkeit von Lösungen von Iterationsverfahren, die meist aus Fixpunktgleichungen gewonnen werden, ist der Banach'sche Fixpunktsatz. Das wichtigste Verfahren zur Lösung dieser Problematik ist das Newton–Verfahren, welches wir auch in einer Variante, nämlich dem gedämpften Newton–Verfahren diskutieren werden.
- **Nichtlineare Ausgleichsprobleme:** Die für lineare Ausgleichsprobleme gewonnenen Erkenntnisse werden hier derart erweitert, dass sie auf nichtlineare Gleichungssysteme angewandt werden können. Außerdem werden wir hier das Gauss–Newton–Verfahren kennenlernen.
- **Iterative Lösung linearer Gleichungssysteme:** Die direkten Lösungsverfahren, wie z.B. die LR–, QR– und Cholesky–Zerlegung sind in vielen Fällen numerisch zu aufwendig, da die zu lösenden Systeme zu groß oder nur sehr dünn besetzt sind. In solchen Fällen bieten sich iterative Verfahren an. Neben den klassischen Gesamt– und Einzelschrittverfahren sowie der Relaxation werden das Gradienten– und das darauf aufbauende cg–Verfahren besprochen.
- **Numerik gewöhnlicher Differentialgleichungen:** Zunächst werden wir zwei interessante Beispiele gewöhnlicher Differentialgleichungen betrachten. Zur numerischen Lösung

werden wir hier Einschrittverfahren, insbesondere das EULER–Verfahren und das RUNGE–KUTTA–Verfahren, und Mehrschrittverfahren, wie die Mittelpunktsregel oder die BDF–Verfahren, kennen lernen.

9 Iterative Verfahren zur Lösung nichtlinearer Gleichungen

9.1 Einführung

Bisher haben wir lediglich Verfahren zur Lösung *linearer* Gleichungssysteme behandelt. Nun gehen wir den Bereich der *nichtlinearen* Gleichungen und deren numerische Behandlung an. Das einfachste Beispiel für einen solchen Fall ist die Berechnung der Nullstellen der Funktion

$$f(x) = ax^2 + bx + c,$$

wobei $a, b, c \in \mathbb{R}$ fest vorgegeben seien. Die Nichtlinearität dieser Gleichung ist durch das quadratische Auftreten der Variablen x gegeben. Im Allgemeinen hat man ein nichtlineares System in n Unbekannten $x_1, \ldots, x_n$ mit r Gleichungen gegeben, also eine Funktion $f : \mathbb{R}^n \to \mathbb{R}^r$, und man sucht die Lösungen der Gleichung

$$f(x) = 0, \tag{9.1.1}$$

wobei

$$f(x) := \begin{pmatrix} f_1(x_1, \ldots, x_n) \\ \vdots \\ f_r(x_1, \ldots, x_n) \end{pmatrix}$$

ist. Für letzteres schreibt man auch $f = (f_1, \ldots, f_r)^T$. In dem speziellen Fall $r = n = 1$ hat man es mit einem skalaren Problem zu tun.

9.2 Kondition des skalaren Nullstellenproblems

Gegeben sei eine Funktion $f : \mathbb{R} \to \mathbb{R}$ und x^* sei eine lokal eindeutige Nullstelle, d.h.

$$f(x^*) = 0. \tag{9.2.1}$$

Sei $\tilde{f}$ eine Störung von f, der Gestalt, dass

$$|\tilde{f}(x) - f(x)| \leq \varepsilon \tag{9.2.2}$$

für alle x aus einer Umgebung $U(x^*)$ gilt. Wir fragen nun, welchen Einfluss diese Störung auf die Berechnung der Nullstelle hat. Bei der Analyse dieses Problems müssen wir berücksichtigen, dass der *relative* Fehler zu dessen Auswertung ungeeignet ist, da der Quotient

$$\frac{|\tilde{f}(x) - f(x)|}{|f(x)|}$$

im Allgemeinen unbeschränkt für $x \to x^*$ ist, weil $f(x^*)$ verschwindet.
Also betrachten wir nur den absoluten Fehler: sei dazu $\tilde{x}^*$ eine Nullstelle von $\tilde{f}$. Dann gilt mit (9.2.2)

$$|f(\tilde{x}^*)| \leq \varepsilon. \tag{9.2.3}$$

Sei weiter m die Vielfachheit dieser Nullstelle x^*. Das bedeutet $f^{(i)}(x^*) = 0$ für alle $i = 0, \ldots, m-1$ und $f^{(m)}(x^*) \neq 0$. Mit einem ξ zwischen x^* und $\tilde{x}^*$ gilt für die TAYLOR–Entwicklung von f um x^*

$$\begin{aligned} f(\tilde{x}^*) &= \sum_{i=0}^{m-1} \frac{(\tilde{x}^* - x^*)^i}{i!} f^{(i)}(x^*) + \frac{(x^* - \tilde{x}^*)^m}{m!} f^{(m)}(\xi) \\ &\approx \frac{(x^* - \tilde{x}^*)^m}{m!} f^{(m)}(x^*). \end{aligned}$$

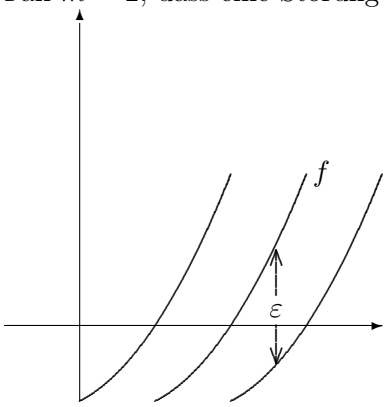
Damit erhalten wir

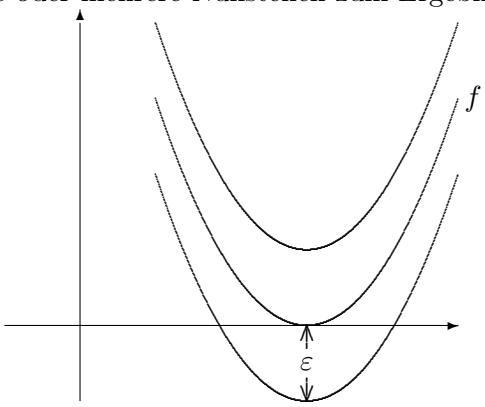
$$\left|\frac{(x^* - \tilde{x}^*)^m}{m!} f^{(m)}(x^*)\right| \approx |f(\tilde{x}^*)| \leq \varepsilon$$

nach (9.2.3) und somit gilt

$$|x^* - \tilde{x}^*| \leq \varepsilon^{\frac{1}{m}} \left|\frac{m!}{f^{(m)}(x^*)}\right|^{\frac{1}{m}}. \tag{9.2.4}$$

Hieraus lesen wir ab, dass das Problem gut konditioniert ist, wenn $m = 1$ ist, und schlecht konditioniert, falls $m > 1$ ist. Dies können wir uns auch an folgenden Skizzen klarmachen. Links sehen wir den Fall $m = 1$, wo eine Störung von f im Rahmen bleibt. Rechts dagegen gilt für den Fall $m = 2$, dass eine Störung von f keine oder mehrere Nullstellen zum Ergebnis haben kann.





Hierzu betrachten wir noch ein Beispiel.

Beispiel 9.2.5. Seien $f(x) := (x-1)^3$ mit der Nullstelle $x^* = 1$ und $\tilde{f}(x) := (x-1)^3 - \varepsilon$ mit der Nullstelle $\tilde{x}^* = 1 + \varepsilon^{\frac{1}{3}}$. Für $\varepsilon = 10^{-12}$ ist

$$|f(x) - \tilde{f}(x)| = 10^{-12}$$

aber

$$|x^* - \tilde{x}^*| = 10^{-\frac{12}{3}} = 10^{-4},$$

was eine enorme Fehlerverstärkung bedeutet. ■

9.3 Fixpunktiteration

Da man in aller Regel keine Eliminationstechniken zur Lösung von $f(x) = 0$ zur Verfügung hat, muss man nach anderen Strategien suchen. Hier bieten sich vor allem iterative Verfahren an, die sich Schritt für Schritt einer Nullstelle nähern. Die Grundidee hierbei ist, dass man (9.2.1) für eine Funktion $f : \mathbb{R}^n \to \mathbb{R}^n$ von der Form eines Nullstellenproblems

$$f(x^*) = 0$$

in die Form eines Fixpunktproblems

$$\Phi(x^*) = x^* \tag{9.3.1}$$

überführt. Um die Funktion Φ zu bestimmen, wählt man dabei den Ansatz

$$\Phi(x) := x - g(x)f(x), \tag{9.3.2}$$

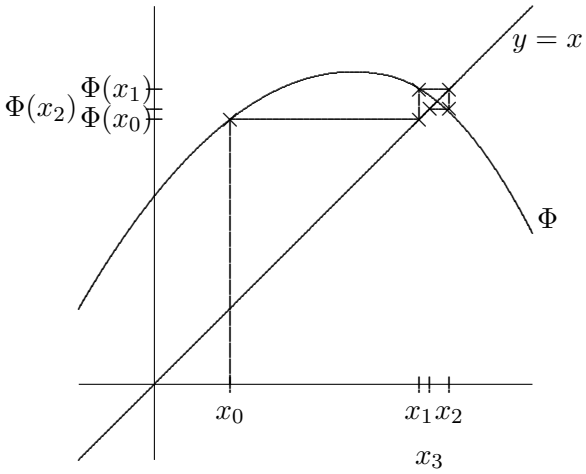
wobei $g(x) \in \mathbb{R}^{n\times n}$ und $\det g(x) \neq 0$ für alle x aus einer Umgebung $U(x^*)$ vorausgesetzt sind. Man beginnt die Iteration, indem man einen Startwert x_0 wählt und dann für $k = 0, 1, 2, \ldots$ die Werte

$$x_{k+1} = \Phi(x_k)$$

berechnet. Hierbei hofft man, dass bei geeigneter Wahl von x_0 die Folge $(x_k)_{k\in\mathbb{N}}$ für $k \to \infty$ gegen x^* konvergiert. Geometrisch stellen wir uns das für $n = 1$ folgendermaßen vor: Die Aussage, Φ hat einen Fixpunkt, bedeutet, dass Φ die Gerade $y = x$ schneidet.

Hierbei ist zu beachten, dass im Fall $|\Phi'(x)| < 1$ für alle $x \in U := \overline{U(x^*)}$ wegen des Mittelwertsatzes gilt, dass für alle $x, y \in U$ ein ξ existiert mit

$$\begin{aligned} |\Phi(x) - \Phi(y)| &= |\Phi'(\xi)(x-y)| \\ &\leq \max_{t\in U} |\Phi'(t)||x-y|. \end{aligned}$$

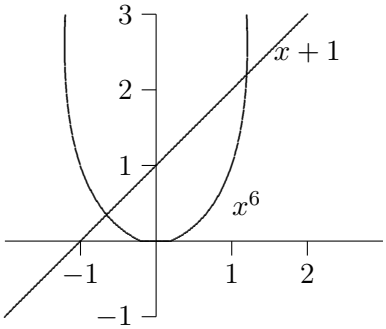


Dies bedeutet, dass Φ kontrahierend ist. In diesem Fall nennt man Φ LIPSCHITZ–stetig und bezeichnet den Term $\max_{t\in U} |\Phi'(t)|$ suggestiv mit L. Im Fall $|\Phi'(x)| > 1$ kann Φ zwar immer noch LIPSCHITZ–stetig sein, jedoch ist Φ dann keine Kontraktion mehr.

Diese Konvergenzeigenschaften wollen wir an einem Beispiel aus [DR] konkret beobachten:

Beispiel 9.3.3. Sei $f(x) := x^6 - x - 1$. Diese Funktion verschwindet genau dann, wenn $x^6 = x + 1$ ist. Aus der Grafik lesen wir ab, dass f *eine* positive Nullstelle im Intervall $[1, 2]$ haben muss. Wir testen zwei Iterationsfunktionen zur Berechnung dieser Nullstelle:

$$\begin{aligned} \Phi_1(x) &= x^6 - 1 \\ \Phi_2(x) &= (x+1)^{\frac{1}{6}}. \end{aligned}$$



Nun gilt $|\Phi_1'(x)| = |6x^5| > 1$ für alle $x \in [1, 2]$ und $|\Phi_2'(x)| = |\frac{1}{6}(x+1)^{-\frac{5}{6}}| \leq \frac{1}{6}$ für alle $x \in [1, 2]$. Mit Φ_2 können wir also eine Kontraktion mit $L = \frac{1}{6}$ erreichen. Die Werte der folgenden Tabelle geben Aufschluss über das Konvergenzverhalten der beiden Iterationsfunktionen.

k	$x_0 = 0.5$ $x_{k+1} = \Phi_2(x_k)$	$x_0 = 0.5$ $x_{k+1} = \Phi_1(x_k)$	$x_0 = 1.13$ $x_{k+1} = \Phi_1(x_k)$	$x_0 = 1.135$ $x_{k+1} = \Phi_1(x_k)$
0	0.50000000	0.50000000	1.13000000	1.14e+00
1	1.06991319	–0.98437500	1.08195175	1.14e+00
2	1.12890836	–0.09016330	0.60415884	1.17e+00
3	1.13420832	–0.99999946	–0.95136972	1.57e+00
4	1.13467844	–0.00000322	–0.25852598	1.38e+01
5	1.13472009	–1.00000000	–0.99970144	6.91e+06
6	1.13472378	0.00000000	–0.00179000	1.09e+41
7	1.13472411	–1.00000000	–1.00000000	1.69e+246

Interessanterweise divergiert bei der Funktion Φ_1 selbst der fast exakte Startwert $x_0 = 1.135$ in sehr starkem Maße. ∎

Die bisherigen, mehr heuristischen Überlegungen dieses Abschnitts werden wir nun mit dem BANACH'schen Fixpunktsatz theoretisch fundieren. Sei X ein linearer normierter Raum mit

Norm $\|\cdot\|$ und $E \subseteq X$ eine vollständige Teilmenge von X. Es gelte

$$\Phi : E \to E \tag{9.3.4}$$

und

$$\|\Phi(x) - \Phi(y)\| \leq L\|x - y\| \tag{9.3.5}$$

für alle $x, y \in E$ und $L < 1$. Die Kontraktion Φ ist also auf E LIPSCHITZ–stetig mit einer LIPSCHITZ–Konstanten kleiner eins. Dann gilt

1.) Es existiert genau ein *Fixpunkt* x^* von Φ in E, also

$$\Phi(x^*) = x^*.$$

2.) Für beliebiges $x_0 \in E$ konvergiert

$$x_{k+1} = \Phi(x_k)$$

für $k = 0, 1, 2, \ldots$ gegen den Fixpunkt x^*.

3.) Es gilt die *A–priori–Fehlerabschätzung*

$$\|x_k - x^*\| \leq \frac{L^k}{1-L}\|x_1 - x_0\|. \tag{9.3.6}$$

4.) Weiter gilt die *A–posteriori–Fehlerabschätzung*

$$\|x_k - x^*\| \leq \frac{L}{1-L}\|x_k - x_{k-1}\|. \tag{9.3.7}$$

Beweis: Übung. ∎

Zu diesem wichtigen Theorem aus der Analysis sei noch Folgendes bemerkt:

a) Ein vollständiger normierter Vektorraum heißt BANACH–Raum. Vollständigkeit bedeutet, dass jede CAUCHY–Folge in dem Raum gegen einen Grenzwert in dem Raum konvergiert. Ist dieser Grenzwert eindeutig, so spricht man auch von HAUSDORFF–Räumen. Standardbeispiele für solche Räume sind die reellen Zahlen, also $X = \mathbb{R}$, in denen jedes abgeschlossene Intervall $E = [a, b] \subset \mathbb{R}$ vollständig ist. Dies kann man verallgemeinern in n Dimensionen, womit auch jede abgeschlossene Teilmenge $E \subset \mathbb{R}^n$ vollständig ist. Der BANACH'sche Fixpunktsatz gilt aber auch für vollständige Teilmengen von unendlichdimensionalen Funktionenräume X.

b) Dieser Satz liefert neben der Existenz– und Eindeutigkeitsaussage über die Lösung einer Fixpunktgleichung auch einen *konstruktiven Algorithmus* sowie Fehlerabschätzungen, die darauf schließen lassen, wie lange man iterieren muss, um eine gewisse vorgegebene Genauigkeit zu gewährleisten. Will man etwa die Genauigkeit ε, so genügt es wegen der A–priori–Abschätzung (9.3.6), k so groß zu wählen, dass die Gleichung

$$\frac{L^k}{1-L}\|x_1 - x_0\| \leq \varepsilon,$$

oder nach k umgestellt

$$k \geq \frac{\log\left(\frac{\|x_1 - x_0\|}{\varepsilon(1-L)}\right)}{\log L^{-1}} \tag{9.3.8}$$

erfüllt ist.

Aus dem Banach'schen Fixpunktsatz folgern wir für die von uns meistens benötigte Situation, noch folgendes

Korollar 9.3.9. Seien $X = \mathbb{R}^n$ und $E \subset X$ abgeschlossen und konvex, sowie $\Phi : E \to E$ stetig differenzierbar mit

$$\max_{x \in E} \|\Phi'(x)\| := L < 1.$$

Man beachte, dass in dieser Gleichung $\Phi'(x)$ natürlich für die Jacobi–Matrix

$$\Phi'(x) := \begin{pmatrix} \frac{\partial \Phi_1}{\partial x_1} & \cdots & \frac{\partial \Phi_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial \Phi_n}{\partial x_1} & \cdots & \frac{\partial \Phi_n}{\partial x_n} \end{pmatrix}$$

steht. Dann ist Φ eine Kontraktion auf E, und es gelten die Aussagen 1.) bis 4.) aus dem Banach'schen Fixpunktsatz.

Wir schließen den Abschnitt noch mit dem folgenden

Beispiel 9.3.10. Das System

$$\begin{aligned} 6x &= \cos x + 2y \\ 8y &= xy^2 + \sin x \end{aligned}$$

besitzt auf $E = [0,1]^2$ eine eindeutige Lösung. Bestimme diese approximativ bis auf eine Genauigkeit von 10^{-3} bezüglich der Norm $\|\cdot\|_\infty$. ∎

9.4 Konvergenzordnung und Fehlerschätzung

Wenn man ein Iterationsverfahren benutzt, um eine Nullstelle einer nichtlinearen Gleichung zu ermitteln, so wünscht man natürlich, dass sich die Iterierten x_k schnell dem Grenzwert x^* annähern. Um ein Maß für die Geschwindigkeit eines Iterationsverfahrens zu erhalten, definieren wir die *Konvergenzordnung.*

Definition 9.4.1 (Konvergenzordnung)**.** Eine konvergente Folge $\{x_k\}_{k \in \mathbb{N}}$ in $\mathbb{R}^n$ mit dem Grenzwert x^* hat die *Konvergenzordnung* p, falls für ein $k_0 \in \mathbb{N}$ für alle $k \geq k_0$ die Abschätzung

$$\|x_{k+1} - x^*\| \leq c\|x_k - x^*\|^p$$

gilt, wobei $0 < c < 1$ falls $p = 1$, und $0 < c < \infty$ falls $p > 1$ ist. ∎

Hierbei kann es natürlich einige Iterationen dauern, bis die Zahl k_0 sichtbar wird. Im Falle $p = 1$ spricht man von *linearer Konvergenz.* Allgemein nennt man im Fall $p > 1$ die Konvergenz *superlinear*, wobei man von *quadratischer Konvergenz* spricht, falls $p = 2$ gilt. Offenbar ist die Konvergenz um so schneller, je höher die Konvergenzordnung ist. Als Faustregel gilt, dass sich die Anzahl der korrekten Dezimalstellen bei jeder Iteration ab k_0 ver–p–facht. Dieses Verhalten wollen wir uns konkret anschauen im folgenden Beispiel. Insbesondere stellen wir den Unterschied zwischen linearer und quadratischer Konvergenz zahlenmäßig dar.

Beispiel 9.4.2. Seien $\|x_0 - x^*\| = 0.2 = 2.000\text{e-}01$ und $e_k := \|x_k - x^*\|$. Die folgende Tabelle gibt absolute Fehler für lineare und quadratische Konvergenz für zwei verschiedene c bei steigendem k an:

c	p	k	1	2	3	4	5	6
$c = \frac{1}{2}$	$p = 1$	$e_k \leq$	1.000e-01	5.000e-02	2.500e-02	1.250e-02	6.250e-03	3.125e-03
$c = 3$	$p = 2$	$e_k \leq$	1.213e-01	4.324e-02	5.618e-03	9.432e-05	3.231e-08	2.108e-15

Man erkennt, dass sich bei dem Verfahren der Ordnung 1 der Fehler für steigendes k jeweils halbiert. Bei dem Verfahren der Ordnung 2 quadriert sich ab etwa $k = 4$ jeweils die Potenz.

■

Üblicherweise haben Fixpunktiterationen nur lineare Konvergenzordnungen. Will man Verfahren von höherer Ordnung für skalare Gleichungen konstruieren, kann man wie folgt vorgehen. Nach dem Banach'schen Fixpunktsatz ist die Approximationsgüte umso besser, je kleiner die Lipschitz–Konstante L ist. Im Hinblick auf Korollar 9.3.9, in dem $L = \max_{\xi \in E} |\Phi'(\xi)|$ war, wird L besonders klein in $U(x^*)$, wenn $\Phi'(x^*) = 0$ ist. Allgemeiner gilt folgender

Satz 9.4.3. Für $\Phi : \mathbb{R} \to \mathbb{R}$ mit $\Phi \in \mathcal{C}^{p+1}(U(x^*))$ und $j = 1, \ldots, p-1$ gelte

$$\Phi^{(j)}(x^*) = 0$$

sowie

$$\Phi^{(p)}(x^*) \neq 0.$$

Für $p = 1$ sei zusätzlich $|\Phi'(x^*)| < 1$. Dann konvergiert

$$x_{k+1} = \Phi(x_k)$$

lokal (d.h. für hinreichend kleines $|x_0 - x^*|$) von genau der Ordnung p.

Beweis: Wir betrachten lediglich den Fall $p > 1$. Für $x_k \in U := \overline{U(x^*)}$ gilt mit Taylor–Entwicklung von Φ um x^*

$$\begin{aligned} x_{k+1} &= \Phi(x_k) = \Phi(x^*) + \sum_{j=1}^{p} \frac{(x_k - x^*)^j}{j!} \Phi^{(j)}(x^*) + \frac{(x_k - x^*)^{(p+1)}}{(p+1)!} \Phi^{(p+1)}(\xi_k) \\ &= x^* + \frac{(x_k - x^*)^p}{p!} \Phi^{(p)}(x^*) + \frac{(x_k - x^*)^{(p+1)}}{(p+1)!} \Phi^{(p+1)}(\xi_k), \end{aligned}$$

Dies zieht

$$\frac{|x_{k+1} - x^*|}{|x_k - x^*|^p} \leq \frac{|\Phi^{(p)}(x^*)|}{p!} + \frac{|x_k - x^*|}{(p+1)!} |\Phi^{(p+1)}(\xi_k)| \tag{9.4.4}$$

nach sich. Setze nun $c := \frac{|\Phi^{(p)}(x^*)|}{p!} + \frac{1}{(p+1)!} \max_{y \in U} |\Phi^{(p+1)}(y)|$ und $\tilde{U} := [x^* - \varepsilon, x^* + \varepsilon]$ mit $0 < \varepsilon < 1$ genügend klein, so dass $\tilde{U} \subset U$ und $c\varepsilon^{p-1} < \frac{1}{2}$ gelten. Falls nun $x_k \in \tilde{U}$ ist, folgt mit (9.4.4):

$$\frac{|x_{k+1} - x^*|}{|x_k - x^*|^p} \leq \frac{|\Phi^{(p)}(x^*)|}{p!} + \frac{|x_k - x^*|}{(p+1)!} |\Phi^{(p+1)}(\xi_k)| \leq c.$$

Also gilt

$$\begin{aligned} |x_{k+1} - x^*| &\leq c|x_k - x^*|^p = c|x_k - x^*|^{p-1}|x_k - x^*| \\ &\leq c\varepsilon^{p-1}|x_k - x^*| \end{aligned}$$

$$< \frac{1}{2}|x_k - x^*|. \tag{9.4.5}$$

Wähle nun $x_0 \in \tilde{U}$. Dann ist wegen (9.4.5) $x_k \in \tilde{U}$ für alle k und $\lim_{k\to\infty} x_k = x^*$. Also haben wir die lokale Konvergenz bereits gezeigt. (9.4.5) besagt aber auch

$$|x_{k+1} - x^*| \leq c|x_k - x^*|^p$$

für alle k, was bedeutet, dass die Konvergenz von der Ordnung p ist. ∎

Bevor wir den Abschnitt schließen, wollen wir noch ein paar Konvergenzbegriffe erwähnen. Konvergiert $x_{k+1} = \Phi(x_k)$ für alle $k = 0, 1, 2, \ldots$ nur für Startwerte $x_0 \in U(x^*)$, so heißt das Verfahren *lokal konvergent*. Konvergiert das Verfahren aber für alle x_0 aus dem Definitionsbereich von Φ, so heißt die Konvergenz *global*. Im Allgemeinen sind Verfahren aufgrund der Problemabhängigkeit aber nur lokal konvergent.

9.5 Iterationsmethoden für eine skalare Gleichung

Es sei $f : \mathbb{R} \to \mathbb{R}$ stetig differenzierbar. In diesem Abschnitt beschäftigen wir uns mit dem Lösen des zugehörigen Nullstellenproblems $f(x^*) = 0$.

9.5.1 Bisektionsverfahren

Das *Bisektions– oder Einschließungsverfahren* ist eine vom Prinzip her einfache und robuste Methode zur Lösung der vorgelegten Aufgabe. Es funktioniert nach der folgenden Idee: gilt für $a_0 < b_0$, dass $f(a_0)f(b_0) < 0$ ist, so gilt nach dem Zwischenwertsatz, dass es eine Nullstelle in $[a_0, b_0]$ gibt. Setze nun $x_0 := \frac{1}{2}(a_0 + b_0)$, und berechne $f(x_0)$. Wähle danach ein neues Intervall $[a_1, b_1] := [a_0, x_0]$ oder $[a_1, b_1] := [x_0, b_0]$ so, dass $f(a_1)f(b_1) \leq 0$ gilt.

Hiermit haben wir schon einen

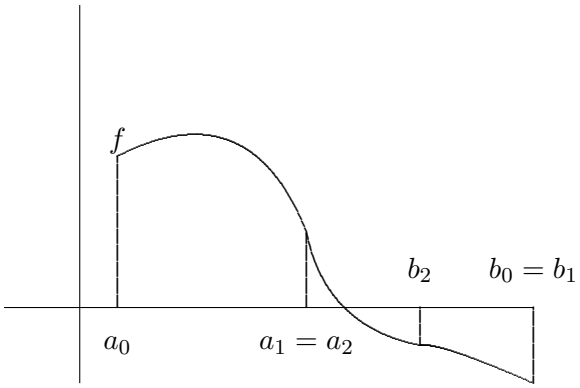
Algorithmus 9.5.1. Für $k = 0, 1, 2, \ldots$ berechne

$$x_k := \frac{1}{2}(a_k + b_k)$$
$$f(x_k)$$

Setze dann

$$\begin{aligned} a_{k+1} &= a_k, b_{k+1} = x_k, \text{falls } f(a_k)f(x_k) \leq 0 \\ a_{k+1} &= x_k, b_{k+1} = b_k, \text{sonst.} \end{aligned}$$

∎



Zur Konvergenzgeschwindigkeit schätzen wir ab

$$|x^* - x_k| \leq \left(\frac{1}{2}\right)^{k+1} |b_0 - a_0|.$$

Also ist die Konvergenz linear. Als Fazit können wir festhalten: Das *Bisektionsverfahren* ist robust, aber dafür auch relativ langsam.

9.5.2 Das NEWTON–Verfahren

Angenommen, es sei $f \in \mathcal{C}^2(U(x^*))$. Wir wollen ein Fixpunktverfahren von quadratischer Ordnung entwickeln. Nach Satz 9.4.3 gilt im Falle $\Phi'(x^*) = 0$, dass $p = 2$ sein muss. Damit setzen wir an

$$\Phi(x) := x - g(x)f(x),$$

was

$$\Phi'(x^*) = 1 - g'(x^*)f(x^*) - g(x^*)f'(x^*) = 1 - g(x^*)f'(x^*)$$

impliziert. Das bedeutet

$$\Phi'(x^*) = 0$$

ist äquivalent zu

$$g(x^*) = \frac{1}{f'(x^*)}.$$

Falls also $f(x^*) = 0$ und $f'(x^*) \neq 0$ ist, so können wir iterieren

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \tag{9.5.2}$$

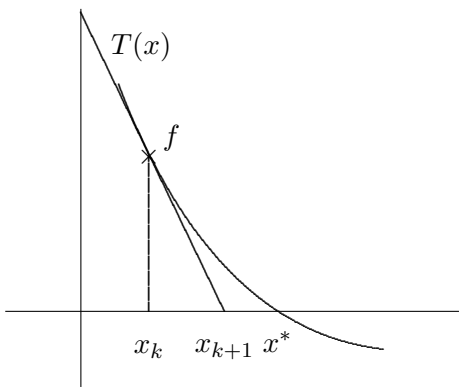
für $k = 0, 1, 2, \ldots$. Dieses Verfahren heißt NEWTON–Verfahren und konvergiert lokal quadratisch. Hierzu bemerken wir, dass dieses Verfahren nicht mehr lokal quadratisch konvergiert, wenn f in x^* eine mehrfache Nullstelle hat. Einen Ausweg hierzu werden wir in Übungsaufgabe 13 sehen.

Wir kommen zur *geometrischen Interpretation* des NEWTON–Verfahrens. Es sei x_k eine bekannte Näherung an x^*. Dann liefert die TAYLOR–Entwicklung

$$f(x) = f(x_k) + (x - x_k)f'(x_k) + \frac{(x - x_k)^2}{2} f''(\xi_k). \tag{9.5.3}$$

Hierbei ist $T(x) := f(x_k) + (x - x_k)f'(x_k)$ die Tangente von f an x_k. Diese ist in einer kleinen Umgebung von x_k eine gute Näherung von f. Dementsprechend ist die Nullstelle von T eine gute Näherung an x^*, und die Nullstelle von $T(x_{k+1})$ liefert

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}.$$



Zur Konvergenz beweisen wir noch folgenden

Satz 9.5.4. Sei $f \in \mathcal{C}^2(U(x^*))$ mit $f(x^*) = 0$ und $f'(x^*) \neq 0$. Dann gilt für $x_k \in U(x^*)$ und das NEWTON–Verfahren aus (9.5.3)

$$x_{k+1} - x^* = \frac{1}{2}\frac{f''(\xi_k)}{f'(x_k)}(x_k - x^*)^2$$

mit $\xi_k \in U(x^*)$. Also ist die Konvergenz lokal quadratisch.

Beweis: Setze $x = x^*$ in (9.5.3). Dann gilt

$$0 = f(x^*) = f(x_k) + (x^* - x_k)f'(x_k) + \frac{(x^* - x_k)^2}{2} f''(\xi_k),$$

was äquivalent ist zu

$$-\frac{f(x_k)}{f'(x_k)} + x_k - x^* = \frac{(x^* - x_k)^2}{2} \frac{f''(\xi_k)}{f'(x_k)},$$

woraus die Behauptung mit $\xi_k \in U(x^*)$ und der Definition von x_{k+1} aus (9.5.1) unmittelbar folgt. ∎

9.5.3 NEWTON–ähnliche Verfahren

Oftmals ist die Berechnung von f' aufwendig, insbesondere dann, wenn f nicht explizit gegeben ist. Deshalb werden wir im Folgenden Verfahren betrachten, die ohne die explizite Berechnung von f' auskommen. Ein Beispiel hierfür haben wir mit dem *Bisektionsverfahren* in 9.5.1 bereits kennengelernt. Dieses war sehr robust, aber leider auch sehr langsam.
Als erstes diskutieren wir nun das *Sekantenverfahren.* Hierbei ersetzt man f' durch die Sekante durch die Punkte $(x_{k-1}, f(x_{k-1}))$ und $(x_k, f(x_k))$, indem man $S(x)$ definiert durch

$$S(x) := \frac{x - x_{k-1}}{x_k - x_{k-1}} f(x_k) + \frac{x_k - x}{x_k - x_{k-1}} f(x_{k-1}),$$

und die neue Näherung x_{k+1} durch

$$S(x_{k+1}) = 0$$

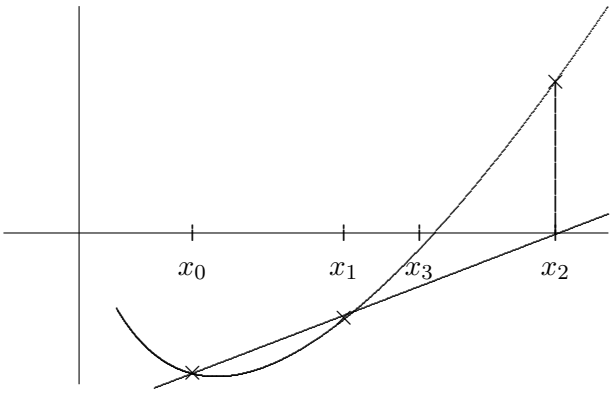
d.h.

$$\begin{aligned} x_{k+1} &= \frac{x_{k-1}f(x_k) - x_k f(x_{k-1})}{f(x_k) - f(x_{k-1})} \\ &= x_k - f(x_k)\left(\frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})}\right) \end{aligned} \tag{9.5.5}$$

bestimmt. Dieses Verfahren konvergiert lokal etwa von der Ordnung

$$p \approx 1.6.$$

Geometrisch bedeutet dies, dass man die Sekante durch die jeweiligen Punkte P_k und P_{k+1} an der Stelle $S = 0$ auswertet, und dann $f(S = 0)$ als neuen Punkt P_{k+2} nimmt, um die Sekante durch P_{k+1} und P_{k+2} zu berechnen, usw.. Als wichtigste *Eigenschaften des Sekantenverfahrens* halten wir die schnellere Konvergenz im Vergleich zum Bisektionsverfahren fest. Trotzdem ist es langsamer als das NEWTON–Verfahren und benötigt außerdem *zwei Startwerte.* Dafür kommt es aber ohne Ableitungen aus, und pro Schritt muss man nur eine Funktionsauswertung machen.



Eine weitere, bisher noch nicht in Betracht gezogene Möglichkeit ist die Kombination aus Bisektions– und Sekantenverfahren, die sogenannte *Regula falsi.* Wie bei der Bisektion macht man hier die Annahme, dass zwei Werte a_0, b_0 existieren mit der Eigenschaft

$$f(a_0)f(b_0) < 0.$$

Bei der Bisektion wurden neue Werte in der Mitte des Intervalls durch $x_0 = \frac{1}{2}(a_0+b_0)$ bestimmt, und dann $f(x_0)$ berechnet. Hier bestimmt man den neuen Wert x_0 durch die Sekante durch a_0, b_0 als

$$x_0 = \frac{a_0 f(b_0) - b_0 f(a_0)}{f(b_0) - f(a_0)}.$$

Danach wählt man das neue Intervall als $[a_1, b_1] = [a_0, x_0]$ oder $[a_1, b_1] = [x_0, b_0]$ so, dass $f(a_1)f(b_1) \leq 0$ gilt. Der zugehörige Algorithmus lautet wie folgt:

Algorithmus 9.5.6. Gegeben seien a_0, b_0 mit $f(a_0)f(b_0) < 0$. Für $k = 0, 1, 2, \ldots$ berechne

$$x_k = \frac{a_k f(b_k) - b_k f(a_k)}{f(b_k) - f(a_k)}$$

sowie

$$f(x_k).$$

Falls $f(x_k)f(a_k) \leq 0$ setze

$$a_{k+1} = a_k, \quad b_{k+1} = x_k.$$

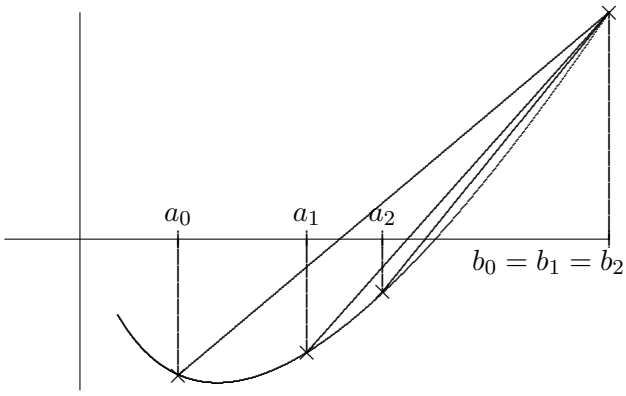
Sonst setze

$$a_{k+1} = x_k, \quad b_{+1} = b_k.$$

■

Hieraus ergibt sich, dass die Nullstelle x^* für alle k in (a_k, b_k) liegt und $a_k \to x^*$ oder $b_k \to x^*$ für $k \to \infty$ geht. Da $x^* \in (a_k, b_k)$, ist diese Methode sehr zuverlässig, und die Konvergenz ist im Allgemeinen schneller als bei Bisektion, hat aber dieselbe Konvergenzordnung $p = 1$. Die zugehörige Geometrie zeigt die folgende Skizze:

Zusammenfassend noch folgende Hinweise:



- Häufig benutzt man zunächst ein robustes Verfahren wie Bisektion und danach zur Beschleunigung das NEWTON–Verfahren.
- Mit der Problemstellung ist meist ein f vorgegeben, für das das Problem $f(x) = 0$ zu lösen ist. Ein Lösungsansatz ist eine *Fixpunktiteration* mit geeignet konstruierter Iterationsfunktion Φ. Diese Methode hat im Allgemeinen die Konvergenzordnung 1.
- Zur Beschleunigung versucht man generell Verfahren höherer Ordnung zu konstruieren, z.B. das NEWTON–Verfahren. Diese sind im Allgemeinen aber nur *lokal* konvergent, d.h. der Erfolg der Iteration hängt wesentlich von der Wahl eines geeigneten Startwertes ab.
- Bisektion und Regula falsi sind sehr zuverlässige Verfahren.
- Das Sekantenverfahren ist eine effiziente Variante des NEWTON–Verfahrens, in der die Berechnung von f' vermieden wird.
- Zur Nullstellenberechnung von Polynomen sollte man das HORNER–Schema verwenden, siehe dazu Übungsaufgabe 15.

9.6 Das Newton–Verfahren für Systeme

Nun diskutieren wir das NEWTON–Verfahren zur Lösung von $f(x) = 0$, wobei $f : \mathbb{R}^n \to \mathbb{R}^n$ zweimal stetig differenzierbar sei und $n > 1$, $f(x)$ also die Form

$$f(x) := \begin{pmatrix} f_1(x_1, \ldots, x_n) \\ \vdots \\ f_n(x_1, \ldots, x_n) \end{pmatrix}$$

hat.

9.6.1 Herleitung und Grundlagen

In diesem Kapitel bezeichnen wir den k-ten Iterationsvektor mit

$$x^k = \begin{pmatrix} x_1^k \\ \vdots \\ x_n^k \end{pmatrix} \in \mathbb{R}^n.$$

In Gleichung (9.5.3) haben wir die TAYLOR–Entwicklung für eine skalare Gleichung betrachtet. Für eine Komponente $f_i : \mathbb{R}^n \to \mathbb{R}$ kann man dies auch schreiben als

$$f_i(x) = f_i(x^k) + \sum_{j=1}^{n} (x_j - x_j^k) \frac{\partial f_i}{\partial x_j}(x^k) + O(\|x - x^k\|_2^2).$$

Bezeichnet man die JACOBI–Matrix von f mit

$$Df(x) = \begin{pmatrix} \frac{\partial f_1(x)}{\partial x_1} & \cdots & \frac{\partial f_1(x)}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n(x)}{\partial x_1} & \cdots & \frac{\partial f_n(x)}{\partial x_n} \end{pmatrix},$$

dann lässt sich dies mit TAYLOR–Entwicklung zusammenfassen als

$$f(x) = f(x^k) + Df(x^k)(x - x^k) + O(\|x - x^k\|_2^2). \tag{9.6.1}$$

Approximiert man nun die Nullstelle von f durch die Nullstelle x^{k+1} der linearen Approximation von f in x^k, also das 1. TAYLOR–Polynom durch

$$0 = f(x^k) + Df(x^k)(x^{k+1} - x^k),$$

dann folgt im Falle $\det Df(x^k) \neq 0$

$$x^{k+1} = x^k - (Df(x^k))^{-1} f(x^k). \tag{9.6.2}$$

Dies ist das vektorwertige Analogon zu (9.5.3). Numerisch wird $(Df(x^k))^{-1}$ *nie* explizit berechnet. Der Ausdruck $(Df(x^k))^{-1} f(x^k)$ steht für die Lösung s^k eines linearen Gleichungssystems

$$Df(x^k) s^k = -f(x^k),$$

was man aus (9.6.2) mittels $s^k = x^{k+1} - x^k$ herleitet. Wir erhalten den folgenden Algorithmus:

Algorithmus 9.6.3 (NEWTON–Verfahren für Systeme)**.** Für $k = 0, 1, 2, \ldots$ berechne $f(x^k)$ und $Df(x^k)$. Löse für s^k das lineare Gleichungssystem

$$Df(x^k)s^k = -f(x^k) \tag{9.6.4}$$

und setze

$$x^{k+1} := x^k + s^k. \tag{9.6.5}$$

■

Den letzten Schritt (9.6.5) bezeichnet man auch als NEWTON–Korrektur. Je nach Struktur von f verwendet man zur Lösung von (9.6.4) ein QR–Verfahren oder iteratives Verfahren. Zur Wahl des Startwerts und der Anzahl der Iterationen kommen wir später. Wir werden nun nochmals Beispiel 9.3.10 vertiefen:

Beispiel 9.6.6. Man löse die System

$$\begin{aligned} f_1(x_1, x_2) &:= 6x_1 - \cos x_1 - 2x_2 = 0 \\ f_2(x_1, x_2) &:= 8x_2 - x_1 x_2^2 - \sin x_1 = 0. \end{aligned}$$

Hierzu berechnen wir

$$Df(x) = \begin{pmatrix} 6 + \sin x_1 & -2 \\ -x_2^2 - \cos x_1 & 8 - 2x_1x_2 \end{pmatrix}.$$

Wähle nun als Startwert $x^0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$. Dann gilt

$$f(x^0) = \begin{pmatrix} -1 \\ 0 \end{pmatrix} \quad \text{und} \quad Df(x^0) = \begin{pmatrix} 6 & -2 \\ -1 & 8 \end{pmatrix}.$$

Nun lösen wir das System

$$\begin{pmatrix} 6 & -2 \\ -1 & 8 \end{pmatrix} s^0 = -\begin{pmatrix} -1 \\ 0 \end{pmatrix}.$$

Als dessen Lösung erhalten wir $s^0 = \frac{1}{46}\begin{pmatrix} 8 \\ 1 \end{pmatrix}$ und damit $x^1 = x^0 + s^0 = \frac{1}{46}\begin{pmatrix} 8 \\ 1 \end{pmatrix}$ usw. ■

Man erwartet wie im skalaren Fall (lokal) *quadratische* Konvergenz. Dies erfordert eine Reihe von Voraussetzungen, die in der Praxis nur sehr schwer zu überprüfen sind. Zur Veranschaulichung des Typs von Voraussetzungen stellen wir hier eine einfache Variante eines Konvergenzsatzes vor. Dazu benötigen wir die Definition einer *konvexen* Menge Ω in $\mathbb{R}^n$. Eine Menge $\Omega \in \mathbb{R}^n$ heißt konvex, wenn für alle $x, y \in \Omega$ auch

$$[x, y] := \{tx + (1-t)y : 0 \le t \le 1\} \subseteq \Omega$$

ist. Diesen und weitere Sätze zur Konvergenz findet man in [S].

Satz 9.6.7. Sei $\Omega \in \mathbb{R}^n$ offen und konvex. Für ein $x^0 \in \Omega$ gebe es Konstanten $r, \alpha, \beta, \gamma, h > 0$ mit

$$B_r(x^0) := \left\{x \in \mathbb{R}^n : \|x - x^0\| < r\right\} \subseteq \Omega \tag{9.6.8}$$

$$h := \frac{\alpha\beta\gamma}{2} < 1 \tag{9.6.9}$$

$$r := \frac{\alpha}{1-h}. \tag{9.6.10}$$

Weiter sei die Funktion $f : \Omega \to \mathbb{R}^n$ stetig differenzierbar für alle $x \in \Omega$ und habe die Eigenschaften

(a) Die JACOBI–Matrix $Df(x)$ ist LIPSCHITZ–stetig mit γ, also

$$\|Df(x) - Df(y)\| \leq \gamma \|x - y\|$$

für alle $x, y \in \Omega$.

(b) $(Df(x))^{-1}$ existiert und erfüllt

$$\|(Df(x))^{-1}\| \leq \beta$$

für alle $x \in \Omega$.

(c) Es gilt

$$\|(Df(x^0))^{-1} f(x^0)\| \leq \alpha.$$

Dann gilt:

(A) Ausgehend von x^0 ist jedes x^{k+1}, definiert durch (9.6.2),

$$x^{k+1} := x^k - (Df(x^k))^{-1} f(x^k),$$

für alle $k = 0, 1, 2, \ldots$ wohldefiniert, und $x^k \in B_r(x^0)$ für alle $k \geq 0$.

(B)

$$\lim_{k \to \infty} x^k = x^*$$

existiert und $x^* \in \overline{B_r(x_0)}$ und $f(x^*) = 0$.

(C) Für alle $k \geq 0$ gilt die Fehlerabschätzung

$$\|x^k - x^*\| \leq \alpha \frac{h^{2^k - 1}}{1 - h^{2^k}}.$$

Wegen $0 < h < 1$ ist das NEWTON–Verfahren also mindestens *quadratisch* konvergent. ∎

Bemerkung 9.6.11. (i) Falls Ω beschränkt und $f \in \mathcal{C}^2(\overline{\Omega}, \mathbb{R}^n)$ ist, dann ist Df LIPSCHITZ–stetig, d.h. (a) gilt für ein γ.

(ii) Die Parameter β und γ sind durch das Problem gegeben. Damit also (9.6.9) gilt, muss demnach α genügend klein sein. Wegen (c) beeinflusst dies die Wahl des Startwertes x^0, d.h. x^0 muss schon nahe genug an x^* liegen, denn dann ist $\|f(x^0)\|$ klein.

∎

Bevor wir zum Beweis von Satz 9.6.7 kommen, brauchen wir noch folgendes

Lemma 9.6.12. Sei $\Omega \subseteq \mathbb{R}^n$ konvex. Für alle $x \in \Omega$ existiere $Df(x)$, und es gebe eine Konstante $0 \leq \gamma < \infty$ mit

$$\|Df(x) - Df(y)\| \leq \gamma \|x - y\| \tag{9.6.13}$$

für alle $x, y \in \Omega$. Dann gilt

$$\|f(x) - f(y) - Df(y)(x - y)\| \leq \frac{\gamma}{2} \|x - y\|^2. \tag{9.6.14}$$

Beweis: Definiere für beliebige $x, y \in \Omega$ die Funktion $\Phi : [0,1] \to \mathbb{R}^n$ durch

$$\Phi(t) := f(y + t(x-y)).$$

Da f auf Ω differenzierbar ist, ist auch Φ auf $[0,1]$ differenzierbar mit Ableitung

$$\Phi'(t) = Df(y + t(x-y))(x-y).$$

Für jedes $t \in [0,1]$ folgt daher

$$\begin{aligned} \|\Phi'(t) - \Phi'(0)\| &= \|(Df(y+t(x-y)) - Df(y))(x-y)\| \\ &\leq \|Df(y+t(x-y)) - Df(y)\| \cdot \|x-y\|. \end{aligned}$$

Aufgrund von Voraussetzung (9.6.13) können wir dies nach oben durch

$$\gamma \|y + t(x-y) - y\| \cdot \|x-y\| = \gamma t \|x-y\|^2 \tag{9.6.15}$$

abschätzen. Nach Definition von Φ gilt nun

$$\begin{aligned} f(x) - f(y) - Df(y)(x-y) &= \Phi(1) - \Phi(0) - \Phi'(0) \\ &= \int_0^1 \Phi'(t)dt - \Phi'(0) \\ &= \int_0^1 (\Phi'(t) - \Phi'(0))dt. \end{aligned}$$

Nun gilt mit (9.6.15)

$$\begin{aligned} \|f(x) - f(y) - Df(y)(x-y)\| &\leq \int_0^1 \|\Phi'(t) - \Phi'(0)\|dt \\ &\leq \gamma \|x-y\|^2 \int_0^1 tdt \\ &= \frac{\gamma}{2}\|x-y\|^2 \end{aligned}$$

■

Beweis von Satz 9.6.7: Wir starten mit (A) und führen eine vollständige Induktion nach k durch. Nach Voraussetzung (b) existiert $(Df(x))^{-1}$ für alle $x \in \Omega$. Daher ist die Gleichung

$$x^{k+1} = x^k - (Df(x))^{-1} f(x^k),$$

die wir bereits in (9.6.2) hergleitet haben, wohldefiniert. Wir müssen noch zeigen, dass $x^k \in B_r(x^0)$ für alle k gilt. Für $k = 0$ ist dies bereits per Definition so. Sei $k = 1$. Dann gilt:

$$x^1 = x^0 - (Df(x^0))^{-1} f(x^0) \Leftrightarrow x^1 - x^0 = -(Df(x^0))^{-1} f(x^0).$$

Hieraus folgt

$$\|x^1 - x^0\| = \|(Df(x^0))^{-1} f(x^0)\| \leq \alpha \tag{9.6.16}$$

nach Voraussetzung (c). Weiter gilt unter Ausnutzung von (9.6.10)

$$\|x^1 - x^0\| < r.$$

Also ist $x^1 \in B_r(x^0)$. Sei nun $x^j \in B_r(x^0)$ für alle $j = 0, 1, \ldots, k$, mit $k \geq 1$. Wir müssen zeigen, dass auch $x^{k+1} \in B_r(x^0)$ ist. Mit (9.6.2) können wir abschätzen

$$\|x^{k+1} - x^k\| = \|(Df(x^k))^{-1} f(x^k)\| \leq \|(Df(x^k))^{-1}\| \cdot \|f(x^k)\|.$$

Nach Voraussetzung (b) gilt aber

$$\|x^{k+1} - x^k\| \leq \beta \|f(x^k)\| \tag{$*$}$$

Weiter ist die Definition von x^k in (9.6.2) äquivalent zu

$$Df(x^{k-1})(x^k - x^{k-1}) + f(x^{k-1}) = 0,$$

was eingesetzt in $(*)$ und nach Lemma 9.6.12

$$\|x^{k+1} - x^k\| \leq \beta \|f(x^k) - f(x^{k-1}) - Df(x^{k-1})(x^k - x^{k-1})\| \leq \beta \frac{\gamma}{2} \|x^k - x^{k-1}\|^2 \tag{9.6.17}$$

ergibt. Mit Gleichung (9.6.17) zeigen wir durch vollständige Induktion, dass

$$\|x^{k+1} - x^k\| \leq \alpha h^{2^k - 1} \tag{9.6.18}$$

ist. Der Induktionsanfang $k = 0$ ist klar mit (9.6.16). Zum Induktionsschritt von $k \to k+1$ schätzen wir ab:

$$\begin{aligned} \|x^{k+1} - x^k\| &\leq \frac{\beta\gamma}{2} \|x^k - x^{k-1}\|^2 \\ &\leq \frac{\beta\gamma}{2} \alpha^2 (h^{2^{k-1}-1})^2 \\ &= \frac{\beta\gamma\alpha}{2} \alpha (h^{2^k - 2}) = \alpha h^{2^k - 1}, \end{aligned}$$

woraus (9.6.18) folgt. Aus dieser Gleichung folgt nun aber

$$\begin{aligned} \|x^{k+1} - x^0\| &\leq \|x^{k+1} - x^k\| + \|x^k - x^{k-1}\| + \ldots + \|x^1 - x^0\| \\ &\leq \alpha (h^{2^k - 1} + h^{2^{k-1}-1} + \ldots + h^7 + h^3 + h + 1) \\ &< \alpha \sum_{i=0}^{\infty} h^i = \frac{\alpha}{1-h} = r. \end{aligned}$$

Also ist $x^{k+1} \in B_r(x^0)$ und damit Behauptung (A) bewiesen.
Nun wird (B) gezeigt: Aus der Abschätzung (9.6.18) folgt, dass $\{x^k\}_{k \in \mathbb{N}_0}$ eine Cauchy–Folge ist, denn für $m \geq n$ gilt

$$\begin{aligned} \|x^{m+1} - x^n\| &\leq \|x^{m+1} - x^m\| + \ldots + \|x^{n+1} - x^n\| \\ &\leq \alpha (h^{2^n - 1} + h^{2^{n+1}-1} + \ldots) \\ &= \alpha h^{2^n - 1} (1 + h^{2^n} + (h^{2^n})^2 + \ldots). \end{aligned}$$

Unter Ausnutzung der geometrischen Reihe erhalten wir

$$\|x^{m+1} - x^n\| < \frac{\alpha h^{2^n - 1}}{1 - h^{2^n}} < \varepsilon \tag{9.6.19}$$

für genügend großes $n \geq N(\varepsilon)$, da $h < 1$ ist. Daher existiert der Grenzwert

$$\lim_{k \to \infty} x^k = x^*, \tag{9.6.20}$$

und $x^* \in \overline{B_r(x^0)}$, da nach (A) alle $x^k \in B_r(x^0)$ sind. Den zweiten Teil von (B) ziehen wir gleich nach. Wir schieben hier kurz den Beweis von Behauptung (C) ein: Aus (9.6.19) folgt nämlich auch

$$\lim_{m \to \infty} \|x^{m+1} - x^n\| = \|x^* - x^n\| < \frac{\alpha h^{2^n - 1}}{1 - h^{2^n}},$$

was die Fehlerabschätzung (C) impliziert. Weiter mit (B): Wir müssen noch zeigen, dass der Grenzwert x^* in (9.6.20) Nullstelle von f in $\overline{B_r(x^0)}$ ist. Wegen Voraussetzung (a) gilt für alle $x^k \in B_r(x^0)$:

$$\|Df(x^k) - Df(x^0)\| \leq \gamma \|x^k - x^0\| < \gamma r,$$

und daher ist

$$\begin{aligned} \|Df(x^k)\| &\leq \|Df(x^k) - Df(x^0)\| + \|Df(x^0)\| \\ &\leq \gamma r + \|Df(x^0)\| =: K \end{aligned}$$

Aus (9.6.2) folgt nun

$$\|f(x^k)\| \leq \|Df(x^k)\| \cdot \|x^{k+1} - x^k\| \leq K\|x^{k+1} - x^k\|.$$

Also gilt

$$\begin{aligned} \lim_{k\to\infty} \|f(x^k)\| &\leq K \lim_{k\to\infty} \|x^{k+1} - x^k\| \\ &\leq K \lim_{k\to\infty} \alpha h^{2^k - 1} = 0, \end{aligned}$$

und, da f an $x^* \in \Omega$ stetig ist, auch

$$\lim_{k\to\infty} \|f(x^k)\| = \|f(x^*)\| = 0,$$

also $f(x^*) = 0$, was bedeutet, dass x^* Nullstelle von f ist, und den Beweis beendet. ∎

Unter ähnlichen Voraussetzungen kann man auch die Eindeutigkeit von x^* in $B_r(x^0)$ zeigen. Dies geschieht für das Newton–Verfahren in Banach–Räumen in folgendem Satz von Newton–Kantorovich von 1948:

Es gelten die Voraussetzungen (a), (c) aus Satz 9.6.7 und anstelle von (b) gelte

(b') $\|(Df(x^0))^{-1}\| \leq \beta$.

Da das nur für den Startwert gilt, ist diese Aussage schwächer. Definiere $h := \alpha\beta\gamma$ und

$$r_{1/2} := \frac{1 \pm \sqrt{1 - 2h}}{h}\alpha.$$

Falls $h \leq \frac{1}{2}$ und $\overline{B_{r_1}(x^0)} \subset \Omega$, dann gilt: Die Folge $\{x^k\}_{k\in\mathbb{N}}$, definiert durch (9.6.2) bleibt in $B_{r_1}(x^0)$ und konvergiert gegen die einzige Nullstelle von f in $\Omega \cap B_{r_2}(x^0)$. Einen Beweis hierzu findet man z.B. in [OR], Theorem 12.6.2, p. 421.

9.6.2 Hinweise zur praktischen Durchführung des Newton–Verfahrens

Oft ist es in der Praxis sehr aufwendig, für jede Iterierte k die Ableitung $Df(x^k)$ zu berechnen. Wenn sich diese Matrix im Laufe der Iteration nur wenig verändert, kann man folgenden Algorithmus anwenden:

Algorithmus 9.6.21 (Vereinfachtes Newton–Verfahren)**.** Berechne $Df(x^0) =: A$. Für $k = 0, 1, 2, \ldots$ löse für s^k:

$$As^k = -f(x^k), \tag{9.6.22}$$

und setze dann

$$x^{k+1} := x^k + s^k.$$

∎

Dies hat allerdings den entscheidenden Nachteil, dass man die quadratische Konvergenz verliert. Daher verwendet man in der Praxis eine Mischform, indem man etwa alle drei bis fünf Schritte die Matrix A erneuert. Dies ist vor allem bei schwachen Nichtlinearitäten ein geeignetes Verfahren.

Auswertung der JACOBI–Matrix

Oft sind die Einträge von Df nicht oder nur mit großem Aufwand geschlossen berechenbar. In solchen Fällen wendet man eine numerische Differentiation an, was bedeutet, dass man die Ableitungen durch Differenzenquotienten ersetzt, also

$$\frac{\partial f_i}{\partial x_j}(x) \approx \frac{f_i(x + he^j) - f_i(x)}{h}.$$

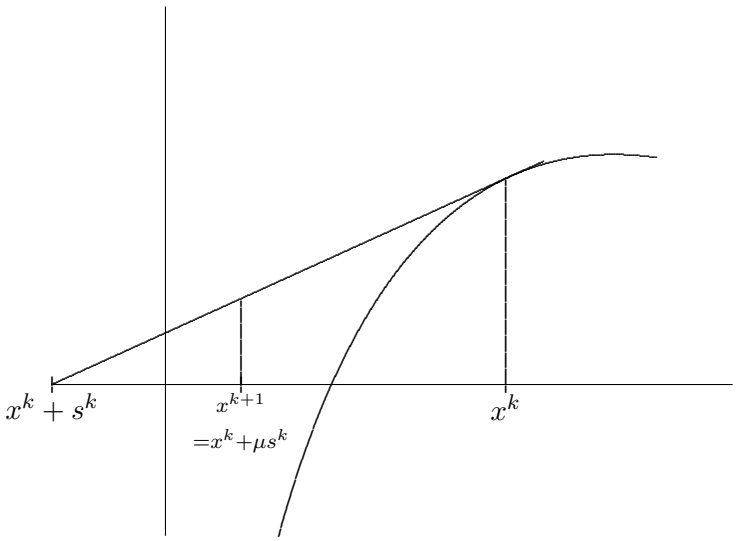
Hierbei hängt die für h anzusetzende Größe vom Problem ab. Bei zu großer Wahl von h, ist die Approximation von Df zu ungenau, was eine langsame Konvergenz zur Folge hat. Wählt man h zu klein, so besteht die Gefahr von Auslöschungen.

Wahl des Startwertes

Hierbei sollte man hauptsächlich Hintergrundinformationen verwenden. Ist dies nicht möglich, so wendet man das *Homotopieverfahren* an, bei dem man versucht, anstelle von $f(x) = 0$ die Gleichung $f(x, \lambda) = 0$ zu lösen, in der λ für einen Parameter steht, und das Problem dadurch, etwa für $\lambda = 1$, auf ein lineares Problem zurückgeführt wird. Diese Lösung wendet man dann als Startwert für kompliziertere Probleme. Öfters findet man auch in der ursprünglichen Gleichung bereits Parameter, die als solche Homotopieparameter verwendet werden können. Dieses Vorgehen hat allerdings den Nachteil, dass es theoretisch nicht abgesichert ist. Eine Idee, um hier Abhilfe zu schaffen, ist, einen Algorithmus zu finden, der sein Konvergenzverhalten *selbst* beeinflusst. Dies geschieht mit folgendem Grundgedanken: Der Korrekturschritt mit $s^k = -(Df(x))^{-1}f(x^k)$ des klassischen NEWTON–Verfahrens liefert die Richtung, in der f abnimmt. Oft ist es günstiger, nur einen Teil des Schritts in diese Richtung zu gehen. Man setze etwa

$$x^{k+1} := x^k + \mu s^k \tag{9.6.23}$$

mit passendem $\mu \in (0, 1]$. Dies lässt sich als eine Art *Dämpfung* interpretieren. Graphisch könnte das etwa so aussehen:



Dies wirft natürlich die Frage nach der Wahl von μ auf. Unser Ziel ist es, $f(x) = 0$ als Maß für die Abweichung von $\|f(x^k)\|$ mit einer geeigneten Norm $\|\cdot\|$ zu betrachten. Daher versuchen wir, μ in jedem Schritt so zu wählen, dass

$$\|f(x^0)\| > \|f(x^1)\| > \|f(x^2)\| > \dots \tag{9.6.24}$$

gilt. Zusätzlich ist zu beachten, dass die Lösung x^* von $f(x) = 0$ auch Lösung von $Bf(x) = 0$ für jedes nichtsinguläre $B \in \mathbb{R}^{n\times n}$ ist. Dies lässt sich als eine Umskalierung interpretieren, denn die NEWTON–Iteration ist unabhängig von B. Man spricht auch davon, dass das Problem *affin-invariant* ist, siehe [DH]. Dies macht man sich etwa durch den Monotonietest

$$\|(Df(x^k))^{-1}f(x^{k+1})\|_2 < \|(Df(x^k))^{-1}f(x^k)\|_2 \tag{9.6.25}$$

klar. Denn wenn man in (9.6.25) anstelle von f die umskalierte Funktion Bf verwendet, kürzt sich diese durch die Bildung der Inversen wieder heraus. Weiter ist zu bemerken, dass die rechte Seite in (9.6.25) in der NEWTON–Iteration bereits vorhanden ist. Die linke Seite erfordert noch die Lösung von

$$Df(x^k)v^k = -f(x^{k+1}).$$

Dies ist nicht zu viel zusätzlicher Aufwand, wenn man die QR– oder LR–Zerlegung von $Df(x^k)$ bereits kennt. Eine geeignete Norm für $\|\!| \cdot |\!\|$ ist dann

$$\|\!|f(x^{k+1})|\!\| := \|(Df(x^k))^{-1}f(x^{k+1})\|_2. \tag{9.6.26}$$

Damit haben wir

Algorithmus 9.6.27 (Gedämpftes NEWTON–Verfahren)**.** Setze $k = 0$.

1.) berechne $s = s^k$ über (9.6.4) $Df(x^k)s^k = -f(x^k)$ und setze $\mu = 1$.

2.) Dämpfungsschritt: setze $x = x^k + \mu s^k$ und prüfe, ob

$$\|\!|f(x)|\!\| \le (1 - \frac{\mu}{4})\|\!|f(x^k)|\!\|. \tag{9.6.28}$$

Falls ja, setze

$$x^{k+1} = x^k + \mu s^k$$

und gehe zu 3.).
sonst ersetze μ durch $\frac{\mu}{2}$. Falls noch $\mu > 2^{-10}$, gehe zum Anfang von 2.), sonst Abbruch.

3.) neuer Iterationsschritt: Ist $k >$ vorgegebene Schrittzahl oder $\|\!|f(x^{k+1})|\!\| <$ tol, dann ist das Ende erreicht. Sonst setze $k = k + 1$ und gehe zu 1.) zurück.

Abschließend noch folgende Bemerkungen

a) Dieses Verfahren enthält zwei ineinander geschachtelte Schleifen, eine äußere über k wie beim klassischen NEWTON–Verfahren, und eine innere über $\mu = 1, \frac{1}{2}, \frac{1}{4}, \ldots$. Bei der Implementierung ist es daher wichtig, beide Schleifen zu begrenzen.

b) Es reicht wegen der Rundungsfehler nicht, statt (9.6.28) nur

$$\|\!|f(x^k + \mu s^k)|\!\| < \|\!|f(x^k)|\!\|$$

zu testen.

10 Nichtlineare Ausgleichsrechnung

Nun betrachten wir das Modell, einen von $x_1, \ldots, x_n$ abhängigen funktionalen Zusammenhang

$$b(t) = \varphi(t, x_1, \ldots, x_n)$$

aus den Daten (t_i, b_i) für $i = 1, \ldots, m$ zurückzugewinnen, wobei $m \geq n$ und $b_i \approx b(t_i)$ gilt. Eine Lösungsmethode hierbei ist die GAUSS'sche Fehlerquadratmethode, bei der man die Parameter x_i so bestimmt, dass gilt

$$\sum_{i=1}^{m} (b_i - \varphi(t_i, x_1, \ldots, x_n))^2 = \text{ min.}$$

Bei der linearen Ausgleichsrechnung hing die Modellfunktion φ stets *linear* von den x_i ab, womit wir diese Abhängigkeit auch als

$$\varphi(t_i, x_1, \ldots, x_n) = \sum_{j=1}^{n} a_{ij} x_j$$

schreiben konnten. Damit war die Lösung durch

$$\min_{x \in \mathbb{R}^n} \|Ax - b\|_2^2$$

mit $b \in \mathbb{R}^n$ gegeben.

Jetzt wollen wir Probleme betrachten, in denen φ nichtlinear von den x_i abhängt. Dies wirft natürlich die Frage auf, wie sich ein solches Problem modellieren lässt. Mit $F(x) = F(x_1, \ldots, x_n)$ und $F_i(x_1, \ldots, x_n) := b_i - \varphi(t_i, x_1, \ldots, x_n)$ lässt sich das Problem

$$\sum_{i=1}^{m} (b_i - \varphi(t_i, x_1, \ldots, x_n))^2 = \text{ min.}$$

in unseren Fall als

$$\|F(x)\|_2^2 = \min_{x \in \mathbb{R}^n} \tag{10.1}$$

schreiben. Nun stellen wir uns folgende Aufgabe: Seien $F \in \mathcal{C}^2(\Omega, \mathbb{R}^m)$ und $\Omega \subset \mathbb{R}^n$ offen. Setze $F(x) := f(x) - b$, wobei $f \in \mathcal{C}^2(\Omega, \mathbb{R}^m)$ und $b \in \mathbb{R}^m$ seien. Finde $x^* \in \Omega$ mit der Eigenschaft

$$\|F(x^*)\|_2 = \inf_{x \in \Omega} \|F(x)\|_2. \tag{10.2}$$

Die Lösungsstrategie hierbei ist: Rückführung auf lineare Ausgleichsprobleme. Sei dazu x^k eine „gute" Startnäherung. Die TAYLOR–Entwicklung von F um x^k liefert

$$F(x) \approx F(x^k) + Df(x^k)(x - x^k).$$

Ersetze nun F in Gleichung (10.2) durch eine lineare Approximation. Finde dazu ein $s^k \in \Omega$, so dass

$$\|F(x^k) + Df(x^k)s^k\|_2 = \min_{x \in \Omega} \|F(x^k) + Df(x^k)x\|_2 \tag{10.3}$$

gilt und setze

$$x^{k+1} := x^k + s^k. \tag{10.4}$$

Hiermit erhalten wir bereits

Algorithmus 10.5 (Gauss–Newton–Verfahren)**.**

1.) Wähle Startwert x^0 und Toleranz ε.

2.) Für $k = 0, 1, 2, \ldots$
berechne $f(x^k), Df(x^k)$.

3.) Ist $\|(Df(x^k))^T F(x^k)\|_2 \leq \varepsilon$, stop.

4.) Löse das lineare Ausgleichsproblem (10.3), bevorzugt mittels QR–Zerlegung von A.

5.) Setze $x^{k+1} = x^k + s^k$ und gehe zu 2.) zurück.

■

Zu diesem Algorithmus sei noch angemerkt:

- Das Abbruchkriterium in Schritt 3.) bedeutet, dass man ein lokales Minimum gefunden haben könnte. Dieser kritische Punkt könnte aber trotzdem auch nur ein Sattelpunkt sein.
- Dämpfungsstrategien können wie in Algorithmus 9.6.27 verwendet werden.
- Konvergenzsätze für dieses Verfahren sind aus Satz 9.6.7 ableitbar.
- Man hat im Allgemeinen nur lineare Konvergenz.
- Ein alternatives Verfahren ist das Levenberg–Marquardt–Verfahren, welches das Problem eines weit vom Minimum entfernten Iterationswertes x^k aufgreift. Hier ist die Linearisierung s^k sicherlich nicht mehr akzeptabel. In einem solchen Fall löst man das Ersatzproblem
$$\min_{s \in \mathbb{R}^n} \|F(x^k) + Df(x^k)s\|_2$$
unter der Nebenbedingung
$$\|s\|_2 \leq \delta,$$
für ein geeignetes δ. Mittels der Lagrange'schen Multiplikatorenregel (Penalty Term) wird dies mit $\lambda > 0$ zu
$$\min_{s \in \mathbb{R}^n} \|F(x^k) + Df(x^k)s\|_2^2 + \lambda \|x\|_2^2.$$
Aus der notwendigen Bedingung für ein lokales Minimum leiten wir somit die Normalengleichung
$$(DF(x^k)^T DF(x^k) + \lambda I)s^k = -DF(x^k)^T F(x^k)$$
ab.
Dieses Verfahren ist viel stabiler als das Gauß–Newton–Verfahren, da die Matrix garantiert nichtsingulär ist; die Lösung wird hier ebenfalls am Besten über QR–Zerlegung berechnet. Allerdings ergibt sich hier das Problem, wie das λ am Geschicktesten gewählt werden soll. Wer sich für weitere Details zum Thema dieses Kapitels interessiert, sei auf [H] und [DH] verwiesen.

11 Iterative Lösung linearer Gleichungssysteme

11.1 Motivation und Einführung

Zur Lösung des Systems

$$Ax = b$$

mit nichtsingulärem $A \in \mathbb{R}^{n\times n}$ und $b \in \mathbb{R}^n$ haben wir schon verschiedene Verfahren kennengelernt, so z.B. die LR–, QR– und CHOLESKY–Zerlegung. Diese sind bedingt durch ihren Rechenaufwand von jeweils rund $O(n^3)$ Multiplikationen besonders geeignet für vollbesetze Matrizen ohne besondere Struktur, aber dennoch nur für verhältnismäßig kleine Matrizen. Für sehr große und darüberhinaus noch dünn besetzte Matrizen sind „billigere“ Verfahren mit einem optimalen Rechenaufwand von $O(n)$ zur Ermittelung der n Unbekannten wünschenswert. Solche Matrizen entstehen beispielsweise bei der mehrdimensionalen Spline–Interpolation auf Δ–Gittern oder bei der Diskretisierung partieller Differentialgleichungen für $d > 1$. Typisch hierfür sind Matrizen der Form

$$A = \begin{pmatrix} * & * & & & * & & 0 \\ * & * & * & & & \ddots & \\ & * & * & * & & & * \\ & & \ddots & \ddots & \ddots & & \\ * & & 0 & \ddots & \ddots & \ddots & \\ & \ddots & & & * & * & * \\ 0 & & * & & & * & * \end{pmatrix}.$$

Oftmals müssen lineare Gleichungssysteme nur bis zu einer bestimmten Genauigkeit gelöst werden, die z.B. durch die Diskretisierung einer Differentialgleichung bestimmt ist, so dass sich auch aus diesem Grunde iterative Verfahren anbieten.

11.2 Klassische Iterationsverfahren

Zur Entwicklung eines iterativen Verfahrens für lineare Gleichungssysteme schreibt man ähnlich wie im nichtlinearen Fall die Gleichung

$$Ax = b \tag{11.2.1}$$

mit nichtsingulärem $A \in \mathbb{R}^{n\times n}$ als Fixpunktgleichung

$$x = x + C(b - Ax) =: \Phi(x) \tag{11.2.2}$$

um, wobei $C \in \mathbb{R}^{n\times n}$ nichtsingulär geeignet zu wählen ist. Damit erhält man für $k = 0, 1, 2, \ldots$ eine Iteration der Form

$$x^{k+1} = \Phi(x^k) = x^k + C(b - Ax^k) = (I - CA)x^k + Cb. \tag{11.2.3}$$

Mit der Notation $\rho(M) := |\lambda_{\max}(M)|$ für den Spektralradius einer Matrix $M \in \mathbb{R}^{n\times n}$ gilt folgender

Satz 11.2.4. Das Verfahren (11.2.3) konvergiert genau dann für jeden Startwert x^0 gegen die Lösung von (11.2.1), wenn

$$\rho(I - CA) < 1 \tag{11.2.5}$$

gilt.

Beweis: Zur Notwendigkeit von (11.2.5) nehmen wir an, es gäbe einen Eigenwert λ von $I - CA$ mit $|\lambda| \geq 1$. Wähle den Startwert x^0 nun gerade so, dass $x^0 - x$ Eigenvektor zu λ ist. Mit (11.2.2) und (11.2.3) folgt dann

$$\begin{aligned} x^{k+1} - x &= (I - CA)(x^k - x) = \dots \\ &= (I - CA)^{k+1}(x^0 - x) \\ &= \lambda^{k+1}(x^0 - x), \end{aligned}$$

was

$$\|x^{k+1} - x\| = |\lambda|^{k+1}\|x^0 - x\|$$

impliziert. Ist nun $|\lambda| \geq 1$, so kann die Folge $\{x^k\}_k$ nicht konvergieren. Für die Umkehrung sei daran erinnert, dass zu jeder Matrix M und jedem $\varepsilon > 0$ stets eine Vektornorm konstruiert werden kann, so dass für die zugehörige Matrixnorm $\|\cdot\|_\varepsilon$

$$\|M\|_\varepsilon \leq \rho(M) + \varepsilon$$

gilt. Wählt man speziell $\varepsilon = \frac{1}{2}(1 - \rho(I - CA)) > 0$, so erhält man

$$\begin{aligned} \|I - CA\|_\varepsilon &\leq \rho(I - CA) + \frac{1}{2}(1 - \rho(I - CA)) \\ &= \frac{1}{2} + \frac{1}{2}\rho(I - CA) =: \tilde{\rho} < 1. \end{aligned}$$

Damit gilt für jeden Startwert x^0 aber

$$\begin{aligned} \|x^{k+1} - x\|_\varepsilon &= \|(I - CA)^{k+1}(x^0 - x)\|_\varepsilon \leq \|(I - CA)^{k+1}\|_\varepsilon \|x^0 - x\|_\varepsilon \\ &\leq \|I - CA\|_\varepsilon^{k+1}\|x^0 - x\|_\varepsilon \leq \tilde{\rho}^{k+1}\|x^0 - x\|_\varepsilon \\ &\to 0 \text{ für } k \to \infty. \end{aligned}$$

■

Bemerkung 11.2.6. Da $\Phi'(x) = I - CA$ ist, ist mit (11.2.5) gerade sichergestellt, dass Φ eine Kontraktion ist, so dass der zweite Teil des Beweises auch aus dem Banach'schen Fixpunktsatz aus Kapitel 9.3 folgt.

Korollar 11.2.7. Falls irgendeine Vektornorm existiert, so dass für die zugehörige Matrixnorm $\|\cdot\|$

$$\|I - CA\| < 1 \tag{11.2.8}$$

gilt, konvergiert (11.2.3) für jeden Startwert x^0. Ferner gelten die *A–priori–Fehlerabschätzung*

$$\|x^k - x\| \leq \frac{\|I - CA\|^k}{1 - \|I - CA\|}\|x^1 - x^0\|$$

und die *A–posteriori–Fehlerabschätzung*

$$\|x^k - x\| \leq \frac{\|I - CA\|}{1 - \|I - CA\|}\|x^k - x^{k-1}\|.$$

■

Es ist zu beachten, dass (11.2.5) und (11.2.8) verlangen, dass C die Inverse von A approximiert, denn für $C = A^{-1}$ wäre $I - CA = 0$ und man wäre sofort fertig. Aber die Berechnung von A^{-1} ist aufwendiger als die Lösung von (11.2.1). Es geht also um folgenden Balanceakt: finde eine Matrix C so, dass einerseits C die Matrix A^{-1} genügend gut approximiert (also so, dass (11.2.8) gilt), andererseits aber die Berechnung und Anwendung von C nur einen Aufwand von $O(n)$ erfordert. Dies ist schwierig, da im Allgemeinen auch für dünn besetzte Matrizen A die Inverse A^{-1} vollbesetzt ist. Verschiedene Wahlen von C führen auf unterschiedliche Methoden, die wir im Folgenden nun vorstellen.

Gesamtschrittverfahren (Jacobi–Verfahren)

Sei $a_{ii} \neq 0$ für $i = 1, \ldots, n$. Wähle

$$C = (\mathrm{diag}(a_{11}, \ldots, a_{nn}))^{-1} =: D^{-1}.$$

Mit dieser Wahl lautet die Iterationsvorschrift (11.2.3)

$$x^{k+1} = (I - D^{-1}A)x^k + D^{-1}b. \tag{11.2.9}$$

Für eine etwas andere Schreibweise zerlege A in

$$A = D + L + U, \tag{11.2.10}$$

wobei L eine untere und U eine obere Dreiecksmatrix bezeichnen. Damit lässt sich (11.2.9) als

$$x^{k+1} = -D^{-1}(L + U)x^k + D^{-1}b \tag{11.2.11}$$

bzw. komponentenweise

$$x_i^{k+1} = a_{ii}^{-1} \left(b_i - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij} x_j^k \right) \tag{11.2.11a}$$

für $i = 1, \ldots, n$ schreiben.

Alle Komponenten von x^{k+1} lassen sich unabhängig voneinander parallel aus x^k ermitteln. Daher heißt dieses Verfahren Gesamtschrittverfahren. Ersetzt man L, D und U durch Blockmatrizen, so ergibt sich sofort die Blockversion des Jacobi–Verfahrens.

Die *Konvergenz des Gesamtschrittverfahrens* benötigt Eigenschaften von A.

Satz 11.2.12. Ist A strikt diagonaldominant (starkes Zeilensummenkriterium), d.h.

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}| \tag{11.2.13}$$

für $i = 1, \ldots, n$, so konvergiert (11.2.11) für jeden Startwert x^0. Eine analoge Aussage gilt für das starke Spaltensummenkriterium.

Beweis: Aus (11.2.11) folgt

$$\begin{aligned} \|I - CA\|_\infty &= \|I - D^{-1}A\|_\infty \\ &= \|D^{-1}(L + U)\|_\infty \\ &= \max_i \sum_{\substack{j=1 \\ j \neq i}}^{n} \left| \frac{a_{ij}}{a_{ii}} \right| < 1. \end{aligned}$$

Nun folgt die Behauptung mit Korollar 11.2.7 und $\|\cdot\| = \|\cdot\|_\infty$. ∎

Die strikte Diagonaldominanz in Satz 11.2.12 ist eine sehr starke Forderung, die selbst bei der der Diskretisierung einfacher Differentialgleichungen wie der Poisson–Gleichung

$$-\Delta u = f$$

nicht erfüllt wird. In diesem Fall kann man den Satz aber auf „$\geq$“ in (11.2.13) abschwächen.

Einzelschrittverfahren (Gauss–Seidel–Verfahren)

Alternativ kann man die Matrix C aber auch in Dreiecksgestalt wählen, womit das System ebenfalls einfach aufzulösen bleibt. Wähle dazu

$$C = (D + L)^{-1}.$$

Dann gilt

$$\begin{aligned} x^{k+1} &= (I - CA)x^k + Cb \\ &= (I - (D+L)^{-1}A)x^k + (D+L)^{-1}b \\ &= (I - (D+L)^{-1}(D+L+U))x^k + (D+L)^{-1}b \\ &= -(D+L)^{-1}Ux^k + (D+L)^{-1}b, \end{aligned}$$

was äquivalent zu

$$(D + L)x^{k+1} = -Ux^k + b \tag{11.2.14}$$

oder

$$Dx^{k+1} = -Lx^{k+1} - Ux^k + b \tag{11.2.14a}$$

ist. Prozedural angewandt in Komponentenschreibweise bedeutet dies

$$\sum_{j=1}^{i-1} a_{ij}x_j^{k+1} + a_{ii}x_i^{k+1} = -\sum_{j=i+1}^{n} a_{ij}x_j^k + b_i$$

oder äquivalent dazu

$$x_i^{k+1} = (a_{ii})^{-1}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij}x_j^k\right) \tag{11.2.14b}$$

für $i = 1, \ldots, n$. Im Unterschied zum Gesamtschrittverfahren werden hier also schon berechnete Komponenten von x^{k+1} bei der Berechnung der restlichen Komponenten miteinbezogen. Dieses Verfahren heißt *Einzelschritt–* oder Gauss–Seidel–Verfahren.

Für die Konvergenz dieses Verfahrens gilt

Satz 11.2.15. Das Einzelschrittverfahren (11.2.14) konvergiert für jeden Startwert x^0, falls die Matrix A symmetrisch positiv definit ist.

Beweis: Einen Beweis findet man in [DH], p. 266. ∎

Hierzu sei noch bemerkt, dass das Einzelschrittverfahren ebenfalls für jeden Startwert x^0 konvergiert, falls A strikt diagonaldominant ist. Dieses Verfahren ist ein Spezialfall einer allgemeineren Methode, die im Folgenden hergeleitet wird.

Relaxationsverfahren

Die Konvergenz des Gesamtschrittverfahrens (11.2.11) und des Einzelschrittverfahrens (11.2.14) hängt vom Spektralradius $\rho(I - CA)$ ab. D.h. je kleiner $\rho(I - CA)$ ist, desto schneller konvergiert das jeweilige Verfahren. Für jedes Verfahren gibt es aber eine einfache Beschleunigung, nämlich die *Relaxation*: hierbei bildet man mit einem Parameter ω eine Konvexkombination aus *alten* und *neuen* Iterierten. Anstelle von

$$x^{k+1} = (I - CA)x^k + Cb =: Mx^k + Cb \tag{4.2.3}$$

bilde

$$x^{k+1} := \omega(Mx^k + Cb) + (1-\omega)x^k.$$

Hierbei entspricht die erste Klammer obiger Gleichung den neuen Iterierten und die zweite den alten. Verschiedene Wahlen von ω haben verschiedene Bezeichnungen: Im Fall $\omega > 1$ spricht man von *Überrelaxation* (overrelaxation) und im Fall $\omega < 1$ von *Unterrelaxation*. Für $\omega > 1$ heißt das Verfahren allgemein *SOR–Verfahren* (successive overrelaxation). ω heißt hierbei *Relaxations–* oder *Dämpfungsparameter*. Eine typische Wahl ist $C = (D+L)^{-1}$ (wie beim Einzelschritt–/Gauss–Seidel–Verfahren) und eine Konvexkombination aus Iterierten in der Form von (11.2.14a), so dass die Iterationvorschrift des SOR–Verfahrens

$$Dx^{k+1} = \omega(-Lx^{k+1} - Ux^k + b) + (1-\omega)Dx^k \tag{11.2.16}$$

bzw. in Komponentenschreibweise

$$a_{ii}x_i^{k+1} = \omega\left(-\sum_{j<i} a_{ij}x_j^{k+1} - \sum_{j>i} a_{ij}x_j^k + b_i\right) + (1-\omega)a_{ii}x_i^k \tag{11.2.16a}$$

lautet.

Satz (von Ostrowski–Reich**)** Sei $A \in \mathbb{R}^{n\times n}$ symmetrisch positiv definit. Dann konvergiert das SOR–Verfahren (11.2.16) für $0 < \omega < 2$. Im Fall $\omega = 1$ entspricht das SOR–Verfahren dem Gauss–Seidel–Verfahren.

Beweis: Einen Beweis findet man in [SB2], p. 581. ∎

SSOR–Verfahren

Eine Variante des SOR–Verfahrens ist das symmetrische SOR–Verfahren, das SSOR–Verfahren. In (11.2.16a) werden die Komponenten von x^{k+1} „von oben nach unten“ durchlaufen. Dies kann man auch umkehren: Ein durchlaufen der Iteration jeweils in Vorwärts– und Rückwärtsrichtung ergibt

$$\begin{aligned} Dx^{k+1/2} &= \omega(-Lx^{k+1/2} - Ux^k + b) + (1-\omega)Dx^k \\ Dx^{k+1} &= \omega(-Lx^{k+1/2} - Ux^{k+1} + b) + (1-\omega)Dx^{k+1/2}. \end{aligned} \tag{11.2.17}$$

Eine Konvergenzaussage ist mit dem Satz von Ostrowski–Reich ableitbar. Der Vorteil dieses Verfahrens ist, dass der Aufwand von k SSOR–Zyklen nicht $2k$ SOR, sondern nur $k + \frac{1}{2}$ Zyklen SOR entspricht. Weiter führt das Verfahren auf eine symmetrische Iterationsmatrix. Bemerkungen zur Güte der einzelnen Verfahren werden später gegeben.

11.3 Gradientenverfahren

Im Folgenden sei die Matrix $A \in \mathbb{R}^{n\times n}$ stets symmetrisch positiv definit. Solche Matrizen entstehen typischerweise bei der Diskretisierung elliptischer partieller Differentialgleichungen. Eine besondere Rolle spielt in diesem Fall das (p)cg–Verfahren (preconditioned conjugate gradient). Vorher wird allerdings das allgemeine Gradientenverfahren behandelt.

Lemma 11.3.1. Sei $A \in \mathbb{R}^{n\times n}$ symmetrisch positiv definit. Dann gilt: $x^* \in \mathbb{R}^n$ löst die Gleichung $Ax = b$ genau dann, wenn x^* Minimum von

$$f(x) = \frac{1}{2}x^T Ax - x^T b \tag{11.3.2}$$

ist.

Beweis: Nach Voraussetzung ist $f \in \mathcal{C}^2(\mathbb{R}^n)$. Die notwendige Bedingung für ein lokales Minimum lautet

$$\nabla f(x) = 0 \Longleftrightarrow Ax - b = 0.$$

Also ist $Ax - b$ ein kritischer Punkt. Da $D^2 f(x) = A$ nach Voraussetzung positiv definit ist, ist dieses lokale Minimum sogar global, und die notwendige Bedingung ist auch hinreichend.

■

In Übungsausfgabe 19 wurde die Kantorovich–Ungleichung verwendet, die wir nun hier noch einmal beweisen wollen:

(Kantorovich–Ungleichung): Sei $A \in \mathbb{R}^{n\times n}$ symmetrisch positiv definit. Dann gilt für alle $x \in \mathbb{R}^n$:

$$\frac{(x^T x)^2}{(x^T Ax)(x^T A^{-1}x)} \geq \frac{4Mm}{(M-m)^2}$$

mit $m := \lambda_{min(A)}$ und $M := \lambda_{max(A)}$.

Beweis: Da A symmetrisch positiv definit ist, existieren n paarweise verschiedene und zueinander orthogonale Eigenvektoren $v_i \in \mathbb{R}^n$, welche eine Orthogonalbasis des $\mathbb{R}^n$ bilden. Schreibt man nun das x als Linerakombination der v_i mit Koeffizienten $\alpha_i \in \mathbb{R}$: $x = \sum\limits_{i=1}^{n} \alpha_i v_i$ und definiert sich $\beta := \frac{\alpha_i^2}{\sum\limits_{i=1}^{n} \alpha_i^2}$, so ergibt sich:

$$\begin{aligned}
\frac{(x^T x)^2}{(x^T Ax)(x^T A^{-1}x)} &= \frac{(\sum\limits_{i=1}^{n} \alpha_i^2)^2}{(\sum\limits_{i=1}^{n} \alpha_i^2 \lambda_i)(\sum\limits_{i=1}^{n} \alpha_i^2 \lambda_i^{-1})} \\
&= \frac{1}{(\sum\limits_{i=1}^{n} \beta_i \lambda_i)(\sum\limits_{i=1}^{n} \beta_i \lambda_i^{-1})}
\end{aligned}$$

wobei die λ_i die Eigenwerte von A sind. Als nächstes definiere $\mu := \sum\limits_{i=1}^{n} \beta_i \lambda_i$, dann gilt: $m \leq \mu \leq M$, denn es gilt $\beta_i \geq 0$ und $\sum\limits_{i=1}^{n} \beta_i = 1$. Schätze dies weiter ab:

$$\begin{aligned}
\sum_{i=1}^{n} \beta_i \lambda_i^{-1} &\leq \frac{m + M + \mu}{mM} \\
\Leftrightarrow \sum_{i=1}^{n} \beta_i \lambda_i^{-1} + \sum_{i=1}^{n} \beta_i \frac{\lambda_i}{mM} &\leq M^{-1} - m^{-1}
\end{aligned}$$

Somit lässt sich nun insgesamt abschätzen:

$$\begin{aligned}
\frac{(x^T x)^2}{(x^T Ax)(x^T A^{-1}x)} &\geq \frac{mM}{M(m+M+\mu)} \\
&\geq \min_{m\leq\mu\leq M} \frac{mM}{M(m+M+\mu} \\
&= \frac{mM}{(\frac{m+M}{2})(\frac{m+M}{2})} \\
&= \frac{4mM}{(m+M)^2}
\end{aligned}$$

■

Das *Gradientenverfahren* besteht nun darin, das Minimum

$$\min_{x\in\mathbb{R}^n} f(x) \tag{11.3.3}$$

einer Funktion $f \in \mathcal{C}^1(\Omega)$ mit $\Omega \subset \mathbb{R}^n$ offen zu finden. Algorithmisch entspricht dies

Algorithmus 11.3.4. (Allgemeines Gradientenverfahren mit vollständiger linearer Suche)
Sei $\Omega \subset \mathbb{R}^n$ offen und $f \in \mathcal{C}^1(\Omega)$. Wähle $x_0 \in \Omega$. Für $k = 0, 1, 2, \ldots$

1.) Bestimmung der Richtung: berechne die „Abstiegsrichtung“

$$d_k = -\nabla f(x_k).$$

2.) Liniensuche: suche das Minimum von f auf der Linie

$$\{x_k + \alpha d_k,\ \alpha \geq 0\} \cap \Omega.$$

Das (lokale) Minimum werde bei $\alpha = \alpha_k$ angenommen. Setze dann

$$x_{k+1} = x_k + \alpha_k d_k.$$

■

Zur Abstiegsrichtung $-\nabla f(x_k)$ und praktischen Durchführung dieses Verfahrens kommen wir gleich. Zunächst betrachten wir aber die quadratische Funktion f in (11.3.2). Hier gilt speziell

$$d_k = b - Ax_k \tag{11.3.5}$$

$$\alpha_k = \frac{d_k^T d_k}{d_k^T A d_k}, \tag{11.3.6}$$

denn setze

$$\begin{aligned}
\tilde{f}(\alpha) &:= f(x_k + \alpha d_k) \\
&= \frac{1}{2}(x_k + \alpha d_k)^T A(x_k + \alpha d_k) - (x_k + \alpha d_k)^T b \\
&= \frac{1}{2}(x_k^T Ax_k + 2\alpha d_k^T Ax_k + \alpha^2 d_k^T Ad_k) - x_k^T b - \alpha d_k^T b.
\end{aligned}$$

Dann gilt

$$\begin{aligned}\tilde{f}'(\alpha) &= d_k^T A x_k + \alpha d_k^T A d_k - d_k^T b \\ &= d_k^T (A x_k - b) + \alpha d_k^T A d_k \\ &= -d_k^T d_k + \alpha d_k^T A d_k \\ &\stackrel{!}{=} 0,\end{aligned}$$

womit

$$\alpha_k = \frac{d_k^T d_k}{d_k^T A d_k}$$

folgt.

Für die Abstiegsrichtung ist ausgehend von $f(x_k)$ ein d zu bestimmen, so dass

$$f(x_k + \beta d) \leq f(x_k)$$

ist. D.h., finde ein d mit $\|d\| = 1$, so dass die Richtungsableitung $D_d f(x_k) = (\nabla f(x_k))^T d$ minimal ist. Die Wahl

$$d = -\frac{\nabla f(x_k)}{\|\nabla f(x_k)\|}$$

liefert die Richtung des steilsten Abstiegs. Da $D_d f = \langle \nabla f, d \rangle = \cos\theta \|\nabla f\| \|d\|$, ist im Fall $\nabla f \neq 0$ die Ableitung $D_d f$ minimal, falls $\theta = 0$, die Richtungsableitung also maximal, wenn ∇f und d dieselbe Richtung haben.

Zur praktischen Durchführung des Gradientenverfahrens bemerken wir noch, dass der zweite Schritt nur näherungsweise ausgeführt wird. Man bestimmt z.B. einen Punkt $x_k + t d_k$, in dem die Richtungsableitung klein, etwa

$$|d_k^T \nabla f(x_k + t d_k)| < \frac{1}{4} |d_k^T \nabla f(x_k)|$$

ist. Hierfür kann man folgende *globale Konvergenzaussage* zeigen: es konvergiert wenigstens eine Teilfolge gegen $x^* \in \Omega$ mit $\nabla f(x^*) = 0$, oder die Folge konvergiert gegen den Rand von Ω. (x^* kann allerdings auch nur lokales Minimum oder Sattelpunkt sein; daher sei im folgenden f stets quadratisch angenommen.)

Bemerkung 11.3.7. Die Funktion $f(x) = \frac{1}{2} x^T A x - x^T b$ ist quadratisch auf $\mathbb{R}^n$, falls A symmetrisch positiv definit ist. In diesem Fall konvergiert das Gradientenverfahren für jeden Startwert gegen das Minimum von f.

Konvergenzgeschwindigkeit

Als Maß für den Abstand betrachte die *Energienorm*

$$\|x\|_A := \sqrt{x^T A x}. \tag{11.3.8}$$

Sie ist wohldefiniert, da jede symmetrisch positiv definite Matrix A ein Skalarprodukt $(x, y)_A := x^T A y$ definiert. Für das Gradientenverfahren in (11.3.4) gilt

Satz 11.3.9. Sei A symmetrisch positiv definit und $\kappa = \kappa_2(A) = \frac{\lambda_{\max}}{\lambda_{\min}}$ ihre spektrale Kondition. Dann liefert das Gradientenverfahren (11.3.4) für $f(x) = \frac{1}{2} x^T A x - x^T b$ eine Folge $\{x_k\}_{k=0,1,2,\ldots}$ (x_0 als Startwert) mit

$$\|x_k - x^*\|_A \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^k \|x_0 - x^*\|_A, \tag{11.3.10}$$

wobei x^* die Lösung von $Ax = b$ bezeichne.

Beweis: Einen Beweis findet man etwa in [Br]. ∎

Ist κ sehr groß, dann ist wegen

$$\frac{\kappa-1}{\kappa+1} = 1 - \frac{2}{\kappa+1} \approx 1 - \frac{2}{\kappa}$$

die Konvergenzrate nahe bei 1 und die Konvergenz damit sehr langsam.

Geometrische Erklärung

Seien $a \gg 1$ und $A = \begin{pmatrix} 1 & 0 \\ 0 & a \end{pmatrix}$ mit $\kappa(A) = a$. D.h. mit $x = (y,z)^T$ gilt

$$f(y,z) = \frac{1}{2}(y^2 + az^2).$$

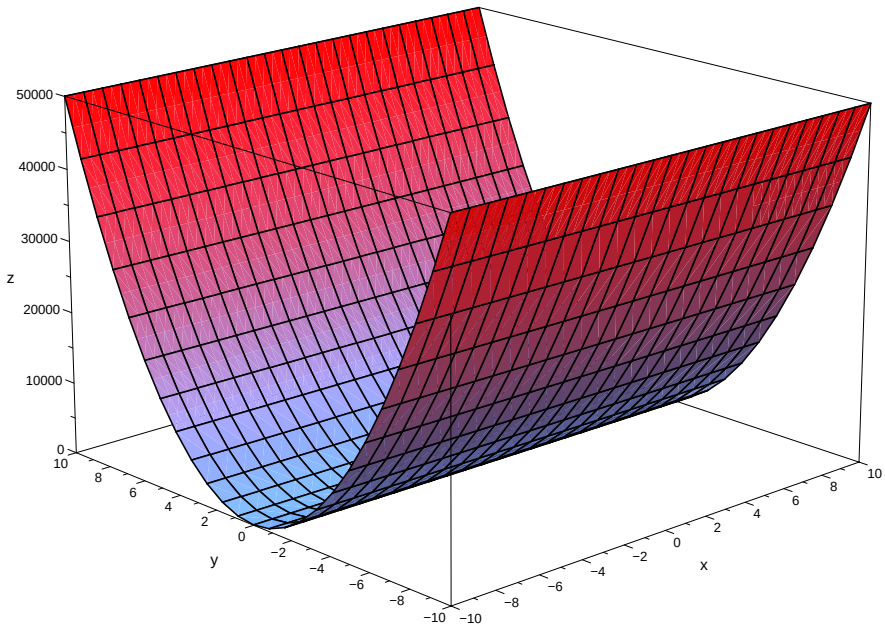
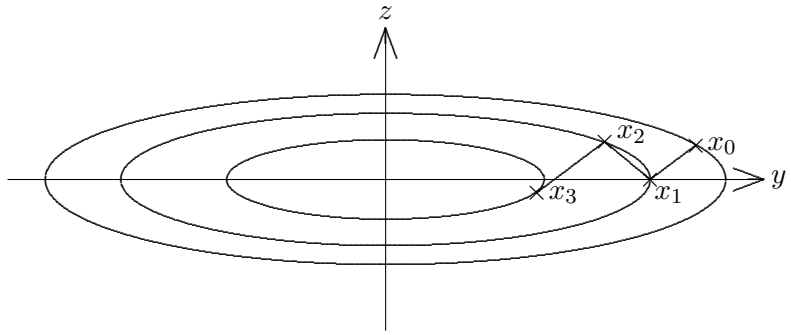


Abbildung 2: Funktion f mit a=1000

Wähle als Startvektor $x_0 = (y_0, z_0)^T = (a, 1)^T$. Für die Abstiegsrichtung gilt dann

$$d_0 = -\nabla f(x_0) = -\begin{pmatrix} y \\ az \end{pmatrix}\Bigg|_{(y,z)=x_0} = a\begin{pmatrix} -1 \\ -1 \end{pmatrix}.$$

Weiter bekommt man $x_k = \begin{pmatrix} y_k \\ z_k \end{pmatrix}$ mit $y_{k+1} = \rho y_k$ und $z_{k+1} = -\rho z_k$, wobei $\rho = \frac{a-1}{a+1}$ ist. Also ist $d_k = -\nabla f(x_k)$ zwar Richtung des steilsten Abstiegs, dieser kann jedoch sehr langsam sein. Der Grund dafür ist, dass die Höhenlinien von f – in die Ebene projiziert – stark gestreckte Ellipsen sind:



Das Verfahren ergibt, wie die Skizze zeigt, eine Art „Zick–Zack–Kurs“, in dem d_k und d_{k+1} zwar orthogonal bzgl. des EUKLIDISCHEN Skalarprodukts, aber fast parallel bzgl. $(\cdot,\cdot)_A$ sind. Dies liegt an den Eigenschaften von A, was zeigt, dass man diese mit in die Richtungssuche einbeziehen sollte. Dies wird im nächsten Abschnitt über das cg–Verfahren der Fall sein.
Zunächst wollen wir uns aber noch einige Zusammenhänge zwischen Kondition, Norm und Spektralradius anschauen:
Die *Kondition* einer Matrix $A \in \mathbb{R}^{m\times n}$ ist gegeben durch $\kappa(A) = \frac{max_{\|x\|=1}\|Ax\|}{min_{\|x\|=1}\|Ax\|} \in [1,\infty]$. Wenn A singulär und ungleich der Nullmatrix ist, dann gilt $\kappa(A) = \infty$.
Ist A invertierbar, so gilt $\kappa(A) = \|A\|\|A^{-1}\|$.
Sei $\|\cdot\|_*$ eine Vektornorm und $\|\cdot\|_{*\to*}$ die zugehörige Operatornorm, dann ist: $\kappa_*(A) = \|A\|_{*\to*}\|A^{-1}\|_{*\to*} \geq \frac{|\lambda_{max}|}{|\lambda_{min}|}$.
Wenn $\|\cdot\|_* = \|\cdot\|_2$ und A symmetrisch ist, so gilt $\kappa_2(A) = \frac{|\lambda_{max}|}{|\lambda_{min}|}$.
Der *Spektralradius* von A ist definiert durch $\rho(A) := |\lambda_{max}(A)|$. In Abschnitt **??** hatten wir bereits die folgende Aussage verwendet.

Satz. Sei $A \in \mathbb{C}^{n\times n}$ und sei $\|\cdot\|$ eine beliebige Matrixnorm, dann gilt:

- $\rho(A) \leq \|A\|$
- Seien A und ein beliebiges $\epsilon > 0$ gegeben. Dann gibt es eine Matrixnorm, zugehörig zu einer Vektornorm, so dass
$$\|A\| \leq \rho(A) + \epsilon$$

Außerdem gilt: Ist A normal, d.h. $AA^* = A^*A$, so gilt $\|A\|_{2\to2} = \rho(A)$.

Beweis: Einen Beweis findet man in [C]. ∎

11.4 cg–Verfahren

Das Verfahren der konjugierten Gradienten/cg–Verfahren (conjugate gradient) wurde 1952 von HESTENES und STIEFEL entwickelt und hat eine wachsende Bedeutung seit den 1970'er Jahren, bedingt durch die Entwicklung effizienter Vorkonditionierer. Der Grundgedanke bei diesem Verfahren ist der, im Gradientenverfahren aus dem vorigen Abschnitt aufeinanderfolgende Richtungen so zu wählen, dass sie orthogonal bzgl. $(\cdot,\cdot)_A$ sind. Unsere Herleitung ist an [Br] angelehnt.

Im Folgenden sei die Matrix A stets symmetrisch positiv definit.

Definition. Zwei Vektoren $x, y \in \mathbb{R}^n$ heißen *A–orthogonal* oder *A–konjugiert*, falls $(x,y)_A = \langle x, Ay\rangle = x^TAy = 0$ ist. Sind x, y A–konjugiert und gilt $x \not\equiv 0, y \not\equiv 0$, so sind x, y linear unabhängig, da A als positiv definit vorausgesetzt ist.

Seien nun $d_0, \ldots, d_{n-1}$ konjugierte Richtungen, d.h. $d_i \in \mathbb{R}^n$ und $(d_i, d_j)_A = 0$ für $i \neq j$. Dann besitzt $x^* = A^{-1}b$ die Darstellung

$$x^* = \sum_{k=0}^{n-1} \gamma_k d_k. \tag{11.4.1}$$

Für die Berechnung der γ_k gilt

$$\begin{aligned}(d_i, x^*)_A &= \left(d_i, \sum_{k=0}^{n-1} \gamma_k d_k\right)_A \\ &= \sum_{k=0}^{n-1} \gamma_k (d_i, d_k)_A \\ &= \gamma_i (d_i, d_i)_A,\end{aligned}$$

womit

$$\gamma_i = \frac{(d_i, x^*)_A}{(d_i, d_i)_A} = \frac{\langle d_i, b\rangle}{(d_i, d_i)_A} = \frac{d_i^T b}{d_i^T A d_i} \tag{11.4.2}$$

folgt. D.h. die Entwicklungskoeffizienten in (11.4.1) lassen sich direkt aus b, d_i berechnen.

Lemma 11.4.3. Seien $d_0, \ldots, d_{n-1} \in \mathbb{R}^n$ konjugierte Richtungen. Sei $x_0 \in \mathbb{R}^n$ ein beliebiger Startvektor und definiere für $k \geq 0$

$$x_{k+1} = x_k + \alpha_k d_k \tag{11.4.4}$$

mit

$$\alpha_k = -\frac{r_k^T d_k}{d_k^T A d_k} \tag{11.4.5}$$

und $r_k := Ax_k - b = \nabla f(x_k)$ als *Residuum*. Dann gilt: die Folge $\{x_k\}_{k\in\mathbb{N}_0}$ liefert nach (höchstens) n Schritten die Lösung $x_n = A^{-1}b$.

Der Unterschied zum Gradientenverfahren ist, dass hier zusätzlich die Residuen r_k eingeführt wurden; vorher waren die Richtungen die Residuen.
Beachte, dass das Verfahren in (11.4.4) noch nicht praktisch verwendbar ist, da die konjugierten Richtungen $d_0, \ldots, d_{n-1}$ im Gegensatz zum eigentlichen Gradientenverfahren noch nicht bekannt sind. Theoretisch könnte man versuchen, die d_i *vor* dem Start des Algorithmus zu berechnen, z.B. mit dem Gram–Schmidt–Verfahren. Dies ist jedoch nicht sinnvoll, da es viel zu aufwendig ist. Später werden wir sehen, dass $m \ll n$ Schritte für die Iteration ausreichen und die d_i direkt mitberechnet werden können.

Beweis: Eine Rekursion von (11.4.4) ergibt

$$x_k = x_{k-1} + \alpha_{k-1} d_{k-1} = x_{k-2} + \alpha_{k-2} d_{k-2} + \alpha_{k-1} d_{k-1} = \ldots = x_0 + \sum_{j=0}^{k-1} \alpha_j d_j. \tag{11.4.6}$$

Sei nun x^* Lösung von $Ax = b$. Wir zeigen, dass $x^* = x_n$ gilt. Seien dazu $d_0, \ldots, d_{n-1}$ konjugiert (also linear unabhängig). Es gilt dann

$$x^* - x_0 = \sum_{j=0}^{n-1} \beta_j d_j$$

mit (irgendwelchen) $\beta_j \in \mathbb{R}$, für die

$$\begin{aligned}
\frac{d_i^T A(x^* - x_0)}{d_i^T A d_i} &= \sum_{j=0}^{n-1} \beta_j \frac{d_i^T A d_j}{d_i^T A d_i} \\
&= \beta_i \frac{d_i^T A d_i}{d_i^T A d_i} \\
&= \beta_i
\end{aligned}$$

gilt und weiterhin auch

$$\beta_i = \frac{d_i^T (Ax^* - Ax_0)}{d_i^T A d_i} = \frac{d_i^T (b - Ax_0)}{d_i^T A d_i} = \frac{-d_i^T r_0}{d_i^T A d_i}.$$

Als nächstes zeigen wir, dass $\beta_i = \alpha_i = \frac{-r_i^T d_i}{d_i A d_i}$ gilt. Dies ist aber der Fall wegen

$$\begin{aligned}
\beta_i &= -\frac{d_i^T (Ax_0 - b)}{d_i^T A d_i} \\
&= -\frac{d_i^T (Ax_0 - b + A \sum_{j=0}^{i-1} \alpha_j d_j)}{d_i^T A d_i} \\
&= -\frac{d_i^T (A(x_0 + \sum_{j=0}^{i-1} \alpha_j d_j) - b)}{d_i^T A d_i} \\
&= -\frac{d_i^T (Ax_i - b)}{d_i^T A d_i} \\
&= -\frac{d_i^T r_i}{d_i^T A d_i} \\
&= \alpha_i.
\end{aligned}$$

Also gilt

$$x^* - x_0 = \sum_{j=0}^{n-1} \alpha_j d_j \Longleftrightarrow x^* = x_0 + \sum_{j=0}^{n-1} \alpha_j d_j = x_n \tag{4.4.5a}$$

nach Definition der Iteration in (11.4.4). ∎

Korollar 11.4.7. Seien $d_0, \ldots, d_{n-1} \in \mathbb{R}^n$ konjugierte Richtungen. Für jedes in (11.4.4) mit (11.4.5) definierte x_k mit Startvektor x_0 gilt

$$f(x_k) = \min_{x \in V_k} f(x_0 + x), \tag{11.4.8}$$

wobei

$$V_k = \operatorname{span}\{d_0, \ldots, d_{k-1}\}. \tag{11.4.9}$$

Insbesondere gilt

$$d_i^T r_k = 0 \tag{11.4.10}$$

für $i < k$, d.h. das k–te Residuum ist orthogonal zu V_i. ∎

Im Unterschied zum Gradientenverfahren aus (11.3.4), bei dem man zunächst die Richtung $\tilde{d}_k = -\nabla f(x_k)$ und dann das Minimum von f auf der Linie $\{x_k + \tilde{\alpha}\tilde{d}_k,\ \tilde{\alpha} \geq 0\}$, welches bei $\tilde{\alpha} = \tilde{\alpha}_k$

angenommen wurde, bestimmte, wird hier durch die konjugierten Richtungen $d_0, \ldots, d_{k-1}$ das Minimum von f bzgl. des *affinen* Teilraums

$$\left\{ x \in \mathbb{R}^n : \; x = x_0 + \sum_{j=0}^{k-1} \gamma_j d_j, \; \gamma_j \in \mathbb{R} \right\}$$

angenommen. Beim Gradientenverfahren ist der affine Teilraum jeweils nur die Gerade $\{x : \; x_k + \tilde{\alpha} \tilde{d}_k\}$.

Beweis von Korollar 11.4.7: Es ist zu zeigen, dass x_k die Funktion f über dem affinen Teilraum V_k minimiert, also $f(x_k) = \min_{x \in V_k} f(x_0 + x)$, und Gleichung (11.4.10) $d_i r_k = 0$ für $i < k$ stimmt. Zur ersten Aussage betrachte

$$\begin{aligned} \tilde{f}(\alpha_0, \ldots, \alpha_{k-1}) := f(x_k) &= f\left(x_0 + \sum_{j=0}^{k-1} \alpha_j d_j \right) \\ &= \frac{1}{2} x_k^T A x_k - b^T x_k, \end{aligned}$$

was

$$\nabla \tilde{f}(\alpha_0, \ldots, \alpha_{k-1}) = \begin{pmatrix} d_0^T r_0 + \alpha_0 d_0^T A d_0 \\ \vdots \\ d_{k-1}^T r_{k-1} + \alpha_{k-1} d_{k-1}^T A d_{k-1} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$$

nach sich zieht. (Dies gilt auch beim Gradientenverfahren, allerdings müssen dort die $\tilde{d}_i$ nicht linear unabhängig sein.) Da die d_i nach Voraussetzung für alle $i = 0, \ldots, k-1$ konjugiert sind, minimiert x_k die Funktion f über $x_0 + V_k = \{x_0 + x, \; x \in V_k\}$. Zum Beweis von (11.4.10) gilt

$$\begin{aligned} d_i^T r_{i+1} &= d_i^T (A x_{i+1} - b) \\ &= d_i^T (A(x_i + \alpha_i d_i) - b) \\ &= d_i^T (A x_i - b) + \alpha_i d_i^T A d_i \\ &= d_i^T r_i + \alpha_i d_i^T A d_i \\ &= 0 \end{aligned}$$

nach Wahl von α_i. Also gilt (11.4.10) für $i = k-1$ und für $k = 1$, da $d_0^T r_1 = 0$. Den Rest zeigen wir per vollständiger Induktion nach k. Sei also $d_i^T r_k = 0$ richtig für $i < k$. Die Gültigkeit für $k+1$ ist zu zeigen. Nach Definition gilt

$$x_{k+1} - x_k = \alpha_k d_k,$$

was

$$\begin{aligned} r_{k+1} - r_k &= A x_{k+1} - b - (A x_k - b) \\ &= A(x_{k+1} - x_k) \\ &= \alpha_k A d_k \end{aligned}$$

impliziert. Damit gilt aber

$$d_i^T (r_{k+1} - r_k) = \alpha_k d_i^T A d_k = 0$$

für $k > i$; insbesondere ist (11.4.10) also auch für $k+1$ richtig. ∎

Aus Lemma 11.4.3 lässt sich ein Algorithmus entwickeln, sobald man die konjugierten Richtungen d_i berechnen kann. Tatsächlich berechnet man die d_i aus $\nabla f(x_i)$ durch Addition einer Korrektur. Dies ist die entscheidende Verwandtschaft zwischen cg– und Gradientenverfahren.

In Lemma 11.3.1 haben wir gezeigt, dass x^* genau dann die Gleichung $Ax = b$ löst, wenn es die Funktion $f(x) = \frac{1}{2}x^T Ax - x^T b$ minimiert. Für das Gradientenverfahren aus (11.3.4) erhält man mit beliebigem Startwert $x_0 \in \mathbb{R}^n$ als erste Richtung

$$d_0 := -\nabla f(x_0) = b - Ax_0 =: -r_0$$

als Residuum des Startwerts. Nehme nun für $k = 0, 1, 2, \ldots$ als d_{k+1} denjenigen Anteil des Residuums $-\nabla f(x_{k+1}) = b - Ax_{k+1}$, der zu $d_0, \ldots, d_k$ konjugiert ist. Setze also an

$$r_{k+1} = -d_{k+1} + \sum_{j=0}^{k} \beta_j d_j \Longleftrightarrow d_{k+1} = -r_{k+1} + \sum_{j=0}^{k} \beta_j d_j.$$

Stelle dabei sicher, dass $r_{k+1} \notin \operatorname{span}\{d_0, \ldots, d_k\}$ ist, denn sonst ist $d_{k+1} = 0$ möglich. Es lässt sich sogar erreichen, dass $\beta_j = 0$ für alle $j = 0, \ldots, k-1$ ist:

Lemma 11.4.11. Sei $d_0 = b - Ax_0 = -r_0$. Für $k = 0, 1, 2, \ldots$ setze

$$\beta_k = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k} \tag{11.4.12}$$

solange $r_k \neq 0$ und

$$d_{k+1} = -r_{k+1} + \beta_k d_k. \tag{11.4.13}$$

Dann gilt: die d_j sind konjugiert, d.h. $d_i^T A d_j = 0$ für $i \neq j$, und

$$d_i^T r_j = -r_i^T r_j = 0 \tag{11.4.14}$$

für $i < j$.

Beachte: Sobald $r_k = 0 = Ax_k - b$ gilt, ist man fertig. $d_i^T r_j = 0$ galt für $i < j$ schon in (11.4.10).

Beweis: Wir zeigen mit vollständiger Induktion über k, dass die d_{k+1} gemäß (11.4.13) A-konjugiert sind. Für den Induktionsanfang sei $k = 0$. Wegen (11.4.10) und der Wahl von d_0 gilt

$$0 = d_0^T r_1 = -r_0^T r_1.$$

Also sind d_0 und r_1 linear unabhängig und man kann ansetzen

$$d_1 = -r_1 + \gamma d_0.$$

Nun gilt mit α_0 gemäß (11.4.5)

$$\begin{aligned}
\alpha_0 d_0^T A d_1 &= \alpha_0 d_0^T A(-r_1 + \gamma d_0) \\
&= -\alpha_0 d_0^T A r_1 + \gamma \alpha_0 d_0^T A d_0 \\
&= -\alpha_0 d_0^T A r_1 - \gamma d_0^T r_0 \\
&= -\alpha_0 d_0^T A r_1 - r_0^T r_1 + \gamma r_0^T r_0 \\
&= -(r_0 + \alpha_0 A d_0)^T r_1 + \gamma r_0^T r_0 \\
&= -r_1^T r_1 + \gamma r_0^T r_0,
\end{aligned}$$

d.h. d_0 und d_1 sind genau dann konjugiert, also $d_0^T A d_1 = 0$, wenn

$$\gamma = \frac{r_1^T r_1}{r_0^T r_0} = \beta_0$$

gilt, was den Induktionsanfang beendet. Induktionsannahme: Seien $d_0, \ldots, d_k$ konjugiert. Wir haben zu zeigen, dass dann auch $d_0, \ldots, d_{k+1}$ mit d_{k+1} gemäß (11.4.12) und (11.4.13) konjugiert sind. Mit α_k aus (11.4.5) gilt wieder

$$\begin{aligned}
\alpha_k d_k^T A d_{k+1} &= \alpha_k d_k^T A(-r_{k+1} + \beta_k d_k) \\
&= -\alpha_k d_k^T A r_{k+1} + \beta_k \alpha_k d_k^T A d_k \\
&= -(r_k + \alpha_k A d_k)^T r_{k+1} - \beta_k r_k^T d_k \\
&= -r_{k+1}^T r_{k+1} + \beta_k r_k^T r_k.
\end{aligned}$$

Also ist d_{k+1} genau dann zu d_k konjugiert, wenn

$$\beta_k = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$$

gilt. Es bleibt zu zeigen, dass $r_k^T r_{k+1} = 0 (= -r_k^T d_k)$ gilt. Dies aber folgt aus

$$\begin{aligned}
r_{k+1}^T r_k &= r_{k+1}^T(\beta_{k-1} d_{k-1} - d_k) \\
&= \beta_{k-1} r_{k+1}^T d_{k-1} - r_{k+1}^T d_k \\
&= 0.
\end{aligned}$$

■

Nun können wir den Algorithmus des cg–Verfahrens angeben.

Algorithmus (cg–Verfahren): Gegeben seien eine Matrix $A \in \mathbb{R}^{n \times n}$ symmetrisch positiv definit und $b \in \mathbb{R}^n$. Gesucht sei die Lösung x^* der Gleichung $Ax = b$, die gleichzeitig die Funktion $f(x) = \frac{1}{2} x^T A x - x^T b$ minimiert.

- Wähle $x_0 \in \mathbb{R}^n$ und setze

$$d_0 := -\nabla f(x_0) = b - A x_0 =: -r_0.$$

- Für $k = 0, 1, 2, \ldots$ berechne dann

$$\begin{aligned}
\alpha_k &= \frac{r_k^T r_k}{d_k^T A d_k} \\
x_{k+1} &= x_k + \alpha_k d_k \\
r_{k+1} &= r_k + \alpha_k A d_k.
\end{aligned}$$

  Falls x_{k+1} genügend genau (oder $r_{k+1} = 0$) breche ab. Sonst setze

$$\begin{aligned}
\beta_{k+1} &= \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k} \\
d_{k+1} &= -r_{k+1} + \beta_k d_k.
\end{aligned}$$

■

Beachte die besondere Form von α_k in diesem Algorithmus:

$$\alpha_k = \frac{-r_k^T d_k}{(d_k, d_k)_A} = \frac{-r_k^T(-r_k + \beta_{k-1} d_{k-1})}{(d_k, d_k)_A} = \frac{r_k^T r_k}{(d_k, d_k)_A} - \frac{\beta_{k-1}(r_k^T d_{k-1})}{(d_k, d_k)_A} = \frac{r_k^T r_k}{(d_k, d_k)_A}.$$

Außerdem ist zu beachten, dass das Abbruchkriterium nur approximativ geprüft werden braucht, z.B. durch $\|r_k\|_2 < tol$, wobei tol vom Problem selbst abhängig ist.

Eine etwas andere Sichtweise des cg–Verfahrens ist die folgende: Nach (4.4.5a) besitzt die exakte Lösung x^* von $Ax = b$ die Darstellung

$$\begin{aligned} x^* &= x_0 + \sum_{j=0}^{n-1} \alpha_j d_j \\ &= x_0 + \sum_{j=0}^{k-1} \alpha_j d_j + \sum_{j=k}^{n-1} \alpha_j d_j \\ &= x_k + \sum_{j=k}^{n-1} \alpha_j d_j, \end{aligned}$$

was äquivalent zu

$$x^* - x_k = \sum_{j=k}^{n-1} \alpha_j d_j$$

ist. Damit gilt weiter

$$\begin{aligned} (x^* - x_k, d_i)_A &= \left(\sum_{j=k}^{n-1} \alpha_j d_j, d_i \right)_A \\ &= \sum_{j=k}^{n-1} \alpha_j (d_j, d_i)_A \\ &= 0 \end{aligned}$$

für $i < k$, was $(x^* - x_k, v_k)_A = 0$ für alle $v_k \in V_k = \operatorname{span}\{d_0, \ldots, d_{k-1}\}$ impliziert. Dies ist nach Satz 11.3.3 äquivalent zu

$$\|x^* - x_k\|_A = \min_{x \in V_k} \|(x^* - x_0) - x\|_A,$$

d.h. das cg–Verfahren erzeugt, da die d_i konjugiert sind, eine A–orthogonale Projektion auf den Teilraum V_k. Das Gradientenverfahren produzierte ebenfalls eine in $x_0 + V_k$ liegende Lösung, diese hatte allerdings nicht minimale Energienorm.

Konvergenzgeschwindigkeit

Satz 11.4.15. Für das cg–Verfahren gilt mit jedem beliebigen Startwert $x_0 \in \mathbb{R}^n$

$$\|x_k - x^*\|_A \leq 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k \|x_0 - x^*\|_A, \tag{11.4.16}$$

wobei $\kappa = \kappa_2(A)$ die spektrale Kondition von A bezeichnet.

Beweis: Einen Beweis findet man in [DH], p. 278f.. ∎

Gleichung (11.4.16) besagt, wie der Anfangsfehler im *ungünstigsten* Fall reduziert wird. Dies bedeutet, wenn k so gewählt ist, dass die rechte Seite kleiner als die Toleranzgrenze gewählt ist, gilt natürlich

$$\|x_k - x^*\|_A \leq \text{ tol.}$$

Vorkonditionierung

Anstelle das cg–Verfahren auf die Gleichung

$$Ax = b$$

anzuwenden, kann man auch das vorkonditionierte pcg–Verfahren (preconditioned conjugate gradient) benutzen. Dann wird das cg–Verfahren auf die Gleichung

$$BAx = Bb$$

angewendet, wobei $B \approx A^{-1}$ entspricht. Dabei wird der Wert für die Kondition in (11.4.16) zu $\kappa = \kappa_2(BA)$. Typische Wahlen von B sind z.B.

$$B = (D - \omega L)^{-1},$$

wie beim SSOR–Verfahren, wobei $A = D + L + U$ gelte. Weiterhin kann man aber auch eine „unvollständige“ Cholesky–Zerlegung der Matrix A derart durchführen, dass man nur ein Band um die Diagonale nach der Cholesky–Methode zerlegt, so dass $A \approx LL^T$ gilt, und dann

$$B = (LL^T)^{-1}$$

setzt. Dies Verfahren heißt ILU–Verfahren. Auf diese Art und Weise kann man die Anwendungsmöglichkeiten des Gradienten–/cg–Verfahrens auch auf Fälle erweitern, in denen die Matrix A nicht positiv definit ist.

Teil 2: Numerik gewöhnlicher Differentialgleichungen

Gewöhnliche Differentialgleichungen

Gewöhnliche Differentialgleichungen beschreiben die Dynamik eines Prozesses abhängig allein von einer Variablen. Dies ist in der Regel die Zeit.

1 Modellierung

Eine erste wichtige Klasse von Anwendungen liefern Populationsmodelle.

Beispiel 1.1 (Populationsmodelle aus [H] pp 465)**.** Sei $y = y(t)$ die Größe einer beliebigen Population zur Zeit t. Unsere Aufgabe ist es, $y(t)$ zu jedem Zeitpunkt zu beschreiben und zu berechnen. Die einfachste Annahme über die Änderung der Population ist, dass die Änderungsrate proportional zum gegenwärtigen Wert der Population ist, d.h.

$$y'(t) = \frac{d}{dt}y(t) = (g - s)y(t), \tag{1.2}$$

wobei $g > 0$ die Geburtenrate, $s > 0$ die Sterberate und $(g - s)$ die Proportionalitätskonstante ist. Falls $g - s > 0$ ist, befindet sich die Population im Wachstum. Falls $g - s < 0$ gilt, befindet sie sich im Schwund mit Zerfallsrate $g - s$. Wir haben es hier also mit exponentiellen Wachstum bzw. Zerfall zu tun. Zur Lösung des Problems benötigen wir eine Anfangsbedingung:

$$y(t_0) = y_0. \tag{1.3}$$

Die Lösung ist gegeben durch:

$$y(t) = y_0\, e^{(g-s)(t-t_0)}, \quad t > t_0\ .$$

Betrachten wir nun ein konkretes Beispiel:
Sei $y(t)$ die Größe der Weltbevölkerung im Jahr t. Schätzungen zufolge ist: $y(1961) = 3.06$ Milliarde und die Zuwachsrate ist $g - s = 0.02$.

Das Problem ist, dass dieses Modell nur für kleine Zeiträume gültig ist, da sonst für $g > s$ die Lösung stets exponentiell wachsen würde. Das liegt vor allem daran, dass hier keine beschränkten Ressourcen, wie z.B. Nahrung oder Platz berücksichtigt wurden. Für obige Werte gilt:

$$\begin{aligned} y(t+35) &= 3.06e^{0.02(t-t_0+35)} \\ &= e^{0.02\cdot 35}y(t) \\ &\approx 2.013 \cdot y(t) \end{aligned}$$

d.h. die Weltbevölkerung verdoppelt sich ca. alle 35 Jahre.

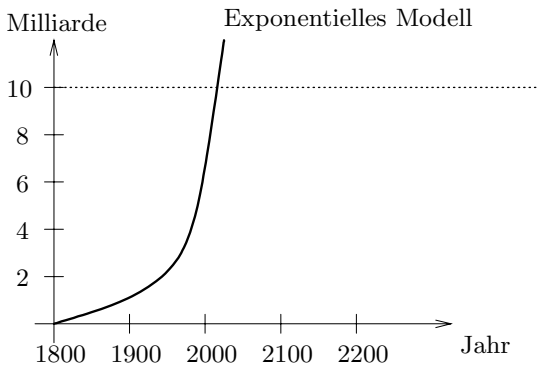


Abbildung 3: Evolution der Weltbevölkerung

Eine Verbesserung wird von Verhulst (1838) vorgeschlagen, hier soll nämlich die Sterberate s auch von der Größe der Population abhängen:

$$s = s(t) = s_0 + ay, \quad a > 0,$$

wobei s_0 die natürliche Sterberate ist und ay abhängig von der Größe der Population ist und Ressourcen berücksichtigt. Wir erhalten damit eine sogenannte *logistische Differentialgleichung*:

$$y'(t) = \frac{d}{dt}y(t) = (g - s_0 - ay(t))\, y(t), \quad y(t_0) = y_0\ . \tag{1.4}$$

Hierbei ist $ay(t) \cdot y(t)$ ein „Bremsterm". Dies ist ein Term höherer Ordnung in y, der für kleinere y wenig Unterschied ausmacht.

Übung: Prüfe nach, dass die Lösung von (1.4) gegeben ist durch:

$$y(t) = \frac{\frac{(g-s_0)}{a}}{1 - c \cdot e^{(g-s_0)(t-t_0)}},$$

wobei sich c aus $y(t_0) = y_0$ berechnen lässt.
Schätzungen zufolge lassen sich die Parameter setzen als

$$g - s_0 = 0.029, \qquad a = 2.941 \cdot 10^{-3}.$$

Da $g - s_0 > 0$, wächst die Population und strebt monoton gegen den Grenzwert $\lim_{t\to\infty} y(t) = \frac{g-s_0}{a}$.
Mit obigen Werten erhält man also:
$\frac{g-s_0}{a} \approx 10$ Milliarden.

Im Fall $g - s_0 < 0$ würde die Bevölkerung mit exponentieller Geschwindigkeit aussterben.

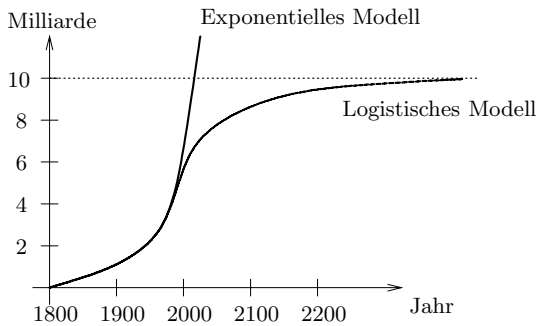


Abbildung 4: Vergleich der Lösung von (1.2) mit der Lösung des Modells (1.4) von Verhulst

Nach heutigen Erkenntnissen vermutet man, dass die Parameter $g - s_0$ und a zu niedrig gewählt sind.

Beispiel 1.5 (Räuber-Beute-Modell [H] pp. 468)**.** Bei diesem Modell sind zwei Populationen gegeben: Raubtiere und Beutetiere. Bezeichne mit $y_1(t)$ die Beutetierpopulation und mit $y_2(t)$ die Raubtierpopulation. Ein einfaches Modell wird durch das folgende System gewöhnlicher Differentialgleichungen beschrieben,

$$\begin{aligned} y_1' &= y_1(d_1 - a_1 y_1 - \ ry_2), \\ y_2' &= y_2(-d_2 + \ by_1 - a_2 y_2). \end{aligned}$$

Dabei sind a_1, a_2, d_1, d_2 und r nichtnegative Parameter, die die folgende Bedeutung haben: mit ry_2 wird die Anzahl der Beutetiere beschrieben, die von den Raubtieren gefressen werden. Da

diese Anzahl aber von der Anzahl der Raubtiere abhängt, muss der Faktor r mit y_2 multipliziert werden. Ähnlich ist dies in der zweiten Gleichung, dort beschreibt by_1 die Anzahl der Raubtiere, die bei mehr Beutetieren hinzukommen, also abhängig von y_2 ist.
Wir betrachten nun verschiedene Fälle in diesem Modell:
Wenn $y_2 = 0$ ist, also keine Raubtiere vorhanden sind, dann erhalten wir für die Beutetierpopulation y_1 das Modell (1.4) mit $d_1 = g - s_0 > 0$. Die Beutetiere leben also unabhängig von den Raubtieren.
Ist nun aber $y_1 = 0$, also keine Beute vorhanden, dann gilt für y_2 ebenfalls das Modell (1.4), wobei diesmal $d_2 = -(g - s_0) < 0$ gilt. Das bedeutet, dass die Raubtiere aussterben, da sie abhängig von der Beutetierpopulation sind.
Wenn $a_1 = a_2 = 0$ gilt, so lässt sich das obige Modell durch Division in die folgende Differentialgleichung:

$$\frac{dy_2}{dy_1} = \frac{by_1 - d_2}{y_1} \frac{y_2}{d_1 - ry_2}$$

überführen. Weiter erhält man durch Trennung der Variablen die implizite Lösung

$$y_1^{d_2} y_2^{d_1} = c\, e^{by_1} e^{ry_2}$$

wobei c aus den Anfangsdaten berechnet wird. Es lässt sich zeigen, dass die Lösungsmenge eine geschlossene Kurve in der y_1/y_2 -Ebene bildet, siehe Abbildung 5. Abbildung 6 zeigt die Dynamik der beiden Populationen in Abhängigkeit von der Zeit.

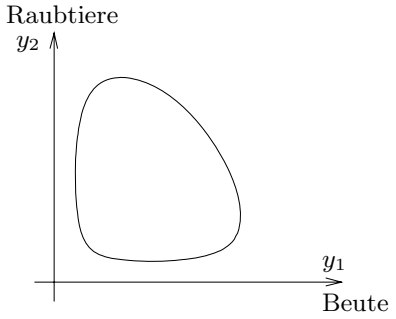


Abbildung 5: Lösungsmenge in der y_1/y_2 -Ebene

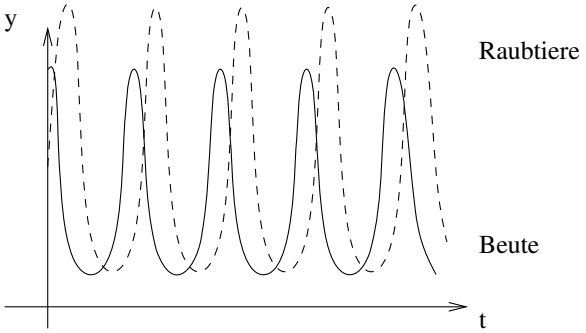


Abbildung 6: Dynamik der Populationen

Ist $a_1, a_2 \neq 0$ lassen sich zwei Fälle unterscheiden:
Für $\frac{d_2}{b} > \frac{d_1}{a_1}$ ist $\lim_{t\to\infty} y_2(t) = 0$, d.h. die Sterberate der Raubtiere ist zu groß, um durch das Nahrungsangebot kompensiert zu werden.
Für $\frac{d_2}{b} < \frac{d_1}{a_1}$ ergibt sich eine positive Lösung für beide Populationen $y_1, y_2 > 0$:

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} a_1 & r \\ b & -a_2 \end{pmatrix}^{-1} \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = \frac{1}{a_1 a_2 + br} \begin{pmatrix} a_2 d_1 + r d_2 \\ b d_1 - a_1 d_2 \end{pmatrix} .$$

Dies ist eine konstante (stationäre) Lösung der Räuber–Beute–Gleichung. Für positive Anfangsdaten streben beide Populationen $y_1(t)$ und $y_2(t)$ für $t \to \infty$ gegen diese Lösung. Diese Lösung entspricht einem Gleichegewichtzustand, berechnet aus der Differentialgleichung für $y_1' \equiv 0$, siehe Abbildung 8. Die Lösungsmenge ist in Abbildung 7 in der y_1/y_2 -Ebene eingetragen.

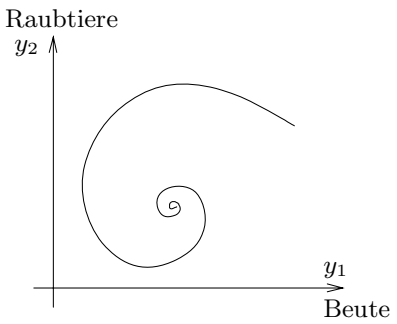


Abbildung 7: Lösungsmenge in der y_1/y_2 -Ebene

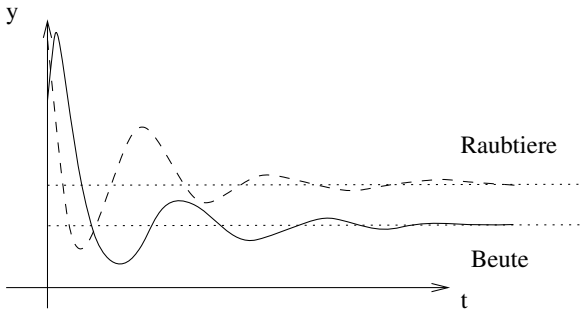


Abbildung 8: stationäre Lösung der Räuber–Beute–Gleichung

Beispiel 1.6 (Einfaches HIV-Modell [H] p. 471)**.** Allgemeines zur Virusdynamik: Ein Virus benötigt eine Wirtszelle zur Reproduktion. Ein einfaches Modell zur Simulation des Krankheitsverlaufs besteht aus drei Populationen:

$$\begin{aligned} x &= x(t) && \text{Anzahl der nichtinfizierten Zellen,} \\ y &= y(t) && \text{Anzahl der infizierten Zellen,} \\ v &= v(t) && \text{Anzahl der freien Viruspartikel.} \end{aligned}$$

Deren Änderungsverhalten kann durch die folgenden Gleichungen beschrieben werden:

$$\begin{aligned} x'(t) &= p - s_x x - r_x x v(t) + g(v)x, \\ y'(t) &= r_x x v - s_y y - g(v)y, \\ v'(t) &= n g(v) y - s_v v - (r_x + r_v) x v + f(v). \end{aligned}$$

Hierbei gilt, dass p eine konstante Rate ist, s_x, s_y und s_v die Sterberaten der Population x, y und v sind, und $r_x v$ beschreibt die vom Virus neu infizierten Zellen. Der Term $r_v x v$ modelliert die Virenabnahme aufgrund der körpereigenen Abwehrkräfte. Die Funktion $g(v)$ beschreibt die Zellteilungsrate der Zellen zur Bekämpfung der Viren. Daher bedeutet $-g(v)y$, dass infizierte Zellen zerstört werden und n neue Viruspartikel produziert werden. Außerdem soll $ng(v)y$ beschreiben, dass n neue, durch Zellteilung produzierte Viruspartikel entstehen. Der Ausdruck $f(v)$ soll die weitere Virenproduktion ausdrücken.
Weiterhin werden bestimmte Bedingungen an die Zellteilungsrate $g(v)$ gestellt:

- Es gilt: $g(0) = 0$, d.h. es gibt keine Zellteilung, wenn keine Viren vorhanden sind.
- g ist monoton wachsend, also je mehr Viren vorhanden sind, desto größer ist die Zellteilungsrate.
- Es gibt eine maximale Rate g_∞, die $g(v)$ beschränkt, es gilt also $g(v) < g_\infty$.

In einem einfachen Modell wählt man $g(v) := \frac{v}{v_g+v} g_\infty$ mit einem Parameter $v_g > 0$.
Obwohl das gesamte Modell sehr einfach erscheint, beschreibt es bereits recht gut die Dynamik von komplexen Immunreaktionen. Es liefert eine sehr gute Übereinstimmung der numerischen Simulation mit dem in der Realität beobachteten Krankeitsverlauf.
Nun wählt man die Parameter wie folgt.

Die Zeiteinheit wird in Tagen gemessen, und x,y und v geben die Zellenzahl pro mm^3 an. Die Anfangsdaten seien

$$\begin{aligned} x(t_0) &= 10^3 \\ y(t_0) &= 0 \\ v(t_0) &= 10. \end{aligned}$$

$s_x = 0.02$	$p = 10$	$n = 1000$	$f(v) = \frac{20v}{v+1}$
$s_y = 0.256$	$r_x = 2.4 \cdot 10^{-5}$	$g_\infty = 0.01$	
$s_v = 0$	$r_v = 7.4 \cdot 10^{-4}$	$v_g = 100$	

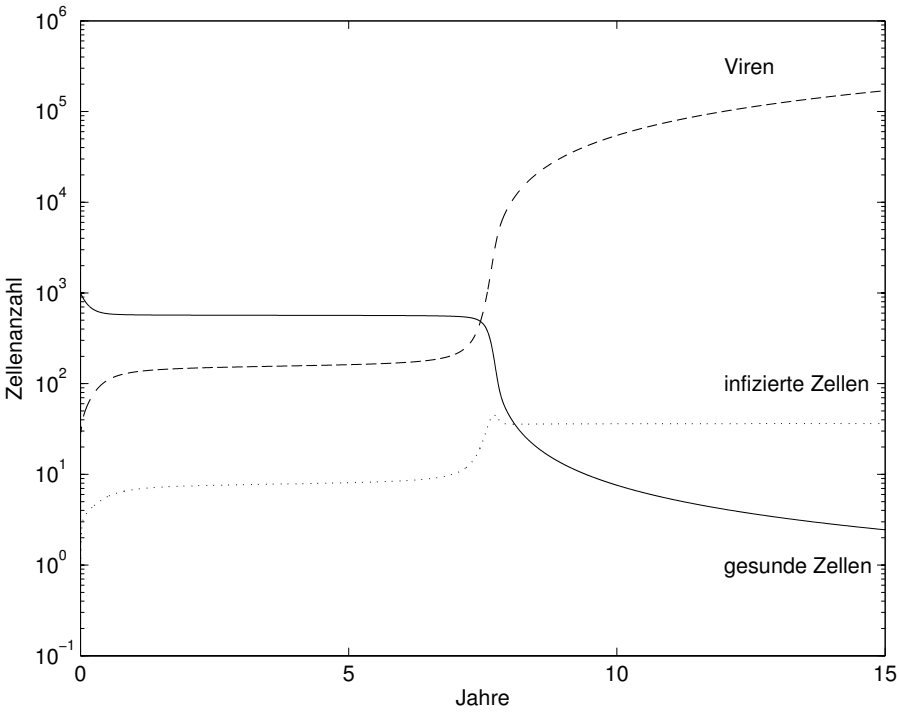


Abbildung 9: Ergebnis der numerischen Simulation

Das Ergebnis einer numerischen Lösung des Systems gewöhnlicher Differentialgleichungen sieht man in Abbildung 9. Das Virus verharrt nach der Infektion ca. 5-10 Jahre im *Lauerzustand*, während der Körper jedoch HIV-infiziert ist, bevor plötzlich AIDS ausbricht. Bis zu diesem Ausbruch scheint das System im Gleichgewicht zu sein. Diese plötzliche Veränderung ist bisher nicht erklärbar.
Näheres zu diesem Modell und Verallgemeinerungen findet man in [NM].

2 Theoretische Grundlagen

Im vorangegangenen Abschnitt haben wir Systeme gewöhnlicher Differentialgleichungen erster Ordnung zur Beschreibung von Populationsdynamik kennengelernt. Doch auch viele andere Prozesse in den Naturwissenschaften lassen sich über Differentialgleichungen modellieren. Ein weiteres wichtiges Beispiel aus der Physik führt auf eine Differentialgleichung zweiter Ordnung.

Beispiel 2.1 (Freier Fall)**.** Ein Körper der Masse m werde zur Zeit $t = 0$ aus der Höhe $H = s(0)$ aus einem Ruhezustand ($s'(0) = 0$) losgelassen. Dann ist die Höhe $s(t)$ für $t > 0$ als Lösung der *Anfangswertaufgabe (AWA) zweiter Ordnung*

$$s'' = g(s(t)) - \frac{1}{m} R(s(t), v(t), t), \ t > 0,$$

gegeben. Dabei ist $g(s(t))$ die Erdbeschleunigung in der Höhe s, R die Reibungskraft und $v(t) = s'(t)$ die Geschwindigkeit zur Zeit t. Die Anfangsbedingungen sind hier $s(0) = H, \ s'(0) = 0$.

■

Eine weitere wichtige Klasse zur Beschreibung dynamischer Prozesse bilden Randwertaufgaben, siehe Kapitel 6.

Diese Vorlesung behandelt numerische Verfahren zur Lösung von Anfangs– und Randwertaufgaben. Wir beginnen mit Anfangswertaufgaben.

Beobachte zunächst, dass sich die Anfangswertaufgabe zweiter Ordnung aus Beispiel 2.1 als *System erster Ordnung* schreiben lässt. Setze dazu

$$\begin{aligned} y_1(t) &:= s(t) \\ y_2(t) &:= s'(t). \end{aligned}$$

Dann erhält die Anfangswertaufgabe aus Beispiel 2.1 die Form

$$\begin{aligned} y_1' &= y_2 \\ y_2' &= g(y_1) - \frac{1}{m} R(y_1, y_2, t) \end{aligned}$$

mit $y_1(0) = H$ und $y_2(0) = 0$. Dies lässt sich folgendermaßen verallgemeinern:

Bemerkung. Die Anfangswertaufgabe n–ter Ordnung

$$\begin{aligned} u^{(n)} &= g(u, u', \ldots, u^{(n-1)}, t) \\ u^{(j)}(t_0) &= u_j, \quad j = 0, \ldots, n-1, \end{aligned}$$

mit gegebenen Anfangswerten $u_0, ..., u_{n-1}$, lässt sich äquivalent als System erster Ordnung

$$\begin{aligned} y' &= f(t, y), \qquad t \geq t_0, \\ y(t_0) &= y^0, \end{aligned}$$

schreiben, wobei $y : \mathbb{R} \to \mathbb{R}^n$ (Kurve im $\mathbb{R}^n$), $y(t) := (y_1(t), \ldots, y_n(t))^T$, $y_0 \in \mathbb{R}^n$ und $f : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$ sind mit

$$y_1 = u, \quad y_2 = u', \quad \ldots \quad y_n = u^{(n-1)}$$

und

$$\begin{aligned} f_1(t,y) &:= y_2(t) \\ f_2(t,y) &:= y_3(t) \\ &\vdots \\ f_{n-1}(t,y) &:= y_n(t) \\ f_n(t,y) &:= g(y_1, y_2, \ldots, y_n, t) \\ (y^0)_j &:= u_{j-1}, \quad j = 1, \ldots, n. \end{aligned}$$

Es reicht daher, folgende *allgemeine Anfangswertaufgabe erster Ordnung* zu betrachten:

$$\begin{aligned} y' &= f(t,y), \qquad t \geq t_0, \\ y(t_0) &= y^0 \end{aligned} \tag{2.2}$$

mit gegebenem $y^0 \in \mathbb{R}^n$ und $f : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$. Gesucht ist die Lösung $y : \mathbb{R} \to \mathbb{R}^n$. ■

Solche Systeme rühren nicht nur von der Reduktion von Systemen höherer Ordnung oder von der Modellierung von n Populationen her. Insbesondere treten sie bei der Diskretisierung chemischer Reaktionen auf. Eine weitere wichtige Klasse entsteht, wenn man die Ortsvariable in zeitabhängigen *partiellen* Differentialgleichungen diskretisiert: hier sei als Beispiel die Diffusionsgleichung $\frac{\partial}{\partial t}u - \Delta u = f$ angegeben. Hier ist $u = u(x,t)$ gesucht.

Die *Existenz* von Lösungen der Anfangswertaufgabe (2.2) lässt sich unter sehr allgemeinen Voraussetzungen der folgende Satz zeigen, siehe etwa [H] oder [Wa].

Existenzsatz von Peano. Sei der Phasenraum $\Omega \subset \mathbb{R} \times \mathbb{R}^n$ offen. Für jedes $f \in \mathcal{C}^0(\Omega, \mathbb{R}^n)$ existiert mindestens eine Lösung von (2.2) in einer Umgebung von $(t_0, y^0) \in \Omega$.

Beweisidee: Der Beweis läuft über eine einfache Diskretisierungsmethode (Euler–Verfahren, siehe Abschnitt 3.1). Man zeigt die gleichgradige Stetigkeit der daraus resultierenden Folge von Näherungslösungen. Der „Satz von Arzela–Ascoli“ liefert dann eine konvergente Teilfolge.

■

Anfangswertaufgaben sind in dem Sinne gutartig, dass bereits die Stetigkeit der rechten Seite ausreichend für die Existenz einer Lösung ist. Ist f sogar Lipschitz–stetig, lassen sich *Existenz* und *Eindeutigkeit* beweisen, siehe dazu [H] oder [Wa].

Satz von Picard–Lindelöf. Sei $f : \Omega \to \mathbb{R}$ stetig auf dem Kompaktum $K := \{(t,y) : \ t \in [t_0, t_0 + a],\ y^0 \in \Omega,\ \|y - y^0\| \leq b\}$ und genüge dort bzgl. y der Lipschitz–Bedingung

$$\|f(t,y^1) - f(t,y^2)\| \leq L\|y^1 - y^2\|, \tag{2.L}$$

wobei $\|\cdot\|$ eine beliebige, aber feste Norm auf $\mathbb{R}^n$ sei. Sei $M := \|f\|_\infty(K),\ \alpha := \{a, \frac{b}{M}\}$. Dann hat die Anfangswertaufgabe (2.2) eine eindeutige Lösung auf $[t_0, t_0 + \alpha]$.
Eine analoge Aussage gilt für $t \in [t_0 - \alpha, t_0]$. ■

Abgesehen von Existenz und Eindeutigkeit ist es für die Güte numerischer Verfahren *essentiell*, dass die Lösung *stetig* von den Anfangsdaten abhängt. Entsprechende Stabilitätsaussagen beruhen häufig auf der Gronwall'schen Ungleichung.

Satz (Ungleichung von GRONWALL**)** Seien u, v stückweise stetig und nichtnegativ auf dem Intervall $[t_0, t_0 + a]$ und $C \geq 0$ eine Konstante. Sei weiter

$$v(t) \leq C + \int_{t_0}^{t} u(s)v(s)ds \tag{2.3}$$

für alle $t \in [t_0, t_0 + a]$. Dann gilt

$$v(t) \leq Ce^{\int_{t_0}^{t} u(s)ds} \tag{2.4}$$

für alle $t \in [t_0, t_0 + a]$. ∎

Motivation von (2.4): Betrachte die Differentialgleichung $v' = uv$ mit dem Startwert $v(t_0) = C$. Die Lösung ist

$$v(t) = Ce^{\int_{t_0}^{t} u(s)ds},$$

denn

$$v' = Ce^{\int_{t_0}^{t} u(s)ds} u(t) = vu$$

und $v(t_0) = Ce^0 = C$. Andererseits lautet die Integraldarstellung der Lösung

$$\begin{aligned} v(t) &= v(t_0) + \int_{t_0}^{t} u(s)v(s)ds \\ &= C + \int_{t_0}^{t} u(s)v(s)ds. \end{aligned}$$

Beweis der GRONWALL**'schen Ungleichung:** Sei zunächst $C > 0$. Setze

$$V(t) := C + \int_{t_0}^{t} u(s)v(s)ds.$$

Aus (2.3) folgt: $v(t) \leq V(t)$ und $0 < C \leq V(t)$ für alle $t \in [t_0, t_0 + a]$. Es ist $V'(t) = u(t)v(t) \leq u(t)V(t)$ und somit

$$\frac{V'(t)}{V(t)} \leq u(t).$$

Daraus folgt durch Integration

$$\begin{aligned} \log \frac{V(t)}{C} &= \log V(t) - \log V(t_0) = \log V(s)\big|_{t_0}^{t} \\ &= \int_{t_0}^{t} \frac{V'(s)}{V(s)} ds \ \leq \int_{t_0}^{t} u(s)ds, \end{aligned}$$

was

$$\frac{V(t)}{C} \leq e^{\int_{t_0}^{t} u(s)ds} \quad \text{oder} \quad V(t) \leq Ce^{\int_{t_0}^{t} u(s)ds}$$

impliziert. Also gilt wegen $v(t) \leq V(t)$

$$v(t) \leq Ce^{\int_{t_0}^{t} u(s)ds}, \qquad t \in [t_0, t_0 + a].$$

Sei nun $C = 0$. Gilt (2.3) für $C = 0$, dann auch für jedes $\tilde{C} > 0$, also

$$v(t) \leq \tilde{C}e^{\int_{t_0}^{t} u(s)ds} \to 0$$

für $\tilde{C} \to 0$. ∎

Stetige Abhängigkeit von den Anfangsdaten: Wie im Satz von PICARD–LINDELÖF sei $f : \Omega \to \mathbb{R}$ stetig auf dem Kompaktum $K := \{(t, y) : \ t \in [t_0, t_0 + a], \ y^0 \in \Omega, \ \|y - y^0\| \leq b\}$ und genüge dort bzgl. y der LIPSCHITZ–Bedingung (2.L) mit L. Dann existiert nach dem Satz von PICARD–LINDELÖF eine eindeutige Lösung $y(t) = y(t; z)$ von

$$y' = f(t, y), \quad y(t_0) = z, \quad z \in \Omega.$$

Satz 2.5. Unter obigen Voraussetzungen gilt

$$\|y(t; y^0) - y(t; y^1)\| \leq e^{L|t-t_0|}\|y^0 - y^1\|$$

für $y^0, \ y^1 \in K$, d.h. die Differenz der Lösungen der Anfangswertaufgabe (2.2) bezüglich zweier verschiedener Anfangswerte lässt sich durch die Differenz der Anfangswerte abschätzen.

Beweis: Die Integraldarstellung der Lösung von (2.2) lautet

$$y(t; z) = z + \int_{t_0}^{t} f(s, y(s; z))ds.$$

Dies impliziert für Anfangswerte $y^0, y^1 \in K$

$$y(t; y^0) - y(t; y^1) = y^0 - y^1 - \int_{t_0}^{t} \left[f(s, y(s; y^1)) - f(s, y(s; y^0))\right] ds$$

sowie mit (2.L)

$$\begin{aligned} \|y(t; y^0) - y(t; y^1)\| &\leq \|y^0 - y^1\| + \int_{t_0}^{t} \|f(s, y(s; y^1)) - f(s, y(s; y^0))\|ds \\ &\leq \|y^0 - y^1\| + \int_{t_0}^{t} L\|y(s; y^1) - y(s; y^0)\|ds. \end{aligned}$$

Mit $v(t) := \|y(t; y^0) - y(t; y^1)\|, \ u(t) := L$ und $C := \|y^0 - y^1\|$ folgt die Behauptung sofort aus der GRONWALL'schen Ungleichung. ∎

Bemerkung 2.6. Die Abschätzung in Satz 2.5 ist scharf, d.h. Gleichheit wird erreicht. Betrachte etwa $y' = Ly$ mit $y(0) = y^0$ und $t_0 = 0$. Dann ist $y(t) = y^0 e^{Lt}$ und

$$y(t; y^0) - y(t; y^1) = e^{Lt}(y^0 - y^1).$$

∎

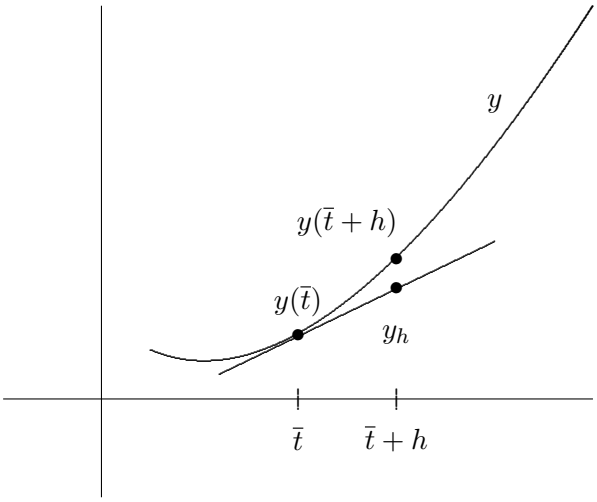
3 Einschrittverfahren zur Lösung von Anfangswertaufgaben

3.1 Die Euler'sche Polygonzugmethode

Jede Lösung $y(t)$ von $y' = f(t, y)$ zeichnet sich gerade dadurch aus, dass der Graph von y an jedem Punkt $(t, y(t))$ die Steigung $f(t, y(t))$ hat. Es liegt daher nahe, die Lösung $y(t; \bar{t}, y(\bar{t}))$ in einer Umgebung von $\bar{t}$ dadurch zu approximieren, dass man der Tangente durch $(\bar{t}, y(\bar{t}))$ bis $\bar{t} + h$ folgt. Der Ausdruck

$$y_h = y(\bar{t}) + hf(\bar{t}, y(\bar{t})) \tag{3.1.1}$$

ist somit eine Approximation an $y(\bar{t} + h)$. Die EULER'sche Polygonzugmethode besteht darin, dieses Verfahren mit dem neuen Startwert y_h zu wiederholen. Formal geht man dabei folgendermaßen vor.



Diskretisierung von (2.2): Zur näherungsweisen Berechnung der Lösung von y an einer Stelle $t > t_0$, wähle eine Folge

$$\{t_j\}_{j=0}^m, \quad t_m = t, \quad h_j = t_{j+1} - t_j$$

und setze

$$y^{j+1} := y^j + h_j f(t_j, y^j), \quad j = 0, \ldots, m-1. \tag{3.1.2}$$

Die Lösung wird dann durch einen Polygonzug durch die Werte $y(t_j) = y_j$, $j = 0, .., m$ approximiert.

Lokaler Abbruchfehler: Ein erstes Maß für das Funktionieren eines solchen Vorgehens ist die Abweichung der Näherungslösung y_h von $y(\bar{t}+h)$ nach einem Schritt (3.1.1) in Abhängigkeit von h. Dazu verwenden wir TAYLOR–Entwicklung von $y(\bar{t} + h)$ um $\bar{t}$,

$$y(\bar{t} + h) = y(\bar{t}) + hy'(\bar{t}) + O(h^2) = y(\bar{t}) + hf(\bar{t}, y(\bar{t})) + O(h^2). \tag{3.1.3}$$

Mit (3.1.1) folgt daraus

$$y(\bar{t} + h) - y_h = O(h^2). \tag{3.1.4}$$

Gleichung (3.1.3) lässt sich auch folgendermaßen interpretieren: Setzt man die exakte Lösung (mit korrektem Startwert $y(\bar{t})$) in das Verfahren (3.1.2) ein, so ergibt sich nach *einem* Schritt ein Fehler der Ordnung h^2.

3.2 Einschrittverfahren

Die EULER'sche Polygonzugmethode (3.1.3) ist ein Spezialfall des Ansatzes

$$y^{j+1} = y^j + h_j \Phi(t_j, t_{j+1}, y^j, y^{j+1}) \tag{3.2.1}$$

für $j = 0, 1, \ldots$ mit $y^0 = y(t_0) \in \mathbb{R}^n$. Hierbei ist $h_j = t_{j+1} - t_j$ die *Schrittweite*, und Φ heißt *Verfahrens–* oder *Inkrementfunktion*. Das Verfahren (3.2.1) heißt *Einschrittverfahren (ESV)*, da zur Berechnung von y^{j+1} nur auf y^j zurückgegriffen wird.

Falls Φ nicht von y^{j+1} abhängt, lässt sich y^{j+1} einfach durch Einsetzen vorher berechneter Größen ermitteln *(explizites ESV)*. Sonst heißt das ESV *implizit*. Bei impliziten Verfahren hat man im Allgemeinen pro Schritt ein *nichtlineares* Gleichungssystem zu lösen.

Durch geeignete Wahl von Φ versucht man die Genauigkeit (im Vergleich zu dem einfachen EULER'schen Polygonzugverfahren) zu erhöhen. Ein möglicher Ansatz, Φ zu entwickeln, liegt in der Beobachtung, dass die exakte Lösung $y(t) = y(t; t_j, y^j)$ von (3.1.2) durch den Punkt (t_j, y^j) durch

$$y(t_{j+1}) = y^j + \int_{t_j}^{t_{j+1}} f(s, y(s))\, ds \tag{3.2.2}$$

gegeben ist. Nun liegt es nahe, auf geeignete Quadraturformeln für das Integral zurückzugreifen. Einige Standardverfahren sind hier mit ihrer Ordnung aufgelistet. Den Begriff der Ordnung werden wir weiter unten definieren.

Beispiel 3.2.3.

(i) EULER'sches Polygonzugverfahren: (explizites EULER–Verfahren) basiert auf der Rechteckregel

$$\Phi(t_j, t_{j+1}, y^j, y^{j+1}) = f(t_j, y^j);$$

und hat die Ordnung 1.

(ii) Implizites EULER–Verfahren:

$$\Phi(t_j, t_{j+1}, y^j, y^{j+1}) = f(t_{j+1}, y^{j+1});$$

dieses hat auch die Ordnung 1.

(iii) Implizite Trapezregel mit Ordnung 2:

$$\Phi(t_j, t_{j+1}, y^j, y^{j+1}) = \frac{1}{2}\left(f(t_j, y^j) + f(t_{j+1}, y^{j+1})\right);$$

(iv) Implizite Mittelpunktsregel:, ebenfalls mit Ordnung 2:

$$\Phi(t_j, t_{j+1}, y^j, y^{j+1}) = f\left(\frac{1}{2}(t_j + t_{j+1}), \frac{1}{2}(y^j + y^{j+1})\right);$$

(v) Später werden wir noch das RUNGE-KUTTA-Verfahren und das Verfahren von HEUN kennenlernen, die beide von der Ordnung 2 sind:

$$\Phi(t_j, t_{j+1}, y^j, y^{j+1}) = \frac{1}{2} f(t_j, y^j) + f(t_{j+1}, y^j + hf(t_j, y^j)).$$

■

Wie beim EULER–Verfahren sollen nun entsprechende Genauigkeitskriterien formuliert werden. Im Folgenden sei die *lokal exakte* Lösung von (2.2) diejenige Lösung $y(t) = y(t; y^j)$, für die $y(t_j) = y^j$ gilt.

Lokaler Abbruchfehler (Diskretisierungsfehler): Wir untersuchen die *Abweichung* der lokal exakten Lösung von der durch das Verfahren berechneten Lösung

$$y(t_{j+1}) - y^{j+1} =: h_j \tau(h_j). \tag{3.2.4}$$

Einsetzen der Verfahrensvorschrift (3.2.1) liefert

$$h_j \tau(h_j) = y(t_{j+1}) - y^{j+1} = y(t_{j+1}) - y^j - h_j \Phi(t_j, t_{j+1}, y^j, y^{j+1}),$$

was nach Division durch die Schrittweite

$$\tau(h_j) = \frac{y(t_{j+1}) - y^j}{h_j} - \Phi(t_j, t_{j+1}, y^j, y^{j+1}) \tag{3.2.5}$$

impliziert.

Bemerkung. Das Maß $\tau(h_j) = \tau(t_j, t_{j+1}, y^j, y^{j+1})$ für den lokalen Abbruchfehler ergibt sich gerade durch *Einsetzen* der *lokal exakten Lösung in das Verfahren.* Eine Abschätzung (der Ordnung) von $\tau(h_j)$ wird in der Regel über TAYLOR–Entwicklung hergeleitet. Um diese anwenden zu können, muss die Lösung selbstverständlich als entsprechend regulär vorausgesetzt werden. ■

Definition 3.2.6. Das Verfahren (3.2.1) heißt *konsistent*, wenn $\tau(h) \to 0$ für $h \to 0$ gilt. Ist allgemeiner

$$\tau(h) = O(h^p) \quad \text{für} \quad h \to 0, \tag{3.2.7}$$

so heißt das Verfahren (3.2.1) *konsistent von der Ordnung p.* ■

Beispiel: Für das explizite EULER–Verfahren in (3.1.3) und 3.2.3 (i) gilt

$$y^{j+1} = y^j + h_j f(t_j, y^j).$$

Nach (3.1.4) folgt mit TAYLOR–Entwicklung wegen $y(t_j) = y^j$ (lokal exakte Lösung)

$$y(t_{j+1}) = y(t_j) + hf(t_j, y(t_j)) + O(h^2) = y^j + hf(t_j, y^j) + O(h^2),$$

was eingesetzt in (3.2.4)

$$\tau(h) = \frac{y(t_{j+1}) - y^j}{h} - f(t_j, y^j) = \frac{y^j + hf(t_j, y^j) - y^j}{h} - f(t_j, y^j) + O(h) = O(h)$$

ergibt. Demnach ist das explizite EULER–Verfahren *konsistent von der Ordnung* 1. ■

3.3 Konvergenz von Einschrittverfahren

Die Tatsache, dass ein Verfahren *konsistent* ist, impliziert zunächst nicht seine Konvergenz, da sich Fehler im Laufe der Rechnung akkumulieren können. Es bezeichne $y(t)$ die Lösung von (2.2) und $y_h(t)$ eine nach einem Einschrittverfahren (3.2.1) berechnete Approximation, d.h. $y_h(t_j) = y^j$, $j = 0, \ldots, N$, $t_j \in \{t_0, \ldots, t_N = T\}$. *Konvergenz* bedeutet, dass für den *globalen Diskretisierungsfehler*

$$\varepsilon(t, h) := y_h(t) - y(t) \to 0, \quad h \to 0 \tag{3.3.1}$$

oder mit $t_n = T$

$$\|\varepsilon(t, h)\| := \|\varepsilon(t, h)\|_\infty := \max_{j=0,\ldots,N} \|y^j - y(t_j)\| \to 0, \quad N \to \infty$$

gilt. Da h nur diskrete Werte durchläuft, heißt das

$$\varepsilon(t, h_\ell) \to 0, \quad \ell \to \infty,$$

wobei $h_\ell = (T - t_0)/\ell$ bezeichne. Konvergenz bedeutet also, dass sich bei Verkleinerung der Schrittweite die approximative Lösung $y_h(t)$ der exakten immer mehr annähert.

Wir werden sehen, dass man, um Konvergenz zu zeigen, zusätzlich zu der Konsistenz noch die *Stabilität* des Verfahrens benötigt. Eine genaue Definition hiervon wird später folgen. Wir benötigen als *Vorbereitung* zunächst ein diskretes Analogon der GRONWALL'schen Ungleichung aus Kapitel 2.

Lemma (Diskrete GRONWALL–Ungleichung) Für $\delta_i > 0$, $\eta_i \geq 0$ und $z_i \geq 0$ gelte

$$z_{m+1} \leq (1 + \delta_m) z_m + \eta_m \tag{3.3.2}$$

für $m = 0, 1, 2, \ldots$. Dann gilt

$$z_{m+1} \leq \left(z_0 + \sum_{j=0}^{m} \eta_j \right) e^{\sum_{j=0}^{m} \delta_j}. \tag{3.3.3}$$

Falls $\delta_j = \delta$ und $\eta_j = \eta$, gilt insbesondere

$$z_m \leq z_0 e^{m\delta} + \eta \begin{cases} \frac{e^{m\delta}-1}{\delta}, & \text{falls } \delta > 0 \\ m, & \text{falls } \delta = 0 \end{cases}. \tag{3.3.4}$$

Beweis: Aus (3.3.2) folgt

$$\begin{aligned} z_{m+1} &\leq z_m + \delta_m z_m + \eta_m \\ &\leq z_{m-1} + \delta_{m-1} z_{m-1} + \eta_{m-1} + \delta_m z_m + \eta_m \\ &\vdots \\ &\leq z_0 + \sum_{j=0}^{m} \eta_j + \sum_{j=0}^{m} \delta_j z_j. \end{aligned}$$

Setze nun $C := z_0 + \sum_{j=0}^{m} \eta_j$, $v(\cdot)\big|_{[j,j+1)} := z_j$ und $u(\cdot)\big|_{[j,j+1)} := \delta_j$. Dann entspricht die letzte der obigen Gleichungen gerade (2.5), und es gilt

$$\begin{aligned} v(m+1) = v(\cdot)\Big|_{[m+1,m+2)} &\leq C + \sum_{j=0}^{m} u(\cdot)\Big|_{[j,j+1)} v(\cdot)\Big|_{[j,j+1)} \\ &= C + \int_0^{m+1} u(s) v(s) ds. \end{aligned}$$

Mit der kontinuierlichen GRONWALL–Ungleichung folgt nun

$$v(m+1) \leq C e^{\int_0^{m+1} u(s) ds},$$

also

$$z_{m+1} \leq \left(z_0 + \sum_{j=0}^{m} \eta_j \right) e^{\sum\limits_{j=0}^{m} \delta_j},$$

womit (3.3.3) gezeigt ist. Bleibt (3.3.4) zu beweisen. Mit $\delta_j = \delta$ und $\eta_j = \eta$ folgt aus (3.3.2)

$$\begin{aligned}
z_1 &\leq (1+\delta)z_0 + \eta \\
z_2 &\leq (1+\delta)z_1 + \eta \leq (1+\delta)^2 z_0 + \eta((1+\delta)+1) \\
&\vdots \\
z_m &\leq (1+\delta)^m z_0 + \eta(1 + (1+\delta) + \ldots + (1+\delta)^{m-1}) \\
&= (1+\delta)^m z_0 + \eta\left(\frac{(1+\delta)^m - 1}{\delta}\right).
\end{aligned}$$

Mit der Abschätzung $1 + x \leq e^x$ folgt für $\delta > 0$

$$z_m \leq e^{\delta m} z_0 + \eta\left(\frac{e^{\delta m} - 1}{\delta}\right).$$

Wegen

$$\lim_{\delta \to 0} \frac{e^{\delta m} - 1}{\delta} = m$$

ergibt sich (3.3.4). ∎

Aus der Lipschitz–Stetigkeit der Funktion f folgt, dass eine eindeutige Lösung der Anfangswertaufgabe existiert. Daher ist es naheliegend, dies bei einer Konvergenzanalyse auch für die Verfahrensfunktion Φ zu fordern. Ab jetzt sei stets vorausgesetzt, dass f stetig auf $K = \{(t,y) : \ t \in [t_0, t_0 + a], \ \|y - y^0\| \leq b, \ y^0 \in \Omega\}$ ist und dort bzgl. y der Lipschitz–Bedingung

$$\|f(t,y^0) - f(t,y^1)\| \leq L\|y^0 - y^1\| \tag{2.L}$$

mit $y^0, y^1 \in K$ genügt. Nach dem Satz von Picard–Lindelöf besitzt die Anfangswertaufgabe $y' = f(t,y)$ mit $y(t_0) = y^0$ in diesem Fall eine eindeutige Lösung $y(t)$, die stetig von y^0 abhängt.

Satz 3.3.5. Die Verfahrensfunktion Φ in der expliziten Fassung von (3.2.1)

$$y^{j+1} = y^j + h_j \Phi(t_j, y^j)$$

sei stetig auf $\mathbb{R}^n$ und genüge dort bzgl. y einer Lipschitz–Bedingung

$$\|\Phi(t,v^1) - \Phi(t,v^2)\| \leq M\|v^1 - v^2\|, \quad v^1, v^2 \in \mathbb{R}^n. \tag{3.3.6}$$

Dann gilt für den *globalen Diskretisierungsfehler*

$$\|\varepsilon(t,h)\| \leq (\|\varepsilon(t_0,h)\| + (t - t_0)\|\tau(h)\|)\, e^{M(t-t_0)}, \tag{3.3.7}$$

wobei $t = t_0 + \sum_{j=0}^{N-1} h_j$ und $t = t_N$ sind. Hierbei bezeichnet $\varepsilon(t_0,h) := y^0 - y(t_0)$ eine eventuell vorhandene Störung im Startwert und $\|\tau(h)\| := \max\limits_{j=0,\ldots,N} \|\tau(h_j)\|$.

Bemerkung 3.3.8. Ist das Verfahren konsistent von der Ordnung p, d.h. nach Definition 3.2.6 $\tau(h) = O(h^p)$, so folgt mit (3.3.7) wegen $\|\tau(h)\| = |h^p|$, dass die Konvergenzordnung der Konsistenzordnung entspricht. ∎

Es reicht für die Aussage des obigen Satzes aus, anstelle von (3.3.6) eine *lokale* LIPSCHITZ–Bedingung bzgl. Φ zu fordern (vergl. [SB2]). Dabei argumentiert man über die Konstruktion einer Fortsetzung von Φ auf $\mathbb{R}^n$, die die Bedingung (3.3.6) erfüllt.

Beweis von Satz 3.3.5: Nach Definition des lokalen Abbruchfehlers (3.2.4) gilt

$$y(t_{j+1}) = y(t_j) + h_j\Phi(t_j, y(t_j)) + h_j\tau(h_j, y(t_j)),$$

wobei wir jetzt deutlicher $\tau(h_j, y(t_j))$ anstelle von $\tau(h_j)$ schreiben, um auf die Abhängigkeit von $y(t_j)$ hinzuweisen. Das Verfahren in (3.2.1) lautet

$$y^{j+1} = y^j + h_j\Phi(t_j, y^j).$$

Differenzbildung liefert

$$y(t_{j+1}) - y^{j+1} = y(t_j) - y^j + h_j(\Phi(t_j, y(t_j)) - \Phi(t_j, y^j)) + h_j\tau(h_j, y(t_j)).$$

Für $\varepsilon_j := \|y(t_j) - y^j\|$ und $\tau_j := \|\tau(h_j, y(t_j))\|$ ergibt sich also mit (3.3.6)

$$\begin{aligned} \varepsilon_{j+1} &\leq \varepsilon_j + h_j\|\Phi(t_j, y(t_j)) - \Phi(t_j, y^j)\| + h_j\tau_j \\ &\leq \varepsilon_j + h_jM\|y(t_j) - y^j\| + h_j\tau_j \\ &= (1 + h_j)M\varepsilon_j + h_j\tau_j. \end{aligned}$$

Mit Blick auf die diskrete GRONWALL–Ungleichung sieht man mit $z_j = \varepsilon_j$, $\delta_j = h_jM$ und $\eta_j = h_j\tau_j$, dass die letzte der obigen Gleichungen gerade von der Form (3.3.2) ist. Anwendung von (3.3.3) impliziert daher

$$\begin{aligned} \varepsilon_{j+1} &\leq \left(\varepsilon_0 + \sum_{s=0}^{j} h_s\tau_s\right) e^{M\sum_{s=0}^{j} h_s} \\ &\leq \left(\varepsilon_0 + \|\tau(h)\| \sum_{s=0}^{j} h_s\right) e^{M\sum_{s=0}^{j} h_s} \end{aligned}$$

für jedes $j = 0, \ldots, N-1$, woraus wiederum

$$\|\varepsilon(t,h)\| = \max_{j=0,\ldots,N} \varepsilon_j \leq \left(\varepsilon_0 + \|\tau(h)\| \max_{j=0,\ldots,N} \sum_{s=0}^{j} h_s\right) e^{M(t-t_0)}$$

folgt. ∎

Es sei noch bemerkt, dass die entsprechende Aussage auch für ein implizites ESV gezeigt werden kann, der Beweis aber aufwendiger ist.

Bisher beinhaltet das Konvergenzresultat nur das Verhalten von lokalen Diskretisierungsfehlern. Die Frage nach zusätzlichen *Rundungsfehlern* wurde bisher noch nicht berücksichtigt.

Bemerkung 3.3.9 (Einfluss von Rundungsfehlern bei ESV)**.** Zur Konsistenz– und damit zur Konvergenzordnung von ESV ist folgendes zu bemerken. Das EULER'sche Polygonzugverfahren (3.2.1) ist zwar sehr einfach, hat aber nur Ordnung 1, weshalb es für die Praxis aufgrund folgender Überlegung meistens ungeeignet ist. Die Aufgabe sei es, die Anfangswertaufgabe (2.2) von $t_0 = 0$ bis $t_N = 1$ zu integrieren. Um eine Genauigkeit von etwa 10^{-8} zu garantieren, benötigt man also Schrittweiten $h \approx 10^{-8}$, d.h. $N = 10^8$ Schritte. Macht man in jedem Schritt *Rundungsfehler* der Größenordnung 10^{-8} (single precision) und addiert diese zum Diskretisierungsfehler, der etwa

$h^2 \approx 10^{-16}$ beträgt, so ergibt sich bei 10^8 Schritten insgesamt ein Fehler von $10^8(10^{-16}+10^{-8}) \approx 1$, also schlimmstenfalls keine Konvergenz!

Daher ist also die Schrittweite so zu bemessen, dass die Rundungsfehler (viel) kleiner als die lokalen Abbruchfehler sind. Bei Rundungsfehlern von etwa 10^{-8} sind beim EULER–Verfahren daher nur Schrittweiten $h \geq 10^{-4}$ sinnvoll. Dann ist aber die Endgenauigkeit $10^4((10^{-4})^2 + 10^{-8}) \approx 10^{-4}$ die bestmögliche Genauigkeit, die sich mit einem Verfahren erster Ordnung erreichen lässt. Aufgrund der gegenläufigen Auswirkung der Rundungsfehler lässt sich daher auch durch *keine* noch so lange Rechnung eine beliebig hohe Genauigkeit erreichen. Abhilfe schaffen hier lediglich Verfahren *höherer* Ordnung. So lässt sich bei einem Verfahren dritter Ordnung feststellen: Bei einer gewünschten Genauigkeit von 10^{-8} beträgt der Rundungsfehler 10^{-8}. Der Diskretisierungsfehler liegt in der Größenordnung von h^4, was $h \approx 10^{-2}$ bedeutet. Also benötigt man $N = 10^2$ Schritte. Hiermit erhält man dann mit $10^2((10^{-2})^4+10^{-8}) \approx 10^{-6}$ eine deutlich bessere Endgenauigkeit.

Daher werden in den folgenden Kapiteln einige Prinzipien zur Konstruktion von *Verfahren höherer Ordnung* vorgestellt. ■

3.4 Die Methode der Taylor–Entwicklung

Gegeben sei wieder die aus (2.2) bekannte Anfangswertaufgabe (CAUCHY–Problem)

$$y' = f(t, y), \quad t \geq t_0, \quad y(t_0) = y^0.$$

Unter der Annahme, dass f hinreichend glatt in einer Umgebung von $\{(t,\ y(t; t_0, y^0)) :\ t \in I = [t_0 - a, t_0 + a]\}$ ist, sind die Ableitungen von y dann durch die Ableitungen von f bestimmt. Für $y(t) = y(t; t_0, y^0)$ gilt $y'(t) = f(t, y)$ und damit

$$y''(t) = \frac{\partial f}{\partial t}(t, y) + \frac{\partial f}{\partial y}(t, y)y'(t) = \frac{\partial f}{\partial t}(t, y) + \frac{\partial f}{\partial y}(t, y)f(t, y) =: (Df)(t, y). \tag{3.4.1}$$

Allgemein gilt

$$y^{(p+1)}(t) = (D^p f)(t, y) = D((D^{p-1} f)(t, y)),$$

womit man

$$\begin{aligned} \frac{y(t+h) - y(t)}{h} &= y'(t) + \frac{h}{2}y''(t) + \ldots + \frac{h^{p-1}}{p!}y^{(p)}(t) + O(h^p) \\ &= f(t, y) + \frac{h}{2}(Df)(t, y) + \ldots + \frac{h^{p-1}}{p!}(D^{p-1} f)(t, y) + O(h^p) \end{aligned}$$

erhält. Definiert man also

$$\Phi(t_j, y^j) := f(t_j, y^j) + \frac{h}{2}(Df)(t_j, y^j) + \ldots + \frac{h^{p-1}}{p!}(D^{p-1} f)(t_j, y^j),$$

erhält man wegen der Darstellung des Diskretisierungsfehlers in (3.2.4) ein Verfahren der Ordnung p. Der Nachteil hierbei ist, dass man für jedes f die Ableitungen $D^\ell f$ (symbolisch) berechnen muss. Abhilfe schaffen hier Techniken der automatischen Differentiation (vgl. etwa [DB]).

Eine weitere Möglichkeit zur Entwicklung von Verfahren höherer Ordnung, die ohne Ableitungen auskommen, wird im folgenden dargestellt.

3.5 Explizite Runge–Kutta–Verfahren

Diese um 1900 entwickelten Verfahren wählen als Ausgangspunkt die Integraldarstellung (vergl. (3.2.2))

$$y(t_{j+1}) = y(t_j) + \int_{t_j}^{t_{j+1}} f(s, y(s))ds.$$

Die Idee hierbei ist es, das Integral nun durch eine genügend genaue Quadraturformel

$$h_j \sum_{r=1}^{m} \gamma_r f(s_r, y(s_r)) \approx \int_{t_j}^{t_{j+1}} f(s, y(s))ds$$

zu approximieren. Da hier $y(s_r)$ auch nicht bekannt ist, approximiert man weiter $f(s_r, y(s_r)) \approx k_r(t_j, y^j)$.

Ein allgemeines *m–stufiges* Runge–Kutta–*Verfahren (RK–Verfahren)* hat die folgende Form. Gegeben seien reelle Gewichte $\alpha_i,\ \gamma_i$ für $i = 1, \ldots, m$ und β_{il} für $i, \ell = 1, \ldots, m$ und $y^0 = y(t_0)$. Berechne für $j = 0, 1, 2, \ldots$

$$\begin{aligned} t_{j+1} &= t_j + h_j, \\ k_i(t_j, y^j) &= f(t_j + \alpha_i h_j, y^j + h_j \sum_{\ell=1}^{m} \beta_{i\ell} k_\ell(t_j, y^j)), \qquad i = 1, \ldots, m, \\ y^{j+1} &= y^j + h_j \sum_{\ell=1}^{m} \gamma_\ell k_\ell(t_j, y^j) \ =: \ y^j + h_j \Phi(t_j, y^j). \end{aligned} \tag{3.5.1}$$

Die Gewichte seien dabei so gewählt, dass möglichst hohe Konsistenzordnung erreicht wird (dazu später mehr). Der Übersichtlichkeit halber werden solche Runge–Kutta–Verfahren üblicherweise in einer Tabelle erfasst, dem Butcher–Tableau:

$$\begin{array}{c|cccc} \alpha_1 = 0 & \beta_{11} & \cdots & \beta_{1,m-1} & \beta_{1,m} \\ \alpha_2 & \beta_{21} & & & \vdots \\ \vdots & \vdots & & & \vdots \\ \alpha_m & \beta_{m1} & \cdots & \beta_{m,m-1} & \beta_{m,m} \\ \hline & \gamma_1 & \cdots & \gamma_{m-1} & \gamma_m \end{array} \tag{3.5.2}$$

Wir werden zunächst *explizite* Verfahren behandeln, bei denen also $\beta_{i\ell} = 0$ für $\ell \geq i$ ist. In diesem Fall lassen sich die Zwischengrößen k_i explizit als

$$k_i(t_j, y^j) = f(t_j + \alpha_i h_j, y^j + h_j \sum_{\ell=1}^{i-1} \beta_{i\ell} k_\ell(t_j, y^j)), \qquad i = 1, \ldots, m,$$

berechnen. Einige Spezialfälle sind für $m = 1$ das Euler–Verfahren

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array}$$

mit der Ordnung 1.

Für $m = 2$ machen wir mit den Abkürzungen

$$\begin{array}{c|cc} 0 & & \\ \alpha & \beta & \\ \hline & \gamma_1 & \gamma_2 \end{array}$$

den Ansatz

$$\begin{aligned} \Phi(t,y) &= \gamma_1 f(t,y) + \gamma_2 f(t+\alpha h, y+\beta h f(t,y)) \\ &= \gamma_1 k_1(t,y) + \gamma_2 k_2(t,y). \end{aligned} \tag{3.5.3}$$

Die Parameter $\gamma_1, \gamma_2, \alpha, \beta$ sind nun so zu bestimmen, dass die Ordnung möglichst hoch ist.

Bemerkung 3.5.4. Für $m = 2$ ergibt sich ein Verfahren der Ordnung $p = 2$, falls

(i) $\alpha = \beta = \frac{1}{2}$, $\gamma_1 = 0$ und $\gamma_2 = 1$, also

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

ist, ergibt dies mit

$$y^{j+1} = y^j + h_j f(t_j + \frac{h_j}{2}, y^j + \frac{h_j}{2} f(t_j, y^j))$$

das *verbesserte* EULER–*Verfahren.*

(ii) (Verfahren von HEUN)

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

Dieses Verfahren lautet also

$$\begin{aligned} y^{j+1} &= y^j + h_j \left(\frac{1}{2} f(t_j, y^j) + \frac{1}{2} f(t_{j+1}, y^j + h_j f(t_j, y^j)) \right)) \\ &= y^j + \frac{h_j}{2} \left(f(t_j, y^j) + f(t_{j+1}, y^j + h_j f(t_j, y^j)) \right). \end{aligned}$$

Beweis: TAYLOR–Entwicklung ergibt (vergl. (3.4.1)

$$\begin{aligned} \frac{y(t+h) - y(t)}{h} &= y'(t) + \frac{h}{2} y''(t) + O(h^2) \\ &= f(t,y) + \frac{h}{2} \left(\frac{\partial f}{\partial t}(t,y) + \left(\frac{\partial f}{\partial y}(t,y) \right) f(t,y) \right) + O(h^2). \end{aligned} \tag{3.5.5}$$

Andererseits gilt für $m = 2$ ebenfalls mit TAYLOR–Entwicklung des zweiten Terms um (t, y)

$$\begin{aligned} \Phi(t,y) &= \gamma_1 f(t,y) + \gamma_2 f(t+\alpha h, y+\beta h f(t,y)) \\ &= \gamma_1 f(t,y) + \gamma_2 \left[f(t,y) + \alpha h \frac{\partial f}{\partial t}(t,y) + \beta h \frac{\partial f}{\partial y}(t,y) f(t,y) + O(h^2) \right] \\ &= (\gamma_1 + \gamma_2) f(t,y) + (\gamma_2 \alpha) h \frac{\partial f}{\partial t}(t,y) + (\gamma_2 \beta) h \frac{\partial f}{\partial y}(t,y) f(t,y) + O(h^2). \end{aligned}$$

Mit der Darstellung des Diskretisierungsfehlers (3.2.5) $\tau(h) = \frac{y(t+h)-y(t)}{h} - \Phi(t,y)$ folgt durch Koeffizientenvergleich $\tau(h) = O(h^2)$, wenn

$$\gamma_1 + \gamma_2 = 1, \quad \gamma_2\alpha = \frac{1}{2}, \quad \gamma_2\beta = \frac{1}{2}$$

gilt, woraus die Behauptung folgt. ∎

Bemerkung 3.5.5a. Die Konsistenzordnung $p = 3$ ist über (3.5.3) nicht erreichbar.

Im Allgemeinen, d.h. für $m > 2$, geht es also darum, die Koeffizienten im BUTCHER–Tableau (3.5.2) so zu bestimmen, dass die Konsistenzordnung p des zugehörigen Verfahrens möglichst hoch wird. Für die TAYLOR–Entwicklung gilt, siehe (3.5.5)

$$\begin{aligned}
\frac{y(t+h)-y(t)}{h} &= y'(t) + \frac{h}{2}y''(t) + O(h^2) \\
&= f(t,y) + \frac{h}{2}\left(\frac{\partial f}{\partial t}(t,y) + \left(\frac{\partial f}{\partial y}(t,y)\right) f(t,y)\right) + O(h^2).
\end{aligned}$$

Andererseits gilt bei m–stufigen expliziten RUNGE–KUTTA–Verfahren nach Definition (3.5.1)

$$\begin{aligned}
k_1(t,y) &= f(t,y) \\
k_2(t,y) &= f(t+\alpha_2 h, y + h\beta_{21}k_1(t,y)) \\
k_3(t,y) &= f(t+\alpha_3 h, y + h(\beta_{31}k_1(t,y) + \beta_{32}k_2(t,y))) \\
&\vdots \\
k_m(t,y) &= f(t+\alpha_m h, y + h(\beta_{m,1}k_1(t,y) + \ldots + \beta_{m,m-1}k_{m-1}(t,y)))
\end{aligned}$$

und für die Verfahrensfunktion wieder mit TAYLOR–Entwicklung um (t,y), d.h. $k_i(t,y) = f(t,y) + O(h)$

$$\begin{aligned}
\Phi(t,y) &= \gamma_1 k_1(t,y) + \ldots + \gamma_m k(t,y) \\
&= \gamma_1 f(t,y) + \gamma_2 f(t+\alpha_2 h, y + h\beta_{21}f(t,y)) + \ldots \\
&\quad + \gamma_m f(t+\alpha_m h, y + h(\beta_{m,1}f(t,y) + \ldots + \beta_{m,m-1}k_{m-1}(t,y))) \\
&= \sum_{\ell=1}^{m}\gamma_\ell f(t,y) + h\sum_{\ell=1}^{m}\gamma_\ell\left[\alpha_\ell\frac{\partial f}{\partial t}(t,y) + \sum_{i=1}^{\ell-1}\beta_{\ell i}\frac{\partial f}{\partial y}(t,y)f(t,y)\right] + O(h^2).
\end{aligned}$$

Gilt also z.B.

$$\sum_{\ell=1}^{m}\gamma_\ell = 1, \qquad \alpha_\ell = \sum_{i=1}^{\ell-1}\beta_{\ell i}, \quad \ell = 1,\ldots,m, \qquad \sum_{\ell=1}^{m}\gamma_\ell\alpha_\ell = \frac{1}{2}, \tag{3.5.6}$$

so hat das Verfahren mindestens eine Konsistenzordnung $p = 2$.

Bemerkung 3.5.7. Die Bedingung $\sum_{\ell=1}^{m}\gamma_\ell = 1$ ist auch notwendig für die Konsistenz aller $f \in \mathcal{C}^1$, denn es gilt

$$\begin{aligned}
\tau(h) &= \frac{y(t+h)-y(t)}{h} - \Phi(t,y) = f(t,y) - \sum_{\ell=1}^{m}\gamma_\ell f(t,y) + O(h) \\
&= \left(1 - \sum_{\ell=1}^{m}\gamma_\ell\right) f(t,y) + O(h) = O(h).
\end{aligned}$$

Ist das Verfahren konsisten, d.h. $\tau(h) = O(h)$, muss also $\sum_{\ell=1}^{m}\gamma_\ell$ gelten. ∎

Entsprechende Bedingungen zur Gewährleistung von Verfahren höherer Ordnung werden schnell unübersichtlich (kombinatorisches Problem). Die bei fortgesetzter Differentiation auftretenden Terme lassen sich am besten mittels Graphendarstellungen (Wurzelbäume) organisieren (Details siehe [HNW]). Allerdings zeigt sich, dass die maximal erreichbare Konsistenzordnung $p = p(m)$ eines m–stufigen expliziten Runge–Kutta–Verfahrens hinter der Anzahl der Stufen zurückbleibt (vgl. [Br]):

m	1	2	3	4	5	6	7	8	9	$m \geq 9$
$p(m)$	1	2	3	4	4	5	6	6	7	$\leq m+2$

1975 bzw. 1978 wurden von Curtis bzw. Hairer Verfahren der Konsistenzordnung $p = 10$ explizit konstruiert, allerdings werden $m = 18$ bzw. $m = 17$ Stufen benötigt.

Beispiel 3.5.8. Sei $m = p = 4$. Wie im Fall $m = 2$ gibt es auch hier mehrere Lösungen.

- klassisches Runge–Kutta–Verfahren (vergl. Simpson–Quadraturregel)

$$\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & \frac{1}{2} & & & \\
\frac{1}{2} & 0 & \frac{1}{2} & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
\end{array}$$

- Gills–Formel

$$\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & \frac{1}{2} & & & \\
\frac{1}{2} & \frac{\sqrt{2}-1}{2} & \frac{1-\sqrt{2}}{2} & & \\
1 & 0 & -\frac{\sqrt{2}}{2} & \frac{1+\sqrt{2}}{2} & \\
\hline
 & \frac{1}{6} & \frac{1}{6}(2-\sqrt{2}) & \frac{1}{6}(2+\sqrt{2}) & \frac{1}{6}
\end{array}$$

- $\frac{3}{8}$–Regel

$$\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{3} & \frac{1}{3} & & & \\
\frac{2}{3} & -\frac{1}{3} & 1 & & \\
1 & 1 & -1 & 1 & \\
\hline
 & \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8}
\end{array}$$

Es gibt auch Formeln, die sich zu solchen höherer Ordnung ergänzen lassen (nützlich zur Schrittweitensteuerung). Weitere Tabellen für RK–Verfahren höherer Ordnung finden sich in [A].

■

Bemerkung 3.5.9. Sei f Lipschitz–stetig mit Konstante L. Die Inkrementfunktion der expliziten Runge–Kutta–Methoden erfüllt dann die Lipschitz–Bedingung (3.3.6). Somit sind die expliziten Runge–Kutta–Methoden *konvergent* mit Konsistenzordnung.

Beweis: Für explizite RUNGE–KUTTA–Verfahren gilt

$$k_i(t,y) = f(t+\alpha_i h, y + h\sum_{\ell=1}^{i-1}\beta_{i\ell}k_\ell(t,y)),$$

was mit der LIPSCHITZ–Bedingung für f

$$\begin{aligned}
\|k_i(t,v)-k_i(t,w)\| &= \|f(t+\alpha_i h, v+h\sum_{\ell<i}\beta_{i\ell}k_\ell(t,v)) - f(t+\alpha_i h, w+h\sum_{\ell<i}\beta_{i\ell}k_\ell(t,w))\| \\
&\le L\|v+h\sum_{\ell}\beta_{i\ell}k_\ell(t,v) - (w+h\sum_{\ell}\beta_{i\ell}k_\ell(t,w))\| \\
&\le L\left(\|v-w\| + h\sum_{\ell=1}^{i-1}|\beta_{i\ell}|\|k_\ell(t,v)-k_\ell(t,w)\|\right)
\end{aligned}$$

impliziert. Ein Induktionsargument zusammen mit $k_1(t,y) = f(t,y)$ liefert nun die LIPSCHITZ–Stetigkeit von

$$\Phi(t,y) = \sum \gamma_\ell k_\ell(t,y).$$

■

Eine Klasse von Verfahren, die sich durch Kombination von Kapitel 3.4 (TAYLOR–Entwicklung) und Kapitel 3.5 (Explizite RUNGE–KUTTA–Verfahren) ergeben, sind die RUNGE–KUTTA–FEHLBERG–Verfahren. Unter Verwendung von *Ableitungen* bis $y^{(k+1)}$ (d.h. k Ableitungen an f) erhält man so Verfahren der Ordnung $k+4$ bei $m=4$ in RUNGE–KUTTA–Verfahren.

3.6 Implizite Runge–Kutta–Verfahren

Später wird sich zeigen, dass für eine wichtige Problemklasse von Differentialgleichungen implizite Verfahren verwendet werden müssen. In (3.5.1) lautete die allgemeine Form eines m–stufigen RUNGE–KUTTA—Verfahrens:
berechne für $j = 0,1,2,\ldots$

$$\begin{aligned}
t_{j+1} &= t_j + h_j \\
k_i(t_j,y^j) &= f(t+\alpha_i h_j, y^j + h_j\sum_{\ell=1}^{m}\beta_{i\ell}k_\ell(t_j,y^j)), \quad i=1,\ldots,m, \\
y^{j+1} &= y^j + h_j\sum_{\ell=1}^{m}\gamma_\ell k_\ell(t_j,y^j) \\
&= y^j + h_j\Phi(t_j,y^j).
\end{aligned}$$

Das zugehörige BUTCHER–Tableau hatte die Form

$$\begin{array}{c|cccc}
0 & \beta_{11} & \cdots & \beta_{1,m-1} & \beta_{1,m} \\
\alpha_2 & \beta_{21} & & & \vdots \\
\vdots & \vdots & & & \vdots \\
\alpha_m & \beta_{m1} & \cdots & \beta_{m,m-1} & \beta_{m,m} \\
\hline
 & \gamma_1 & & \gamma_{m-1} & \gamma_m
\end{array}$$

Im Allgemeinen muss also in jedem Schritt, d.h. für jedes j, ein nichtlineares Gleichungssystem in mn Unbekannten $k_i \in \mathbb{R}^n$ für $i = 1,\ldots,m$ zur Bestimmung der k_i gelöst werden. Zur Lösbarkeit des Gleichungssystems hat man folgende Aussage.

Satz 3.6.1. Sei f LIPSCHITZ–stetig bzgl. y auf $\mathbb{R}^n$ mit LIPSCHITZ–Konstante L, und alle Schrittweiten $h = h_j$ seien klein genug, so dass

$$q := hL \max_{i=1,\dots,m} \sum_{\ell=1}^{m} |\beta_{i\ell}| < 1$$

ist. Dann existieren für alle $(t, y) \in \mathbb{R} \times \mathbb{R}^n$ eindeutige Lösungsvektoren

$$k_1(t, y), \dots, k_m(t, y)$$

von (3.5.1).

Beweis: Die Beweisidee besteht darin, ein Iterationsverfahren zu definieren und zu zeigen, dass es eine Kontraktion ist. Dann folgt die Aussage aus dem BANACH'schen Fixpunktsatz. Wähle dazu zu beliebigem, aber festen (t, y) beliebige Startvektoren $k_i^{(0)}(t, y)$ für jedes $i = 1, \dots, m$ und definiere das Iterationsverfahren für $r = 1, 2, \dots$

$$k_i^{(r)} := k_i^{(r)}(t, y) := f(t + \alpha_i h, y + h \sum_{\ell=1}^{m} \beta_{i\ell} k_\ell^{(r-1)}(t, y))$$

(Gesamtschrittverfahren, das allerdings hier nichtlinear in k_i ist). Wir zeigen nun, dass die Grenzwerte

$$\lim_{r\to\infty} k_i^{(r)} = k_i$$

für $i = 1, \dots, m$ die Gleichungen in (3.5.1) lösen. Dazu schaffen wir einen geeigneten Rahmen für die Anwendung des BANACH'schen Fixpunktsatzes wie folgt. Setze

$$K := \begin{pmatrix} k_1 \\ \vdots \\ k_m \end{pmatrix} \in \mathbb{R}^{nm}, \quad \|\!|K|\!\| := \max_{i=1,\dots,m} \|k_i\|$$

und

$$F : \mathbb{R}^{nm} \to \mathbb{R}^{nm}, \quad F(K) = \begin{pmatrix} F_1(K) \\ \vdots \\ F_m(K) \end{pmatrix}.$$

Mit der Definition der k_i lautet das Iterationsverfahren dann $K^{(r)} = F(K^{(r-1)})$, $r = 1, 2, \dots$, mit $F_i(K) := f(t + \alpha_i h, y + h \sum_{\ell=1}^{m} \beta_{i\ell} k_\ell)$. Zeige nun, dass F eine Kontraktion ist. Die LIPSCHITZ–Bedingung für f (bzgl. y) liefert nun

$$\begin{aligned}
\|\!|F(K) - F(\tilde{K})|\!\| &= \max_{i=1,\dots,m} \|F_i(K) - F_i(\tilde{K})\| \\
&\le L \max_i \|y + h \sum_{\ell=1}^{m} \beta_{i\ell} k_\ell - y - h \sum_{\ell=1}^{m} \beta_{i\ell} \tilde{k}_\ell\| \\
&\le Lh \max_i \sum_{\ell=1}^{m} |\beta_{i\ell}| \|k_\ell - \tilde{k}_\ell\| \\
&\le Lh \max_i \sum_{\ell=1}^{m} |\beta_{i\ell}| \max_j \|k_j - \tilde{k}_j\| \\
&= Lh \max_i \sum_{\ell=1}^{m} |\beta_{i\ell}| \|\!|K - \tilde{K}|\!\|.
\end{aligned}$$

Nach Voraussetzung ist $Lh \max\limits_i \sum\limits_{\ell=1}^m |\beta_{i\ell}| < 1$ und F somit eine Kontraktion. Damit folgen Existenz und Eindeutigkeit aus dem BANACH'schen Fixpunktsatz. ∎

Bemerkung 3.6.2. Die Forderung nach LIPSCHITZ–Stetigkeit lässt sich wiederum auf eine lokale LIPSCHITZ–Bedingung abschwächen. Hierzu und um $q < 1$ zu gewährleisten, muss h bisweilen *sehr klein* gewählt werden.

Mit der möglicherweise hohen Anzahl von Iterationen geht dabei der Vorteil einer hohen Konsistenzordnung aufgrund des hohen Rechenaufwands verloren. Daher setzt man meist besser das NEWTON–Verfahren als das obige Gesamtschrittverfahren zur Lösung von (3.5.1) ein, sobald man im Konvergenzbereich des NEWTON–Verfahrens ist. ∎

Zur Konsistenzordnung halten wir Folgendes fest.

Bemerkung 3.6.3. (Explizite und implizite) RUNGE–KUTTA–Methoden sind genau dann konsistent, wenn $\sum_{\ell=1}^m \gamma_\ell = 1$ gilt. ∎

Satz 3.6.4. Sei f LIPSCHITZ–stetig auf $\mathbb{R}^n$ bzgl. y mit L. Wählt man zu paarweise verschiedenen $\alpha_j \in [0,1]$, $j = 1,\dots,m$, Parameter γ_ℓ, $\beta_{k\ell}$, $k,\ell = 1,\dots,m$ so, dass für ein $r \le m$

$$\sum_{k=1}^m \beta_{jk}\alpha_k^\ell = \frac{\alpha_j^{\ell+1}}{\ell+1}, \quad \ell = 0,\dots,r-1, \quad j = 1,\dots,m, \tag{3.6.5a}$$

und

$$\sum_{k=1}^m \gamma_k\alpha_k^\ell = \frac{1}{\ell+1}, \quad \ell = 0,\dots,r-1, \tag{3.6.5b}$$

gilt, dann sind die entsprechenden RUNGE–KUTTA–Formeln konsistent von der Ordnung $p = r$. ∎

Beachte, dass (3.6.5a) und (3.6.5b) sich als Systeme

$$\underbrace{\begin{pmatrix} \beta_{11} & \cdots & \beta_{1m} \\ \vdots & & \vdots \\ \beta_{m1} & \cdots & \beta_{mm} \end{pmatrix}}_{=:B} \underbrace{\begin{pmatrix} \alpha_1^0 & \cdots & \alpha_1^{r-1} \\ \vdots & & \vdots \\ \alpha_m^0 & \cdots & \alpha_m^{r-1} \end{pmatrix}}_{=:A} = \underbrace{\begin{pmatrix} \alpha_1 & \cdots & \frac{\alpha_1^r}{r} \\ \vdots & & \vdots \\ \alpha_m & \cdots & \frac{\alpha_m^r}{r} \end{pmatrix}}_{=:C} \tag{5a}$$

und

$$\begin{pmatrix} \alpha_1^0 & \cdots & \alpha_m^0 \\ \vdots & & \vdots \\ \alpha_1^{r-1} & \cdots & \alpha_m^{r-1} \end{pmatrix} \begin{pmatrix} \gamma_1 \\ \vdots \\ \gamma_m \end{pmatrix} = \begin{pmatrix} 1 \\ \vdots \\ 1/r \end{pmatrix} \tag{5b}$$

schreiben lassen. (5a) ist äquivalent zu $A^T B^T = C^T$. Weiter ist A eine VANDERMONDE–Matrix. Da die α_j paarweise verschieden sind, ist $\operatorname{rang} A = r$, d.h. A^T ist surjektiv und (5a) somit stets lösbar. Selbiges gilt für (5b). Insbesondere sind für $r = m$ beide Gleichungssysteme eindeutig lösbar.

Beweis von 3.6.4: (3.6.5a) und (3.6.5b) bedeuten gerade, dass die Quadraturformeln (NEWTON–COTES–Formeln) mit den Gewichten β_{jk} auf $[0,\alpha_j]$ und γ_k auf $[0,1]$ exakt vom Grad $r-1$ sind. Zur Erinnerung: die Quadraturformeln sind exakt vom Grad $r-1$, wenn

$$\int_0^1 p(t)dt = \sum_{k=1}^m \gamma_k p(\alpha_k)$$

für alle $p \in \mathcal{P}_{r-1}$ (Polynome vom Grad $r-1$) sind. Wähle $p(t) = t^\ell$. Dann gilt

$$\frac{1}{\ell+1} = \int_0^1 t^\ell dt = \sum_{k=1}^m \gamma_k (\alpha_k)^\ell$$

für alle $\ell = 0, \ldots, r-1$, was (3.6.5b) entspricht. Des weiteren betrachte zu (5a)

$$\int_0^{\alpha_j} p(t)dt = \sum_{k=1}^m \beta_{jk} p(\alpha_k)$$

für alle $p \in \mathcal{P}_{r-1}$. Speziell für $p(t) = t^\ell$ gilt dann

$$\frac{(\alpha_j)^{\ell+1}}{\ell+1} = \int_0^{\alpha_j} t^\ell dt = \sum_{k=1}^m \beta_{jk} (\alpha_k)^\ell$$

für jedes $j = 1, \ldots, m$. Wir zeigen nun, dass die entsprechenden Runge–Kutta–Formeln konsistent von der Ordnung r sind. Wegen der Exaktheit der Quadratur vom Grad $r-1$ gilt

$$\begin{aligned}
y(t+h) - y(t) &= \int_t^{t+h} f(\tilde{s}, y(\tilde{s}))d\tilde{s} \\
&= h \int_0^1 f(t+hs, y(t+hs))ds \\
&= h \left(\sum_{\ell=1}^m \gamma_\ell f(t+\alpha_\ell h, y(t+\alpha_\ell h)) + O(h^r) \right) \\
&= h \sum_{\ell=1}^m \gamma_\ell f(t+\alpha_\ell h, y(t+\alpha_\ell h)) + O(h^{r+1}).
\end{aligned}$$

Damit gilt mit (3.5.1) und der Lipschitz–Stetigkeit von f

$$\begin{aligned}
\|\tau(h)\| &= \left\| \frac{y(t+h)-y(t)}{h} - \Phi(t,h) \right\| \\
&= \left\| \frac{y(t+h)-y(t)}{h} - \sum_{\ell=1}^m \gamma_\ell k_\ell(t,y) \right\| \\
&= \left\| \sum_{\ell=1}^m \gamma_\ell f(t+\alpha_\ell h, y(t+\alpha_\ell h)) - \sum_{\ell=1}^m \gamma_\ell f(t+\alpha_\ell h, y + h \sum_{i=1}^m \beta_{\ell i} k_i(t,y)) \right\| + O(h^r) \\
&= \left\| \sum_{\ell=1}^m \gamma_\ell \left[f(t+\alpha_\ell h, y(t+\alpha_\ell h)) - f(t+\alpha_\ell h, y + h \sum_{i=1}^m \beta_{\ell i} k_i(t,y)) \right] \right\| + O(h^r) \\
&\leq L \sum_{\ell=1}^m |\gamma_\ell| \left\| y(t+\alpha_\ell h) - y(t) - h \sum_{i=1}^m \beta_{\ell i} k_i(t,y) \right\| + O(h^r).
\end{aligned}$$

Auf die gleiche Weise folgt unter Ausnutzung der Quadraturformeln auf $[0, \alpha_j]$, also (3.6.5a), und

der Definition der k_i

$$\begin{aligned}
A_j &:= \left\| y(t+\alpha_j h) - y(t) - h\sum_{i=1}^m \beta_{ji} k_i(t,y) \right\| \\
&= \left\| h\int_0^{\alpha_j} f(t+sh, y(t+sh))ds - h\sum_{i=1}^m \beta_{ji} k_i(t,y) \right\| \\
&= \left\| h\sum_{i=1}^m \beta_{ji} f(t+\alpha_i h, y(t+\alpha_i h)) - h\sum_{i=1}^m \beta_{ji} k_i(t,y) \right\| + O(h^{r+1}) \\
&= \left\| h\cdot(\sum_{i=1}^m \beta_{ji}(f(t+\alpha_i h, y(t+\alpha_i h)) - \sum_{i=1}^m \beta_{ji} f(t+\alpha_i h, y(t) + \sum_{\ell=1}^m \beta_{i\ell} k_\ell(t,y)))) \right\| + O(h^{r+1})
\end{aligned}$$

und weiter

$$\begin{aligned}
A_j &= \left\| h\sum_{i=1}^m \beta_{ji} \left[f(t+\alpha_i h, y(t+\alpha_i h)) - f(t+\alpha_i h, y(t) + h\sum_{\ell=1}^m \beta_{i\ell} k_\ell(t,y)) \right] \right\| + O(h^{r+1}) \\
&\le Lh\sum_{i=1}^m |\beta_{ji}| \| y(t+\alpha_i h) - y(t) - h\sum_{\ell=1}^m \beta_{i\ell} k_\ell(t,y) \| + O(h^{r+1}) \\
&= hL\sum_{i=1}^m |\beta_{ji}| A_i + O(h^{r+1}) \\
&\le hL\left(\max_{\ell=1,\dots,m} B_\ell\right)\left(\max_{i=1,\dots,m} A_i\right) + O(h^{r+1}).
\end{aligned}$$

Setze $B := \max\limits_{l=1,\dots,m} B_l$ und $A := \max\limits_{i=1,\dots,m} A_i$. Damit folgt also

$$A \le hLBA + O(h^{r+1}) \quad \text{oder} \quad A(1-hLB) \le O(h^{r+1}),$$

also für h genügend klein, d.h. $hLB < 1$ ist $A(1-hLB) > 0$ und somit $A = O(h^{r+1})$. Schließlich folgt für $\tau(h)$

$$\begin{aligned}
\|\tau(h)\| &\le L\sum_{\ell=1}^m |\gamma_\ell||A_\ell| + O(h^r) \\
&\le LA\sum_{\ell=1}^m |\gamma_\ell| + O(h^r) \\
&= O(h^r).
\end{aligned}$$

■

Beachte, dass die Aussage von 3.6.4 für *beliebige* paarweise verschiedene Stützstellen α_j gilt. Im Hinblick auf die Gauss–Quadratur erwartet man höhere Genauigkeit bei geeigneter Wahl von α_j. Für ein genügend glattes f lässt sich dann $p = 2m$ erreichen.

Beispiel 3.6.6 (Implizite Runge–Kutta–Verfahren)**.**

(a) Gauss–Form: α_j sind Nullstellen von Orthogonalpolynomen. Für $m = 1$ ist $p = 2$ und

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array} \qquad k_1 = f(t+\tfrac{h}{2}, y+\tfrac{h}{2}k_1) \qquad y^{j+1} = y^j + hk_1(t_j, y^j)$$

(b) RADAU–Form: entweder $\alpha_1 = \beta_{11} = \ldots = \beta_{1m} = 0$ (erste Zeile) oder $\beta_{1m} = \ldots = \beta_{mm} = 0$ (letzte Spalte). k_1 oder k_m sind dann explizit berechenbar. Für $m = 1$ mit $p = 1$ erhält man das EULER–Verfahren

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}$$

Für $m = 2$ folgt $p = 3$ und

$$\begin{array}{c|cc} 0 & 0 & 0 \\ \frac{2}{3} & \frac{1}{3} & \frac{1}{3} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array} \qquad \begin{array}{c|cc} \frac{1}{3} & \frac{1}{3} & 0 \\ 1 & 1 & 0 \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$$

Allgemein lässt sich hier $p = 2m - 1$ erreichen.

(c) LOBATTO–Form: $\alpha_1 = \beta_{11} = \ldots = \beta_{1m} = \beta_{2m} = \ldots = \beta_{mm} = 0$ (erste Zeile und letzte Spalte 0). k_1 und k_m sind explizit berechenbar, und es gilt $p = 2m - 2$. Im Fall $m = 3$ mit $p = 4$ erhält man das Schema

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 0 \\ 1 & 0 & 1 & 0 \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

■

Kurze Einordnung und Zusammenfassung von Einschrittverfahren

Generell ist eine hohe Konsistenzordnung entscheidend zur Vermeidung von Rundungsfehlereinflüssen. Ist f glatt genug, so folgt aus der LIPSCHITZ–Stetigkeit, dass das Verfahren mit der entsprechenden Konsistenzordnung konvergiert. Dennoch haben Einschrittverfahren einen gemeinsamen *Nachteil*:

Die Realisierung höherer Ordnung ist recht mühsam und wird im Allgemeinen über eine hohe Anzahl von Funktionsauswertungen erkauft. D.h. der Vorteil höherer Genauigkeit wird durch erhöhten Aufwand zunichte gemacht.

Vorteile von Einschrittverfahren sind

- *Schrittweitensteuerung (adaptive Verfahren)*: die Schrittweite h_j lässt sich im Laufe der Rechnung variieren. Dadurch lässt sich der Gesamtaufwand, der für die gewünschte Genauigkeit nötig ist, *erheblich* reduzieren;
- *Extrapolation*: bei Einschrittverfahren lässt sich die Existenz einer asymptotischen Entwicklung für den lokalen Abbruchfehler zeigen. Damit kann man über *Extrapolationsverfahren* die Genauigkeit erhöhen;
- später wird sich zeigen, dass sich über Mehrschrittverfahren eine höhere Genauigkeit mit weniger Funktionsauswertungen erzielen lässt. Diese benötigen aber Einschrittverfahren mit hoher Genauigkeit als *Anlaufrechnung*.

3.7 Asymptotische Entwicklung, Extrapolation und Schrittweitensteuerung

Es wurde gezeigt, dass unter geeigneten Voraussetzungen an f und Φ für den *lokalen* Abbruchfehler

$$\tau(h) = O(h^p)$$

gilt. Dies legt die Existenz einer *asymptotischen Entwicklung* von $y_h(t)$ für ein Verfahren der Ordnung p nahe:

$$y_h(t) = y(t) + e_p(t)h^p + \ldots + e_{m+p}(t)h^{m+p} + h^{m+p+1}E(t,h), \tag{3.7.1}$$

wobei $E(t,h)$ für jedes feste t in h beschränkt ist.

Satz (Gragg (1964)). Für alle „vernünftigen“ Einschrittverfahren und glatten f liegt eine asymptotische Entwicklung (3.7.1) vor.

Anwendungen

(i) Die Schranke im Konvergenzsatz 3.3.5

$$\|\varepsilon(t,h)\| \leq (t-t_0)\|\tau(h)\|e^{M(t-t_0)}$$

ist oft zu pessimistisch. (3.7.1) erlaubt, den tatsächlichen Diskretisierungsfehler abzuschätzen. Betrachte dazu (3.7.1) für h und $h/2$,

$$\begin{aligned} y_h(t) &= y(t) + e_p(t)h^p + O(h^{p+1}) \\ y_{\frac{h}{2}}(t) &= y(t) + e_p(t)\left(\frac{h}{2}\right)^p + O(h^{p+1}), \end{aligned}$$

was

$$y_h(t) - y_{\frac{h}{2}}(t) = e_p(t)h^p(1-2^{-p}) + O(h^{p+1})$$

oder äquivalent dazu

$$e_p(t)h^p = \frac{2^p}{2^p-1}(y_h(t) - y_{\frac{h}{2}}(t)) + O(h^{p+1}) \tag{3.7.2}$$

impliziert.

(ii) Extrapolation: Einsetzen von (3.7.2) in (3.7.1) liefert

$$y_h(t) - \frac{2^p}{2^p-1}(y_h(t) - y_{\frac{h}{2}}(t)) = y(t) + O(h^{p+1}). \tag{3.7.3}$$

Setzt man die linke Seite als $\hat{y}_h(t)$, so liefert dies ein Verfahren der Ordnung $p+1$ (entspricht der Auswertung des linearen Interpolationspolynoms bzgl. $(h^p, y_h(t))$ und $((h/2)^p, y_{h/2}(t))$ an der Stelle $h=0$). Dies verwendet man nun zur

(iii) Schrittweitensteuerung: Ist h_j zu groß, so ist der globale Diskretisierungsfehler $\|\varepsilon(t,h)\|_\infty$ zu groß. Ist umgekehrt h_j zu klein, so ergeben sich zu viele Rundungsfehler. Man sollte das Problem bis zur gewünschten Genauigkeit mit möglichst wenig Aufwand lösen. Heuristisch lautet das Vorgehen wie folgt:

- spezifiziere die gewünschte Genauigkeit durch Vorgabe einer Toleranz $\varepsilon > 0$. Der von t_0 bis $t = t_N$ akkumulierte *globale* Diskretisierungsfehler soll dann

$$\|\varepsilon(t_N,h)\| \leq \varepsilon|t_N - t_0| = \varepsilon\sum_{j=0}^{N-1} h_j$$

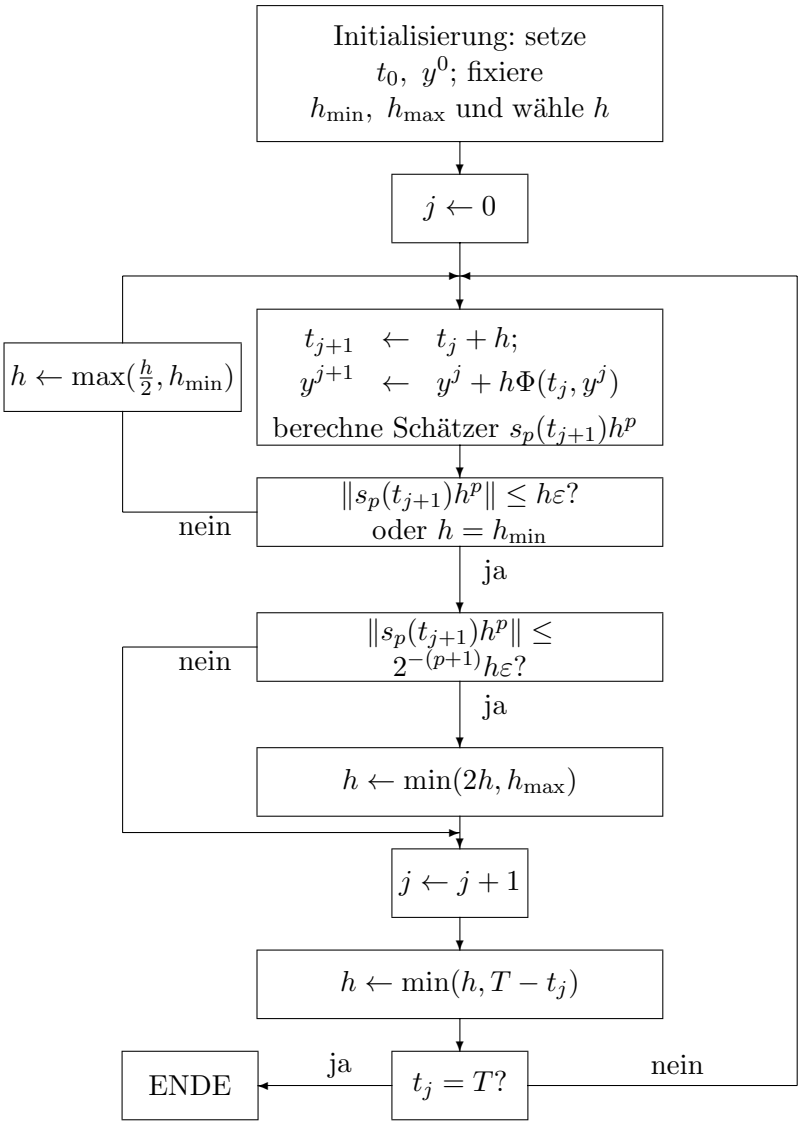
  erfüllen.

- Mit dem Konvergenzbeweis sieht man, dass sich im Wesentlichen die lokalen Abbruchfehler summieren. Man fordert also, dass in jedem Schritt der Abbruchfehler $\tau(h_j)$ höchstens $h_j\varepsilon$ mit aktueller Schrittweite h_j ist.
- man benutze obiges Extrapolationsschema zur *Schätzung* von $\tau(h_j)$ wie folgt: berechne den führenden Fehlerterm aus (3.7.2) durch

$$s_p(t)h^p := \frac{2^p}{2^p - 1}(y_h(t) - y_{\frac{h}{2}}(t)).$$

  $y_h(t)$ entspricht der Berechnung von y^{j+1} mit Schrittweite h, und $y_{\frac{h}{2}}(t)$ bedeutet die Berechnung von zwei Schritten mit Schrittweite $\frac{h}{2}$, um eine weitere Approximation $\tilde{y}^{j+1}$ zu erhalten.

Dies führt zu folgendem *Flussdiagramm* zur Berechnung von $y(T)$ zu einem Endzeitpunkt T.



Bemerkung: Es ist wichtig, $h_{\min}$ (Kontrolle der Rundungsfehler!) und $h_{\max}$ zu setzen. Dadurch verhindert man, dass die Schrittweite außer Kontrolle gerät. Falls $h = h_{\min}$, überlässt man dem Nutzer, die Rechnung abzubrechen oder mit $h_{\min}$ weiterzurechnen. Beachte, dass die Schrittweitensteuerung allerdings nicht verhindert, dass lokale Singularitäten übersehen werden können.

4 Mehrschrittverfahren zur Lösung von Anfangswertaufgaben

Der generelle Nachteil bei Einschrittverfahren ist der hohe Aufwand zur Realisierung von Verfahren höherer Ordnung (d.h. viele Funktionsauswertungen), da jeder Schritt so behandelt wird, als sei er der erste. Eine andere Strategie ist die folgende.

4.1 Definition von k–Schritt–Verfahren

Für die Anfangswertaufgabe

$$y' = f(t, y), \quad t \geq t_0, \quad y(t_0) = y^0 \tag{4.1.1}$$

ist ein *k–Schritt–Verfahren* durch

$$\sum_{\ell=0}^{k} a_\ell y^{j+\ell} = h\Phi(t_j, \ldots, t_{j+k}, y^j, \ldots, y^{j+k}, h), \quad j = 0, 1, 2, \ldots, \tag{4.1.2}$$

definiert, wobei $t_j = t_0 + jh$ für $j = 0, 1, 2, \ldots$ und $a_l \in \mathbb{R}$ gegebene Koeffizienten, $l = 0, \ldots, k$, sind. Der Wert y^{j+k} wird dabei aus den vorherigen Werten $y^j, \ldots, y^{k+k-1}$ berechnet. Im Unterschied zu den Einschrittverfahren wählen wir hier die Schrittweite immer gleich und schreiben $h_j = h$. Ein Mehrschrittverfahren heißt *explizit*, falls Φ unabhängig von y^{j+k} ist, sonst *implizit*. Im impliziten Fall ist in der Regel ein (nichtlineares) Gleichungssystem zu lösen. Beachte, dass man bei einem Mehrschrittverfahren stets k *Anlaufwerte* $y^j = y_h^j = \xi^j$ für $j = 0, \ldots, k-1$ benötigt, die man etwa über ein Einschrittverfahren berechnen kann.

Bei einem *linearen Mehrschrittverfahren* (lineares MSV) hat die Verfahrensfunktion die Form

$$\Phi(t_j, \ldots, t_{j+k}, y^j, \ldots, y^{j+k}, h) = \sum_{\ell=0}^{k} b_\ell f_{j+\ell} \tag{4.1.3}$$

mit $f_{j+\ell} := f(t_{j+\ell}, y^{j+\ell})$ und gegebenen Koeffizienten $b_\ell \in \mathbb{R}$, $\ell = 0, \ldots, k$, d.h. ein lineares MSV hat die Form

$$\sum_{\ell=0}^{k} a_\ell y^{j+\ell} = h \sum_{\ell=0}^{k} b_\ell f_{j+\ell}, \quad j = 0, 1, 2, \ldots, \tag{4.1.4}$$

mit Anlaufwerten $y_h^j = \xi^j$ für $j = 0, \ldots, k-1$. Falls das Verfahren implizit ist ($b_k \neq 0$), lässt sich mit dem Banach'schen Fixpunktsatz für genügend kleines h zeigen, dass stets eine eindeutige Lösung y^{j+k} von (4.1.4) existiert.

4.2 Lokaler Abbruchfehler und Konsistenz

Sei

$$\tau(t, y, h) := \frac{1}{h} \sum_{\ell=0}^{k} a_\ell y(t + \ell h) - \Phi(t, \ldots, t + kh, y(t), \ldots, y(t + kh), h) \tag{4.2.1}$$

mit $y(t)$ als exakter Lösung. Dann ist $h\tau(t, y, h)$ der *lokale Abbruch–* oder *Diskretisierungsfehler*.

Definition 4.2.2. Das Mehrschrittverfahren (4.1.3) heißt *konsistent* (von der Ordnung p) mit der Anfangswertaufgabe (4.1.1), falls für alle genügend glatten f

$$\|y_h^j - y^0\| \to 0 \quad \text{bzw.} \quad O(h^p),\ h \to 0, \quad j = 0, \ldots, k-1,$$

und

$$\max_{j=0,\dots,N-k} \|\tau(t_j, y, h)\| \to 0 \quad \text{bzw.} \quad O(h^p),\ h \to 0$$

gilt. Hier ist zusätzlich gefordert, dass die ersten k Werte mit $h \to 0$ gegen den Anfangswert y^0 streben. (Bei Einschrittverfahren ist dies wegen des alleinigen Startwerts $y(t_0) = y^0$ stets erfüllt.)

■

Beispiel 4.2.3 (Mittelpunktsregel)**.** Dieses explizite Verfahren ist gegeben durch

$$y^{j+2} - y^j = 2hf_{j+1}.$$

TAYLOR–Entwicklung um $y(t)$ liefert

$$\begin{aligned} y(t+h) &= y(t) + hy'(t) + \frac{h^2}{2}y''(t) + \frac{h^3}{6}y'''(t) + O(h^4), \\ y(t-h) &= y(t) - hy'(t) + \frac{h^2}{2}y''(t) - \frac{h^3}{6}y'''(t) + O(h^4). \end{aligned}$$

Abziehen der Gleichungen, d.h. Einsetzen der exakten Lösung in das Verfahren und (4.1.1) impliziert

$$y(t+h) - y(t-h) - 2hf(t, y(t)) = 2h(y'(t) - f(t, y(t))) + \frac{h^3}{3}y'''(t) + O(h^4) = O(h^3).$$

Also gilt

$$\tau(t-h, y, h) = \frac{y(t+h) - y(t-h)}{h} - 2f(t, y(t)) = O(h^2),$$

womit das Verfahren von der Ordnung $p = 2$ ist. ■

4.3 Allgemeine Konsistenzkriterien

Für das Mehrschrittverfahren (4.1.2) definieren wir das *erste charakteristische Polynom* durch

$$\rho(z) := \sum_{\ell=0}^{k} a_\ell z^\ell, \tag{4.3.1}$$

wobei a_ℓ die Koeffizienten der linken Seite von (4.1.2) sind.

Bemerkung 4.3.2. Sei $y \in \mathcal{C}^1([t_0, T])$. Das Verfahren (4.1.2) ist genau dann mit der Anfangswertaufgabe (4.1.1) konsistent, wenn

$$\begin{aligned} \|y_h^j - y^0\| &\to 0, \quad h \to 0, \quad j = 0, \dots, k-1, \\ \rho(1) &= 0 \\ \lim_{h\to 0} \max_{j=0,\dots,N-k} \|\Phi(t_j, \dots, t_{j+k}, y(t_j), &\dots, y(t_{j+k}), h) - \rho'(1)f(t_j, y(t_j))\| = 0 \end{aligned} \tag{4.3.3}$$

gelten.

Beweis: Aus der Definition des lokalen Abbruchfehlers (4.2.1) folgt mit TAYLOR–Entwicklung von y um t

$$\begin{aligned}
\tau(t,y,h) &= \frac{1}{h}\sum_{\ell=0}^{k} a_\ell y(t+\ell h) - \Phi(t,\ldots,t+kh,y(t),\ldots,y(t+kh),h) \\
&= \frac{1}{h}\left[\left(\sum_{\ell=0}^{k} a_\ell\right) y(t) + \left(\sum_{\ell=0}^{k} a_\ell(\ell h)\right) y'(t) + O(h^2)\right] - \Phi(\ldots) \\
&= \frac{1}{h}\rho(1)y(t) + \rho'(1)y'(t) + O(h) - \Phi(\ldots). \qquad (4.3.4)
\end{aligned}$$

Also ist

$$\max_{j=0,\ldots,N-k} \|\tau(t_j+kh,y,h)\| = \max_{j=0,\ldots,N-k} \left(\left\|\frac{1}{h}\rho(1)y(t_j) + \Phi(\ldots) - \rho'(1)f(t_j,y(t_j))\right\|\right) + O(h).$$

Daher folgt die Konsistenz aus (4.3.3). Umgekehrt zieht die Konsistenz $\|\tau(\ldots)\| \to 0$ aber auch (4.3.3) und insbesondere $\rho(1) = 0$ nach sich, da die Konsistenz für alle $y,\ f,\ \Phi$ gilt. ∎

Die dritte Bedingung in (4.3.3) lässt sich bei *linearen Mehrschrittverfahren*

$$\Phi(t_j,\ldots,t_{j+k},y^j,\ldots,y^{j+k},h) = \sum_{\ell=0}^{k} b_\ell f_{j+\ell}$$

wie folgt präzisieren. Betrachte dazu das *zweite charakteristische Polynom*

$$\sigma(z) := \sum_{\ell=0}^{k} b_\ell z^\ell. \qquad (4.3.5)$$

Bemerkung 4.3.6. Das lineare Mehrschrittverfahren (4.1.4)

$$\begin{aligned}
\sum_{\ell=0}^{k} a_\ell y^{j+\ell} &= h\sum_{\ell=0}^{k} b_\ell f_{j+\ell}, \quad j=0,1,\ldots \\
y_h^j &= \xi^j, \quad j=0,\ldots,k-1,
\end{aligned}$$

ist für alle glatten f genau dann mit der Anfangswertaufgabe (4.1.1) konsistent, wenn

$$\begin{aligned}
y_h^j &\to y^0, \quad h\to 0, \quad j=0,\ldots,k-1, \\
\rho(1) &= 0 \qquad\qquad (4.3.7) \\
\rho'(1) &= \sigma(1).
\end{aligned}$$

gilt.

Beweis: Die dritte Bedingung in (4.3.3) ist für lineare MSV gerade mit TAYLOR–Entwicklung von f um $(t,y(t))$ gegeben:

$$\begin{aligned}
\rho'(1)y'(t) - \Phi(t,\ldots,t+kh,y(t),\ldots,y(t+kh),h) &= \rho'(1)y'(t) - \sum_{\ell=0}^{k} b_\ell f(t+\ell h, y(t+\ell h)) \\
&= \rho'(1)y'(t) - \sum_{\ell=0}^{k} b_\ell f(t,y(t)) + O(h) \\
&= y'(t)(\rho'(1)-\sigma(1)) + O(h),
\end{aligned}$$

woraus die Behauptung mit Bemerkung 4.3.2 folgt. Die Konsistenz ist also äquivalent zu den algebraischen Bedingungen (4.3.7) an die charakteristischen Polynome ρ und σ. ∎

Allgemeiner gilt

Satz 4.3.8. Für lineare Mehrschrittverfahren (4.1.4) sind äquivalent:

(i) Es gilt

$$\sum_{\ell=0}^{k} \left(\ell^i a_\ell - i\ell^{i-1} b_\ell\right) = 0 \qquad \text{für } i = 0, \ldots, p. \tag{4.3.9}$$

(ii) Das Mehrschrittverfahren (4.1.4) ist für jedes $f \in \mathcal{C}^p(U)$, U Umgebung der Lösung der Anfangswertaufgabe (4.1.1), *konsistent von der Ordnung* p mit der Anfangswertaufgabe (4.1.1).

(iii) Die Konsistenzordnung für $y' = y,\ y(t_0) = 1$ beträgt p.

(iv) Die Konsistenzordnung ist p für die Anfangswertaufgaben, deren Lösungen von den Polynomen vom Grad $\leq p$ aufgespannt werden.

Beweis: (ii) $\Longrightarrow$ (iii) und (ii) $\Longrightarrow$ (iv) sind trivial, da (iii) und (iv) Spezialfälle von (ii) sind. Wir zeigen (iv) $\Longrightarrow$ (i) $\Longrightarrow$ (ii) und (iii) $\Longrightarrow$ (iv).

(iv) $\Longrightarrow$ (i): Nach Voraussetzung ist das Mehrschrittverfahren konsistent für $y(t) = t^i$, $i = 0, \ldots, p$, d.h.

$$f(t, y(t)) = y'(t) = it^{i-1}, \quad i = 0, \ldots, p.$$

Hiermit und mit der Darstellung von τ werden jetzt die Bedingungen (4.3.9) hergeleitet.

Aus der Definition von τ in (4.2.1) folgt für jedes $i = 0, \ldots, p$ mit Hilfe der binomischen Formel und Summenvertauschung

$$\begin{aligned}
\tau(t, y, h) &= \frac{1}{h} \sum_{\ell=0}^{k} a_\ell (t + \ell h)^i - \sum_{\ell=0}^{k} b_\ell i (t + \ell h)^{i-1} \\
&= \frac{1}{h} \sum_{\ell=0}^{k} a_\ell \left(\sum_{\nu=0}^{i} \binom{i}{\nu} (\ell h)^\nu t^{i-\nu} \right) - \sum_{\ell=0}^{k} b_\ell i \left(\sum_{\nu=0}^{i-1} \binom{i-1}{\nu} (\ell h)^\nu t^{i-1-\nu} \right) \\
&= \frac{1}{h} \sum_{\ell=0}^{k} a_\ell \left(t^i + \sum_{\nu=1}^{i} \binom{i}{\nu} \ell^\nu h^\nu t^{i-\nu} \right) - \sum_{\ell=0}^{k} b_\ell i \left(\sum_{\nu=1}^{i-1} \binom{i-1}{\nu} \ell^\nu h^\nu t^{i-1-\nu} \right) \\
&= \frac{1}{h} \left(\sum_{\ell=0}^{k} a_\ell \right) t^i + \sum_{\nu=1}^{i} \binom{i}{\nu} h^{\nu-1} t^{i-\nu} \left(\sum_{\ell=0}^{k} a_\ell \ell^\nu \right) \\
&\qquad - \sum_{\nu=0}^{i-1} i \binom{i-1}{\nu} h^\nu t^{i-1-\nu} \sum_{\ell=0}^{k} b_\ell \ell^\nu .
\end{aligned}$$

Zwecks Angleichung folgt nun eine Indexverschiebung und Erweiterung mit $\frac{\nu}{\nu}$ im dritten Summanden,

$$\begin{aligned}
\sum_{\nu=0}^{i-1} i \binom{i-1}{\nu} h^\nu t^{i-1-\nu} \sum_{\ell=0}^{k} b_\ell \ell^\nu &= \sum_{\nu=1}^{i} \frac{i}{\nu} \binom{i-1}{\nu-1} h^{\nu-1} t^{i-\nu} \sum_{\ell=0}^{k} \nu b_\ell \ell^{\nu-1} \\
&= \sum_{\nu=1}^{i} \binom{i}{\nu} h^{\nu-1} t^{i-\nu} \sum_{\ell=0}^{k} \nu b_\ell \ell^{\nu-1},
\end{aligned}$$

was oben eingesetzt zu

$$\begin{aligned}
\tau(t,y,h) &= \frac{1}{h}\left(\sum_{\ell=0}^{k} a_\ell\right) t^i + \sum_{\nu=1}^{i} \binom{i}{\nu} h^{\nu-1} t^{i-\nu} \left(\sum_{\ell=0}^{k} a_\ell \ell^\nu\right) \\
&\qquad - \sum_{\nu=1}^{i} \binom{i}{\nu} h^{\nu-1} t^{i-\nu} \left(\sum_{\ell=0}^{k} \nu b_\ell \ell^{\nu-1}\right) \\
&= \frac{1}{h}\left(\sum_{\ell=0}^{k} a_\ell\right) t^i + \sum_{\nu=1}^{i} \binom{i}{\nu} h^{\nu-1} t^{i-\nu} \left(\sum_{\ell=0}^{k} \left(a_\ell \ell^\nu - \nu b_\ell \ell^{\nu-1}\right)\right) \\
&=: \frac{1}{h} G(0) t^i + \sum_{\nu=1}^{i} \binom{i}{\nu} h^{\nu-1} t^{i-\nu} G(\nu) \qquad (4.3.10)
\end{aligned}$$

führt. Also impliziert $\|\tau(t,y,h)\| = O(h^p)$, dass $G(0) = \rho(1) = 0$ und $G(\nu) = 0$ für $\nu = 1,\ldots,i,\quad i \le p$, ist, womit (i) folgt.

(i) $\Longrightarrow$ (ii): Taylor–Entwicklung von f liefert eine Reduktion auf den polynomialen Fall:

$$\begin{aligned}
\tau(t,y,h) &= \frac{1}{h}\sum_{\ell=0}^{k} a_\ell y(t+\ell h) - \sum_{\ell=0}^{k} b_\ell y'(t+\ell h) \\
&= \frac{1}{h}\sum_{\ell=0}^{k} a_\ell \left(\sum_{\nu=0}^{q} \frac{(\ell h)^\nu}{\nu!} y^{(\nu)}(t)\right) + O(h^q) - \sum_{\ell=0}^{k} b_\ell \left(\sum_{\nu=0}^{q-1} \frac{(\ell h)^\nu}{\nu!} y^{(\nu+1)}(t)\right) + O(h^q) \\
&= \frac{1}{h}\left(\sum_{\ell=0}^{k} a_\ell\right) y(t) + \sum_{\nu=1}^{q} \frac{h^{\nu-1}}{\nu!} y^{(\nu)}(t) \sum_{\ell=0}^{k} a_\ell \ell^\nu \\
&\qquad - \sum_{\nu=1}^{q} \frac{h^{\nu-1}}{\nu!} y^{(\nu)}(t) \sum_{\ell=0}^{k} \nu b_\ell \ell^{\nu-1} + O(h^q) \\
&= \frac{1}{h}\left(\sum_{\ell=0}^{k} a_\ell\right) y(t) + \sum_{\nu=1}^{q} \frac{h^{\nu-1}}{\nu!} y^{(\nu)}(t) \sum_{\ell=0}^{k} \left(a_\ell \ell^\nu - \nu b_\ell \ell^{\nu-1}\right) + O(h^q).
\end{aligned}$$

Also zieht (i) für $q \ge p$

$$\begin{aligned}
\tau(t,y,h) &= \sum_{\nu=p+1}^{q} \frac{h^{\nu-1}}{\nu!} y^{(\nu)}(t) \sum_{\ell=0}^{k} \left(a_\ell \ell^\nu - \nu b_\ell \ell^{\nu-1}\right) + O(h^q) \\
&= \frac{h^p}{(p+1)!} y^{(p+1)}(t) \sum_{\ell=0}^{k} \left(a_\ell \ell^{p+1} - (p+1) b_\ell \ell^p\right) + O(h^{p+1}) \qquad (4.3.11)
\end{aligned}$$

nach sich, und damit gilt (i) $\Longrightarrow$ (ii).

(iii) $\Longrightarrow$ (iv): Betrachte das Anfangswertproblem $y' = y$ mit $y(t_0) = 1$. Die Lösung lautet $y(t) = e^{t-t_0}$. Nun liefern Taylor–Entwicklung und eine Argumentation wie in (iv) $\Longrightarrow$ (i) den Beweis. ∎

Korollar 4.3.12. Ein lineares Mehrschrittverfahren, das konsistent von der Ordnung p ist, liefert die exakte Lösung, falls y ein Polynom vom Grad $\le p$ ist. ∎

Definition. Der Koeffizient

$$c_p := \frac{1}{(p+1)!} \sum_{\ell=0}^{k} \left(a_\ell \ell^{p+1} - (p+1) b_\ell \ell^p \right) \tag{4.3.13}$$

vor $h^p y^{(p+1)}(t)$ in der Darstellung (4.3.11) von τ heißt *Fehlerkonstante*. Falls $c_p \neq 0$, ist p die *genaue* Konsistenzordnung. Der Ausdruck

$$\hat{c}_p := \frac{c_p}{\sigma(1)} \tag{4.3.14}$$

heißt *normierte Fehlerkonstante*. Anhand dieser Fehlerkonstanten können Verfahren *gleicher Ordnung* verglichen werden.

Korollar 4.3.15.

a) Zu $0 \leq r \leq k$ existieren eindeutige $\rho(z) = \sum_{\ell=0}^{k} a_\ell^* z^\ell$, $a_k^* = 1$, und $\sigma(z) = \sum_{\ell=0}^{r} b_\ell^* z^\ell \in \mathcal{P}_r$, so dass das zugehörige k–Schrittverfahren die Ordnung $p = k + r$ hat (falls $f \in \mathcal{C}^{k+r}$ in einer Umgebung der Lösung ist). Also ist maximal eine Ordnung von $2k$ erreichbar.

b) Zu $\rho(z) \in \mathcal{P}_k$, $\rho(1) = 0$ existiert für jedes $r \leq k$ ein eindeutiges $\sigma \in \mathcal{P}_r$, so dass das entsprechende Verfahren die Ordnung $p = r + 1$ hat. ∎

Bemerkung 4.3.16. Später wird sich zeigen, dass eine geeignete Wahl von ρ für die *Stabilität* des Verfahrens entscheidend ist. Korollar 4.3.15 besagt, dass man dazu stets ein geeignetes σ finden kann. ∎

Beweis von Korollar 4.3.15:
a) Wegen der Bedingungen (4.3.9) (Satz 4.3.8 (i)) erfüllen die a_l^*, b_l^* das homogene System vom VANDERMONDE–Typ

$$\underbrace{\begin{pmatrix} 1 & & \dots & & 1 & 0 & & \dots & & 0 \\ 0 & 1 & 2 & \dots & k & -1 & & \dots & & -1 \\ & \vdots & & & & & & & \vdots & \\ 0 & 1 & 2^i & \dots & k^i & 0 & -i & -i2^{i-1} & \dots & -ir^{i-1} \\ & \vdots & & & & & & & \vdots & \\ 0 & 1 & 2^p & \dots & k^p & 0 & -p & -p2^{p-1} & \dots & -pr^{p-1} \end{pmatrix}}_{=:A_{k,r,p} \in \mathbb{R}^{(p+1)\times(k+r+2)}} \begin{pmatrix} a_0^* \\ \vdots \\ a_k^* \\ b_0^* \\ \vdots \\ b_r^* \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{4.3.17}$$

Für $p = k + r + 1$ hat das System eine triviale Lösung, wenn $A_{k,r,k+r+1}$ vollen Rang hat. Die Behauptung a) folgt somit genau dann, wenn der Rang von $A_{k,r,k+r}$ maximal, also $k + r + 1$ ist. Betrachte dazu zunächst $B^T := A_{k,r,k+r+1} \in \mathbb{R}^{(k+r+2)\times(k+r+2)}$ und dasjenige Polynom $P(x) = \sum_{i=0}^{k+r+1} c_i x^i \in \mathcal{P}_{k+r+1}$, das die Bedingungen

$$P(j) = g(j), \quad j = 0, \dots, k \quad \textit{und} \quad -P'(j) = g'(j), \quad j = 0, \dots, r$$

für ein gegebenes $g \in \mathcal{C}^1$ erfüllt (d.h. $j = 0, \dots, r$ sind doppelte Nullstellen). Die zugehörige *Kollokationsmatrix* ist gerade B. Wegen dem Satz über die Eindeutigkeit des Interpolationspolynoms bei HERMITE–Interpolation hat B^T vollen Rang $k+r+2$, woraus folgt, dass $A_{k,r,k+r}$ (Zeile $k + r + 1$ weggenommen) vollen Rang $k + r + 1$ hat. D.h. $A_{k,r,k+r}$ hat einen eindimensionalen Kern. Daher legt die Normierung $a_k^* = 1$ für $p = k+r$ die eindeutige Lösung des Systems (4.3.17) fest.

b) Betrachte für $p = r+1$ die aus den Zeilen $i = 2, \dots, r+1$ und den Spalten $j = k+2, \dots, k+r+2$ bestehende Untermatrix

$$C = -\begin{pmatrix} 1 & 1 & \dots & 1 & \dots & 1 \\ 0 & 2 & \dots & 2r & \dots & 2r \\ 0 & 3 & \dots & 3 \cdot 2^2 & \dots & 3r^2 \\ & \vdots & & & \vdots & \\ 0 & (r+1) & \dots & (r+1)2^r & \dots & (r+1)r^r \end{pmatrix} \in \mathbb{R}^{(r+1)\times(r+1)}$$

von $A_{k,r,r+1}$. Durch Herausziehen des gemeinsamen Faktors i in Zeile i folgt, dass die resultierende VANDERMONDE–Matrix regulär ist,

$$\det C = (-1)(r+1)! \det \begin{pmatrix} 1 & \dots & & \dots & 1 \\ 0 & 1 & 2 & \dots & r \\ & & \vdots & & \\ 0 & 1 & 2^r & \dots & r^r \end{pmatrix} \neq 0,$$

womit b) folgt. ∎

4.4 Konstruktion von Mehrschrittverfahren mittels numerischer Differentiation

Bei linearen Mehrschrittverfahren (4.1.4) sind Verfahren der Ordnung p über die Lösung des linearen Gleichungssystems (4.3.17) für die Koeffizienten a_i und b_i von ρ bzw. σ bestimmbar. Daneben spielen besonders *die* Verfahren eine Rolle, die eine Interpretation über Differentiation oder Quadratur haben.

Gegeben seien Approximationen $y^j, \dots, y^{j+k-1}$. Man bestimme nun y^{j+k} dadurch, dass zunächst formal das Interpolationspolynom der Werte $y^j, \dots, y^{j+k}$ an den Stellen $t_j, \dots, t_{j+k}$ gebildet wird,

$$P_{j,k}(t) = P(y|t_j, \dots, t_{j+k})(t) \in \mathcal{P}_k, \quad t_j = t_0 + jh.$$

Weiter wird dann die Ableitung

$$P'_{j,k}(t_{j+k-r}) = f(t_{j+k-r}, y^{j+k-r}) \tag{4.4.1}$$

für *ein* $r \leq k$ ausgewertet. Für $r \geq 1$ heißt das Verfahren *explizit* (Auswertung an $t_j, \dots, t_{j+k-r}$), sonst *implizit* (auch Auswertung an t_{j+k}).

Die Herleitung der Verfahrenskoeffizienten kann jetzt über die NEWTON–Form des Interpolationspolynoms

$$P(y|t_j, \dots, t_{j+k})(t) = \sum_{i=0}^{k} ([t_j, \dots, t_{j+i}]y) \prod_{m=0}^{i-1} (t - t_{j+m})$$

erfolgen. Bei dem hier vorliegenden Fall äquidistanter Knoten kann man die dividierten Differenzen durch Rückwärtsdifferenzen ausdrücken, d.h. das Interpolationspolynom erhält die Form

$$P_{j,k}(t) = \sum_{i=0}^{k} (-1)^i \binom{-s}{i} \nabla_h^i y^{j+k}, \tag{4.4.2}$$

wobei $t = t_0 + (j+k+s)h = t_{j+k} + sh$ und die Rückwärtsdifferenzen rekursiv durch $\nabla_h^0 y^j := y^j$, $\nabla_h y^j := \nabla_h^1 y^j := y^j - y^{j-1}$, $\nabla_h^i := \nabla_h^{i-1}(\nabla_h)$ definiert sind. Differentiation von $P_{j,k}$ und Einsetzen in (4.4.1) ergibt

$$P'_{j,k}(t_{j+k-r}) = \sum_{i=0}^{k} (-1)^i \left(\frac{d}{dt} \binom{-s}{i} \Big|_{s=-r} \right) \nabla_h^i y^{j+k} = \frac{1}{h} \sum_{i=0}^{k} \rho_{r,k,i} \nabla_h^i y^{j+k}, \tag{4.4.3}$$

wobei

$$\rho_{r,k,i} := (-1)^i \frac{d}{ds} \binom{-s}{i} \Big|_{s=-r} \tag{4.4.4}$$

ist. Damit ergibt sich das Verfahren

$$\sum_{i=0}^{k} \rho_{r,k,i} \nabla_h^i y^{j+k} = h f_{j+k-r} \quad (= hf(t_{j+k-r}, y^{j+k-r})). \tag{4.4.5}$$

Bemerkung 4.4.6.

a) Um die Koeffizienten a_i in der Form (4.1.4) (lineares Mehrschrittverfahren) direkt zu berechnen, kann man auch die LAGRANGE–Form benutzen.

b) Die NEWTON–Form erlaubt das „Anhängen“ weiterer Terme zur Ordnungserhöhung.

c) Das Verfahren lässt sich genauso für *nicht–äquidistante* Stützstellen definieren.

d) Für $r = 1,\ k = 1, 2$ erhält man das explizite EULER–Verfahren und die Mittelpunktsregel. Für $r = 1,\ k \geq 3$ sind die Verfahren *instabil* (genaue Definition später).

e) Für $r = 0$ erhält man *implizite* Verfahren. In diesem Fall ist

$$(-1)^i \frac{d}{ds} \binom{-s}{i} \Big|_{s=0} = \begin{cases} 0, & i = 0 \\ \frac{1}{i}, & i > 0 \end{cases},$$

woraus

$$\sum_{i=1}^{k} \frac{1}{i} \nabla_h^i y^{j+k} = h f_{j+k} \tag{4.4.7}$$

folgt. Diese heißen *BDF–Verfahren (Backward Difference Formulas)*. Für $k > 6$ sind sie auch unbrauchbar, ansonsten aber für *steife Probleme* geeignet (vgl. Kap. 5).

f) *Konsistenz*: Nach Konstruktion löst das Verfahren alle Anfangswertaufgaben, die eine polynomiale Lösung vom Grad k haben, exakt. Daher beträgt die Konsistenzordnung nach Satz 4.3.8 (iv) $p = k$. ∎

4.5 Konstruktion von Mehrschrittverfahren über numerische Integration

Die exakte Lösung der Anfangswertaufgabe (4.1.1) erfüllt die Integralgleichung

$$y(t_{j+k}) = y(t_{j+r-\ell}) + \int_{t_{j+r-\ell}}^{t_{j+k}} f(s, y(s)) ds. \tag{4.5.1}$$

Für verschiedene Wahlen von r und l ersetzt man nun f durch das Interpolationspolynom bzgl. f vom Grad r,

$$P_{j,r}(t) = P(f|t_j, \ldots, t_{j+r})(t) \in \mathcal{P}_r. \tag{4.5.2}$$

Da die t_j paarweise verschieden sind, erhält man $P_{j,r}$ in der LAGRANGE–Darstellung als

$$P_{j,r}(t) = \sum_{i=0}^{r} f_{j+i} L_i(t), \quad L_i(t) = \prod_{\substack{q=0 \\ q \neq i}}^{r} \frac{t - t_{j+q}}{t_{j+i} - t_{j+q}}.$$

Daher lautet das Verfahren in der LAGRANGE–Form

$$y^{j+k} - y^{j+r-\ell} = h \sum_{i=0}^{r} b_{k,i}^{(r,\ell)} f_{j+i} \tag{4.5.3}$$

mit

$$hb_{k,i}^{(r,\ell)} := \int_{t_{j+r-\ell}}^{t_{j+k}} L_i(t)dt. \tag{4.5.4}$$

Mit der Substitution $t = t_{j+r} + hs$, also $s = h^{-1}(t - t_{j+r})$, mit $dt = hds$ erhält man

$$\int_{t_{j+r-\ell}}^{t_{j+k}} L_i(t)dt = h\int_{-\ell}^{k-r} \prod_{\substack{q=0\\q\neq i}}^{r} \frac{t_0 + (j+r+s)h - t_0 - (j+q)h}{t_0 + (j+i)h - t_0 - (j+q)h} ds = h\int_{-\ell}^{k-r} \prod_{\substack{q=0\\q\neq i}}^{r} \frac{r+s-q}{i-q} ds,$$

also

$$hb_{k,i}^{(r,\ell)} = h\int_{-\ell}^{k-r} \prod_{\substack{q=0\\q\neq i}}^{r} \frac{r+s-q}{i-q} ds.$$

Die Newton–Form des Interpolationspolynoms lautet entsprechend

$$P_{j,r}(t) = \sum_{i=0}^{r} \binom{-s}{i} (-1)^i \nabla_h^i f(t_{j+r}, y(t_{j+r})). \tag{4.5.5}$$

Mit $t = t_0 + (j+r)h + sh = t_{j+r} + sh$ lautet das zugehörige Verfahren

$$y^{j+k} = y^{j+r-\ell} + h\sum_{i=0}^{r} \gamma_{k,i}^{(r,\ell)} \nabla_h^i f_{j+r} \tag{4.5.6}$$

mit

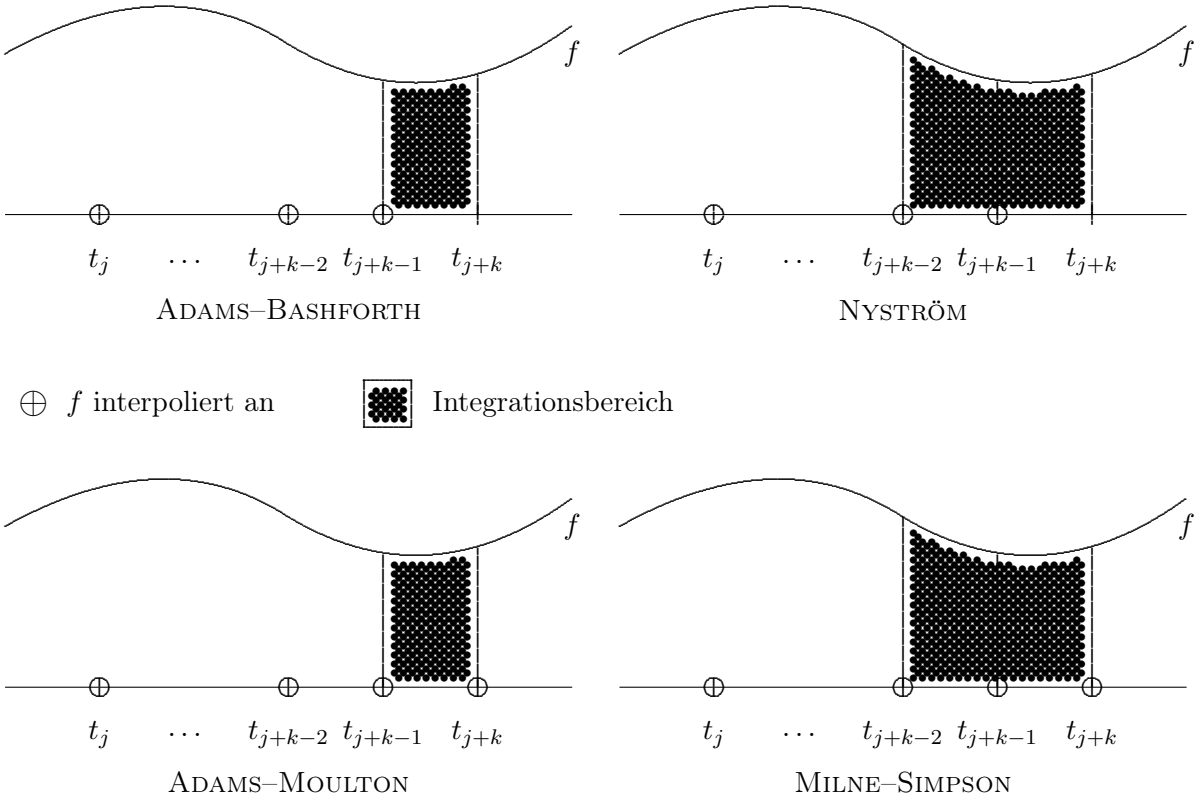
$$\gamma_{k,i}^{(r,\ell)} = (-1)^i \int_{-\ell}^{k-r} \binom{-s}{i} ds. \tag{4.5.7}$$

Vorteile der Newton–Form: Wegen der Restglieddarstellung entspricht der nächste Term der Schätzung des Abschneidefehlers. Das Anhängen höherer Differenzen erlaubt eine Variation der Ordnung. Weiterhin hat diese Form bessere numerische Eigenschaften als die Lagrange–Form, da $\nabla_h^i f_{j+r}$ bei glattem f fallend für steigende i ist (Fehlerdämpfung).

Vorteile der Lagrange–Form: Der Aufwand ist geringer und eine Änderung der Schrittweite lässt sich bequemer realisieren.

Die folgende Tabelle gibt Grundtypen für Verfahren vom Typ (r, ℓ) mit üblichen Wahlen für r, ℓ an:

		r	ℓ
explizite Verfahren	Adams–Bashforth	$k-1$	0
	Nyström	$k-1$	1
implizite Verfahren	Adams–Moulton	k	1
	Milne–Simpson	k	2



Adams–Bashforth

Das Verfahren lautet

$$y^{j+k} = y^{j+k-1} + h \sum_{i=0}^{k-1} b_{k,i} f_{j,i} \tag{4.5.8}$$

mit $b_{k,i} = b_{k,i}^{(k-1,0)}$ aus (4.5.4). Es benötigt pro Schritt nur eine Funktionsauswertung. Einige Koeffizienten sind

	Nenner	i	0	1	2	3	4
$k=1$		$b_{1,i}$	1				
$k=2$	2	$b_{2,i}$	-1	3			
$k=3$	12	$b_{3,i}$	5	-16	23		
$k=4$	24	$b_{4,i}$	-9	37	-59	55	
$k=5$	720	$b_{5,i}$	251	-1274	2616	-2774	1101

Z.B. für $k=2$ ergibt sich als 2. charakteristisches Polynom $\sigma(z) = \frac{1}{2}(3z-1)$, für $k=3$ erhält man $\sigma(z) = \frac{1}{12}(23z^2 - 16z + 5)$.

Für $r = k-1 = 0$ erhält man das explizite Euler–Verfahren.

Nyström

Das Verfahren lautet

$$y^{j+k} = y^{j+k-2} + h \sum_{i=0}^{k-1} b_{k,i}^{(k-1,1)} f_{i+j}. \tag{4.5.9}$$

Für $r = 0$, d.h. $k = 1$ gilt

$$y^{j+1} = y^{j-1} + 2hf_j,$$

was gerade die Mittelpunktsregel ist.

Adams–Moulton

Mit $b_{k,i} = b_{k,i}^{(k,1)}$ hat dieses Verfahren die Form

$$y^{j+k} = y^{j+k-1} + h\left(\sum_{i=0}^{k} b_{k,i} f_{j+i}\right). \tag{4.5.10}$$

Die Koeffizienten werden ebenso aus (4.5.4) berechnet. Man erhält

	Nenner	i	0	1	2	3	4
$k=0$	1	$b_{0,i}$	1				
$k=1$	2	$b_{1,i}$	1	1			
$k=2$	12	$b_{2,i}$	-1	8	5		
$k=3$	24	$b_{3,i}$	1	-5	19	9	
$k=4$	720	$b_{4,i}$	-19	106	-264	646	251

Speziell für $r = k = 0$ ergibt sich das implizite Euler–Verfahren

$$y^j = y^{j-1} + hf_j,$$

und für $r = k = 1$ die Trapezregel,

$$y^{j+1} = y^j + h\left(\frac{1}{2}f_j + \frac{1}{2}f_{j+1}\right).$$

Milne–Simpson

Hier ist

$$y^{j+k} = y^{j+k-2} + h\sum_{i=0}^{k} b_{k,i} f_{j+i} \tag{4.5.11}$$

mit entsprechenden Koeffizienten $b_{k,i} := b_{k,i}^{(k,2)}$ gemäß folgender Tabelle.

	Nenner	i	0	1	2	3	4
$k=0$	1	$b_{0,i}$	2				
$k=1$	1	$b_{1,i}$	2	0			
$k=2$	3	$b_{2,i}$	1	4	1		
$k=4$	90	$b_{4,i}$	-1	4	4	44	29

Speziell für $r = k = 0$ ergibt sich das implizite Euler–Verfahren mit Schrittweite $2h$,

$$y^j = y^{j-2} + 2hf_j,$$

und für $r = k = 1$ die Mittelpunktsregel

$$y^{j+1} = y^{j-1} + 2h\, f_j.$$

Das Verfahren für $r = k = 2$ bezeichnet man auch als Milne–Verfahren, für das sich Ordnung 4 zeigen lässt.

Konsistenzordnungen

Satz 4.5.12. Falls $f \in \mathcal{C}^{r+1}(U)$ ist, sind die Mehrschrittverfahren des Typs (r, ℓ) konsistent mit der Anfangswertaufgabe von der Ordnung $p = r + 1$, d.h.

(i) Adams–Bashforth $p = k$;

(ii) Nyström $p = k$ (für $k > 2$; Mittelpunktsregel hat Ordnung 2);

(iii) Adams–Moulton $p = k + 1$;

(iv) Milne–Simpson $p = k + 1$ (für $k > 3$, bei $k = 2$ lässt sich Ordnung 4 erreichen).

Beweis: Ist die Lösung der Anfangswertaufgabe ein Polynom vom Grad $\leq r + 1$, so liefert das Verfahren wegen (4.5.2) (Interpolationspolynom) die exakte Lösung, da die rechte Seite dann vom Grad $\leq r$ ist und exakt interpoliert wird. Daher folgt die Behauptung mit Satz 4.3.8. ∎

Alternativ kann man auch direkt die Restgliedformel der Polynominterpolation benutzen. Zur Übung bestimme man die Fehlerkonstanten (4.3.13) und (4.3.14) für die Verfahren von Adams–Bashforth, Nyström, Adams–Moulton und Milne–Simpson.

4.6 Prädiktor–Korrektor–Verfahren

Bei impliziten Verfahren kann y^{j+k} im Allgemeinen nur iterativ bestimmt werden. Dazu geht man folgendermaßen vor. Zunächst bestimmt man mit einem *expliziten Verfahren (Prädiktor)* einen Startwert $y^{j+k,0}$ für das iterative Verfahren. Danach soll durch eine Fixpunktiteration mit einem *impliziten Verfahren (Korrektor)* dieser Wert verbessert werden. Die $i$te Iterierte heiße $y^{j+k,i}$. Mit den Bezeichnungen

$$\begin{aligned}
&\text{(A): Auswertung von } f(t_{j+k}, y^{j+k,i}) \\
&\text{(P): Anwendung des Prädiktors} \\
&\text{(K): Anwendung des Korrektors}
\end{aligned}$$

lautet für $m \in \{1, 2, 3\}$ das Vorgehen

(I) $P(AK)^m$ oder

(II) $P(AK)^m A$.

Bei (I) wird mit $f(t_{j+k}, y^{j+k,m-1})$, bei (II) mit $f(t_{j+k}, y^{j+k,m})$ weitergearbeitet. Sind genauer die Verfahren

$$\begin{aligned}
\text{Prädiktor:} \quad \sum_{i=0}^{k} a_i^* y^{j+i} &= h \sum_{i=0}^{k-1} b_i^* f_{j+i}, \quad \text{mit } a_k^* = 1, \\
\text{Korrektor:} \quad \sum_{i=0}^{k} a_i y^{j+i} &= h \sum_{i=0}^{k} b_i f_{j+i}, \quad \text{mit } a_k = 1,
\end{aligned}$$

so ergibt sich für $P(AK)^m A$

$$\text{(P)} \qquad y^{j+k,0} + \sum_{i=0}^{k-1} a_i^* y^{j+i,m} = h \sum_{i=0}^{k-1} b_i^* f_{j+i,m}$$

für $s = 0, \ldots, m - 1$

$$\text{(A)} \qquad f_{j+k,s} = f(t_{j+k}, y^{j+k,s})$$

$$\text{(K)} \qquad y^{j+k,s+1} + \sum_{i=0}^{k-1} a_i y^{j+i,m} = h b_k f_{j+k,s} + \sum_{i=0}^{k-1} b_i f_{j+i,m}$$

$$\text{(A)} \qquad f_{j+k,m} = f(t_{j+k}, y^{j+k,m})$$

(entsprechend für $P(AK)^m$).

Konsistenz der Prädiktor–Korrektor–Verfahren

Satz 4.6.1. Sei p^* bzw. p die Konsistenzordnung des Prädiktors bzw. Korrektors. Sei f in einer Umgebung der Lösung der Anfangswertaufgabe genügend glatt, um Ordnungen p und p^* zuzulassen. Dann gilt für die Ordnung $p = p(P, K)$ des Prädiktor–Korrektor–Verfahrens $P(AK)^m A$ bzw. $P(AK)^m$

$$p(P,K) = \begin{cases} p, & \text{falls } p^* \geq p \text{ oder } m \geq p - p^* + 1 \\ & \text{(mit gleicher Fehlerkonstante wie bei Korrektor)} \\ p, & \text{falls } m = p - p^* > 0 \text{ (mit einer anderen Fehlerkonstanten)} \\ p^* + m, & \text{falls } m \leq p - p^* - 1 \end{cases},$$

d.h. $p(P, K) = \min\{p, p^* + m\}$. Also kann die Ordnung des Prädiktors kleiner als die des Korrektors gewählt werden. ∎

4.7 Konvergenz und Stabilität von Mehrschrittverfahren

4.7.1 Vorbemerkungen

Bei Einschrittverfahren folgte aus der Konsistenz und einer Lipschitz–Bedingung für Φ die Konvergenz des Verfahrens. Ebenso ergab die Konsistenzordnung zusammen mit der Lipschitz–Bedingung die Konsistenzordnung. Im Folgenden wird untersucht, ob dies auch bei Mehrschrittverfahren gilt.

Beispiel 4.7.1. Für das explizite Zweischrittverfahren

$$y^{j+2} + 4y^{j+1} - 5y^j = h(4f_{j+1} + 2f_j)$$

gilt

$$\begin{aligned} \rho(z) &= z^2 + 4z - 5, \qquad \rho'(z) = 2z + 4 \\ \sigma(z) &= 4z + 2, \end{aligned}$$

und damit $\rho(1) = 0,\ \rho'(1) = 6 = \sigma(1)$. Also sind die Bedingungen in (4.3.7) erfüllt, und Konsistenz liegt vor. Mit (4.3.9) prüft man, dass Konsistenzordnung $p = 3$ vorliegt. Zur *Konvergenzfrage* betrachten wir die spezielle Anfangswertaufgabe

$$y' = -y, \quad t \geq 0, \quad y(0) = 1,$$

dessen Lösung $y(t) = e^{-t}$ ist. Als Startwerte seien die exakten Werte $y^0 = 1$ und $y^1 = e^{-h}$ vorgegeben. Das Verfahren hat für diese Aufgabe die Form

$$y^{j+2} + 4y^{j+1} - 5y^j = h(-4y^{j+1} - 2y^j),$$

was äquivalent zur linearen homogenen *Differenzengleichung*

$$y^{j+2} + 4(1+h)y^{j+1} + (-5+2h)y^j = 0 \tag{4.7.2}$$

ist. Zur Bestimmung der Lösung mit obigen Startwerten machen wir den Ansatz $y^j = z^j$ (Monom in z vom Grad j; zur allgemeinen Lösungstheorie siehe Abschnitt 4.7.2), und erhalten

$$z^j(z^2 + 4(1+h)z + (-5+2h)) = 0,$$

was unter der Annahme $z \neq 0$

$$z^2 + 4(1+h)z + (-5+2h) = 0$$

impliziert. Die Lösungen dieser Gleichung lauten

$$\begin{aligned} z_1 &= -2(1+h) + 3\sqrt{1 + \frac{2}{3}h + \frac{4}{9}h^2} \\ z_2 &= -2(1+h) - 3\sqrt{1 + \frac{2}{3}h + \frac{4}{9}h^2}. \end{aligned}$$

Setzt man nun $t := \frac{2}{3}h + \frac{4}{9}h^2$, so kann man die Lösungen unter Ausnutzung von

$$(1+t)^\alpha = \sum_{j=0}^{\infty} \binom{\alpha}{j} t^j, \quad \alpha \in \mathbb{R}$$

mit $\binom{\alpha}{1} = \alpha$, $\binom{\alpha}{2} = \frac{\alpha(\alpha-1)}{2}, \ldots$ folgendermaßen für $\alpha = \frac{1}{2}$ entwickeln:

$$\begin{aligned} z_1 &= -2 - 2h + 3\left(\left(1 + \frac{2}{3}h + \frac{4}{9}h^2\right)^{1/2}\right) \\ &= -2 - 2h + 3\left(1 + \frac{1}{2}\left(\frac{2}{3}h + \frac{4}{9}h^2\right) + \frac{1}{2}\left(\frac{1}{2} - 1\right)\left(\frac{1}{3}h + \frac{2}{9}h^2\right)^2 + \ldots\right) \\ &= 1 - h + \frac{h^2}{2} - \frac{h^3}{6} + \frac{h^4}{72} + O(h^5) \qquad \longrightarrow 1 \text{ für } h \to 0 \qquad (4.7.3) \\ z_2 &= -2 - 2h - 3\left(\left(1 + \frac{2}{3}h + \frac{4}{9}h^2\right)^{1/2}\right) \\ &= -5 - 3h + O(h^2) \qquad \longrightarrow -5 \text{ für } h \to 0. \end{aligned}$$

Beachte, dass die Grenzwerte von z_1, z_2 für $h \to 0$ gerade die Nullstellen des ersten charakteristischen Polynoms $\rho(z) = z^2 + 4z - 5 = (z-1)(z+5)$ sind.

Da $(z_1)^j$ und $(z_2)^j$ Lösungen der homogenen Differenzengleichung sind, löst jede Linearkombination dieser die Differenzengleichung (4.7.2). Daher ist y^j von der Form

$$y^j = c_1(z_1)^j + c_2(z_2)^j.$$

Mit den Anfangsbedingungen $y^0 = 1$ und $y^1 = e^{-h}$ gilt

$$1 = c_1 + c_2 \qquad \text{und} \qquad e^{-h} = c_1 z_1 + c_2 z_2,$$

was

$$c_1 = \frac{z_2 - e^{-h}}{z_2 - z_1} \qquad \text{und} \qquad c_2 = \frac{e^{-h} - z_1}{z_2 - z_1}$$

impliziert. Mit (4.7.3) gilt nun

$$c_1 = 1 + O(h^2) \qquad \text{und} \qquad c_2 = -\frac{1}{36}h^4 + O(h^5).$$

Damit folgt für $h = \frac{t}{j}$ mit $j = 1, 2, \ldots$ und t als Endzeit

$$\begin{aligned} y^j &= c_1(z_1)^j + c_2(z_2)^j \\ &= \left(1 + O\left(\left(\frac{t}{j}\right)^2\right)\right) \cdot \left[1 - \frac{t}{j} + O\left(\left(\frac{t}{j}\right)^2\right)\right]^j \\ &\quad -\frac{1}{36}\left(\frac{t}{j}\right)^4 \left(1 + O\left(\frac{t}{j}\right)\right) \left[-5 - 3\frac{t}{j} + O\left(\left(\frac{t}{j}\right)^2\right)\right]^j. \end{aligned}$$

Der erste Summand konvergiert (wegen der Form des zweiten Faktors) für $j \to \infty$ gegen $(e^{-t/j})^j = e^{-t}$, der zweite oszilliert gegen $-\frac{1}{36}\frac{(-5)^j}{j^4}t^4 e^{3t/5}$, also gegen $\pm\infty$, da $\frac{5^j}{j^4} \to \infty$ für $j \to \infty$. Insgesamt dominiert der zweite Term über dem ersten, obwohl das Verfahren wie die exakte Lösung $y(t) = e^{-t}$ abklingen sollte. Dies liegt daran, dass das erste charakteristische Polynom $\rho(z) = z^2 + 4z - 5$ außer der *Hauptwurzel* 1 noch die Wurzel -5 hat. Durch die Approximation der Ableitung durch höhere Differenzen schleichen sich zusätzliche Lösungswege ein, die von *parasitären* Wurzeln von $\rho(z)$ (spurious roots) herrühren. Dies kann bei Einschrittverfahren wegen $\rho(z) = z - 1$ nicht passieren.

Wichtig zu bemerken ist, dass dies ein Fehler des Verfahrens und nicht der Differentialgleichung ist! ∎

Vorüberlegung zur Stabilität

Betrachte das lineare Mehrschrittverfahren (4.1.4)

$$\sum_{\ell=0}^{k} a_\ell y^{j+\ell} = h \sum_{\ell=0}^{k} b_\ell f_{j+\ell},$$

welches konsistent mit der Anfangswertaufgabe (4.1.1) für alle glatten f sei. Dann ist das Mehrschrittverfahren konsistent mit der speziellen Anfangswertaufgabe $y' = 0$ mit $y(0) = 1$. Für diese Anfangswertaufgabe erfüllt die Folge $\{y^j\}_{j=0,\ldots,\infty}$ der Näherungswerte aus (4.1.4) die Gleichung

$$\sum_{\ell=0}^{k} a_\ell y_{j+\ell} = 0, \quad j = 0, 1, 2, \ldots \tag{4.7.4}$$

(Zur Unterscheidung zu den Näherungswerten $y^j \approx y(t_j)$ seien die Folgen, die die Differenzengleichung (4.7.4) lösen, mit y_j bezeichnet.) (4.7.4) ist eine *lineare homogene* Differenzengleichung k–ter Ordnung mit *konstanten* Koeffizienten.

Bemerkung 4.7.5. Damit das Mehrschrittverfahren (4.1.4) konvergiert, muss $\{y^j\}_{j=0,\ldots,\infty}$ *beschränkt* bleiben, wenn die exakte Lösung beschränkt ist. Das Mehrschrittverfahren heißt dann *stabil*. ∎

4.7.2 Einige Bemerkungen zur Theorie linearer Differenzengleichungen

Wir untersuchen die Lösungsstruktur für die homogene lineare Differenzengleichung (4.7.4),

$$a_k y^{j+k} + \ldots + a_0 y_j = 0.$$

Für den allgemeinen Fall $y_j \in \mathbb{R}^n$ kann man sich auf eine komponentenweise Betrachtung zurückziehen. Daher sei in diesem Abschnitt o.B.d.A. $n = 1$. Weiter sei bei einer Gleichung $k$ter Ordnung vorausgesetzt, dass der führende Koeffizient $a_k \neq 0$, d.h. o.B.d.A. normiert, also $a_k = 1$, sei.

Folgen $Y := \{y_j\}_{j=0}^{\infty}$, die (4.7.4) für alle $j \in \mathbb{Z}^+$ erfüllen, heißen *Lösungen* von (4.7.4).

Bemerkung 4.7.6.

(a) Die Lösungen von (4.7.4) bilden einen linearen Raum.

(b) Jede Lösung ist durch k Werte $y_0, \ldots, y_{k-1}$ eindeutig festgelegt.

(c) Die Dimension des Lösungsraums ist genau k.

Beweis: (a) trivial, da eine homogene lineare Differenzengleichung vorliegt.
(b) aus $a_k = 1$ folgt $y_{j+k} = -a_0 y_j - \ldots - a_{k-1} y_{j+k-1}$ für $j = 0, 1, 2, \ldots$.
(c) wähle eine beliebige Basis des $\mathbb{R}^k$ als Startvektoren. ∎

Zur Darstellung von Y das folgende

Lemma 4.7.7. Jede Lösung $Y = \{y_j\}_{j=0}^{\infty}$ von (4.7.4) hat eine eindeutige Entwicklung der Form

$$Y = \sum_{p=1}^{m} \sum_{q=0}^{\mu_p - 1} c_{p,q} Y^{(p,q)}, \tag{4.7.8}$$

wobei

$$\rho(z) = (z - z_1)^{\mu_1} \cdot \ldots \cdot (z - z_m)^{\mu_m} \tag{4.7.9}$$

die Zerlegung des ersten charakteristischen Polynoms mit $z_j \in \mathbb{C}$ und $\mu_1 + \ldots + \mu_m = k$ und

$$Y^{(p,q)} := \left\{ j(j-1) \cdot \ldots \cdot (j-q+1) z_p^{j-q} \right\}_{j=0}^{\infty} \tag{4.7.10}$$

zur p–ten Nullstelle und q–ten Ableitung sind, d.h.

$$Y_j^{(p,0)} = (z_p)^j, \ j \in \mathbb{Z}_0^+, \quad \text{und} \quad Y_j^{(p,q)} = 0 \ \text{ für } q > 0 \text{ und } j = 0, \ldots, q-1 \text{ (Anfangswerte).} \tag{4.7.11}$$

Beachte: $Y_j^{(p,q)} = j(j-1) \cdot \ldots \cdot (j-q+1) z_p^{j-q}$ ist also das $j$te Folgenglied der Folge $Y^{(p,q)}$. Wegen der Darstellung (4.7.10) spielen nur die $Y^{(p,q)}$ für $z_p \neq 0$ eine Rolle. Die Anteile bzgl. $z_p = 0$ außer bei $0^0 = 1$ fallen aus dem Differenzenverfahren heraus.

Beweis: Wir zeigen zuerst, dass jede Folge $Y^{(p,q)}$ Lösung von (4.7.4) ist. Dazu sei $0 \leq q \leq \mu_p - 1$. z_p ist eine μ_p–fache Nullstelle von $\rho(z)$. Daher gilt

$$\begin{aligned} 0 = \left(\frac{d}{dz}\right)^q \rho(z)\Big|_{z=z_p} &= \left(\left(\frac{d}{dz}\right)^q \sum_{\ell=0}^{k} a_\ell z^\ell \right)\Big|_{z=z_p} \\ &= \sum_{\ell=0}^{k} a_\ell \ell(\ell-1)\ldots(\ell-q+1)(z_p)^{\ell-q} \\ &= \sum_{\ell=0}^{k} a_\ell Y_\ell^{(p,q)}. \end{aligned}$$

Also erfüllen die ersten k Folgenglieder von $Y^{(p,q)}$ die Differenzengleichung (4.7.4). Genauso ist z_p μ_p–fache Nullstelle von $z^j\rho(z)$ für alle $j \in \mathbb{Z}_0^+$, d.h.

$$0 = \sum_{\ell=0}^{k} a_\ell Y_{j+\ell}^{(p,q)}, \qquad j \geq 0.$$

Daher ist $Y^{(p,q)}$ Lösung von (4.7.4). Wegen Bemerkung 4.7.5 genügt es zu zeigen, dass die k Lösungsfolgen $Y^{(p,q)}$ linear unabhängig sind. Dazu genügt es nach Bemerkung 4.7.6 (b) zu zeigen, dass die k–Startabschnitte linear unabhängig sind. Wir beobachten, dass $Y^{(p,q)}$ durch den Startvektor

$$\xi^{(p,q)} = \left(\frac{d}{dz}\right)^q \begin{pmatrix} 1 \\ z \\ z^2 \\ \vdots \\ z^{k-1} \end{pmatrix}\Bigg|_{z=z_p} \tag{4.7.12}$$

erzeugt wird (siehe (4.7.11)). Betrachte die Matrix

$$M = \left(\xi^{(1,0)}, \ldots, \xi^{(1,\mu_1-1)}, \ldots, \xi^{(m,0)}, \ldots, \xi^{(m,\mu_m-1)}\right).$$

M ist gerade die Transponierte der Kollokationsmatrix des HERMITE–Interpolationsproblems, bei dem es gilt, $P \in \mathcal{P}_{k-1}$ mit $P^{(j)}(z_p) = g^{(j)}(z_p)$ für $j = 0, \ldots, \mu_p - 1$ und $p = 1, \ldots, m$ zu finden. Dies hat nach dem Satz über die „allgemeine HERMITE–Interpolation für Polynome“ eine eindeutige Lösung. Also ist $\det M \neq 0$, d.h. $\xi^{(p,q)}$ linear unabhängig. ∎

Bemerkung 4.7.13. Die Lösungen der Differenzengleichung (4.7.4) sind alle genau dann beschränkt, wenn $\rho(z)$ die DAHLQUIST*'sche Wurzelbedingung*

$$\begin{aligned} \rho(z) &= 0 \implies |z| \leq 1 \\ \rho(z) &= 0,\ |z| = 1 \implies z \text{ ist } \textit{einfache} \text{ Nullstelle} \end{aligned} \tag{WB}$$

erfüllt.

Beweis: Die Bemerkung sagt, dass (WB) äquivalent dazu ist, dass alle $Y^{(p,q)}$ beschränkt sind. Nach Lemma 4.7.7 gilt, falls die Vielfachheit $\mu_p \geq 1$ ist, $|Y_j^{(p,0)}| = |z_p|^j \leq C$ für alle j, was äquivalent zu $|z_p| \leq 1$ ist. Falls $\mu_p > 1$, gilt $|Y_j^{(p,q)}| = |j(j-1)\ldots(j-q+1)||z_p|^{j-q} \leq C$ für alle j, was äquivalent zu $|z_p| < 1$ ist. ∎

Jetzt haben wir alle Hilfsmittel zusammen, um die Konvergenz von Mehrschrittverfahren zu betrachten. Wir können also der Frage nachgehen, wie sich Fehler (Diskretisierungsfehler, Rundungsfehler) im Lauf der Rechnung kontrollieren lassen.

4.7.3 Konvergenz von Mehrschrittverfahren

Zur Anfangswertaufgabe (4.1.1) betrachte das allgemeine Mehrschrittverfahren

$$\begin{aligned} \sum_{\ell=0}^{k} a_\ell y^{j+\ell} &= h\Phi(t_j, \ldots, t_{j+k}, y^j, \ldots, y^{j+k}, h) + h\varepsilon^{j+k}(h) \\ y_h^j &= y_h^j(\varepsilon) = y(t_j) + \varepsilon^j(h), \quad j = 0, \ldots, k-1 \\ t_j &= t_0 + jh, \quad t_N = t_0 + Nh. \end{aligned} \tag{4.7.14}$$

wobei t_N der Endzeitpunkt sei. Beachte, dass Fehler/Störungen in den Anlaufwerten und im Lauf der Rechnung z.B. durch Rundung berücksichtigt werden.

Definition: Das durch (4.7.14) gegebene allgemeine Mehrschrittverfahren heißt *konvergent*, falls für alle Startwerte $y_h^j = y_h^j(\varepsilon)$ für $j = 0, \ldots, k-1$ mit $\|y_h^j - y^0\| \to 0$, $h \to 0$, und alle Fehler mit

$$\|\varepsilon^j(h)\| \leq \Psi(h) = o(1), \quad h \to 0, \quad j \leq N \to \infty \tag{4.7.15}$$

gilt:

$$\lim_{N\to\infty} \max_{0\leq j\leq N} \|y^j - y(t_j)\| = 0. \tag{4.7.16}$$

Satz 4.7.17. Φ erfülle in einer Umgebung U der Lösung der Anfangswertaufgabe die Bedingung

$$f(t, u(t)) = 0 \quad \Longrightarrow \quad \Phi(t_j, \ldots, t_{j+k}, \eta^j, \ldots, \eta^{j+k}, h) = 0 \tag{4.7.18}$$

für alle $(t_{j+l}, \eta^{j+l}) \in U$. Dann impliziert die Konvergenz des Mehrschrittverfahrens (4.7.14), dass $\rho(z)$ die Wurzelbedingung (WB) erfüllt.

Beachte, dass Bedingung (4.7.18) etwa für lineare MSV erfüllt ist.

Beweis: Die Konvergenz des Mehrschrittverfahrens impliziert insbesondere für $y' = 0$ mit $y(0) = 1$ und (4.7.18), dass die Näherungslösungen mit $\varepsilon^j(h) = 0$, $j \geq k$, also keinem Fehler auf der rechten Seite, die homogene Differenzengleichung (4.7.14) erfüllen. Aus der Konvergenz des Mehrschrittverfahrens folgt, dass die Lösungen für *alle* Startwerte beschränkt sind. Also folgt die Behauptung mit Bemerkung 4.7.13. ∎

Wir wollen nun die Umkehrung des Satzes (unter weiteren Bedingungen) zeigen. Dazu benötigen wir zunächst das folgende Lemma, das wir schon in ähnlicher Form im ersten Teil des Skriptes als Korollar **??** kennengelernt haben.

Lemma 4.7.19. Zu jedem $A \in \mathbb{C}^{n\times n}$ und jedem $\varepsilon > 0$ existiert eine (Vektor–)Norm $\|\cdot\|$ auf $\mathbb{C}^n$, so dass für die induzierte Matrixnorm

$$\|A\| := \sup_{\|x\|=1} \|Ax\|$$

gilt

$$\|A\| \leq r(A) + \varepsilon.$$

Falls ferner für jeden Eigenwert λ mit $|\lambda| = r(A)$ gilt, dass λ nur lineare Elementarteiler hat, dann gilt sogar $\|A\| \leq r(A)$. Da man andererseits stets $r(A) \leq \|A\|$ hat, folgt in diesem Fall

$$r(A) = \|A\|.$$

∎

Außerdem benötigen wir noch eine Bedingung an die Verfahrensfunktion Φ: In einer Umgebung U der exakten Lösung und $f \in \mathcal{C}^1(I \times \mathbb{R}^n)$, $([t_0, t_N] \subseteq I)$ gelte die Lipschitz–Bedingung

$$\|\Phi(t_i, \ldots, t_{i+k}, \eta^0, \ldots, \eta^k, h) - \Phi(t_i, \ldots, t_{i+k}, \zeta^0, \ldots, \zeta^k, h)\| \leq c \sum_{\ell=0}^{k} \|\eta^\ell - \zeta^\ell\| \tag{4.7.20}$$

mit $h \leq h_0$ für alle (t_{i+j}, η^j), $(t_{i+j}, \zeta^j) \in U$.

Satz 4.7.21. Das allgemeine Mehrschrittverfahren (4.7.14) mit Störungen sei *konsistent* und Φ erfülle (4.7.18) und (4.7.20). Dann ist das Mehrschrittverfahren für alle $f \in \mathcal{C}^1$ genau dann konvergent, wenn die Wurzelbedingung (WB) erfüllt ist.

Beweis: Die Notwendigkeit von (WB) wurde bereits in Satz 4.7.17 gezeigt.

Zum Beweis der Hinlänglichkeit bezeichne $y(t)$ die exakte Lösung. Für $e^j := y^j - y(t_j)$ gilt nach Definition (4.2.1)

$$\begin{aligned}
\sum_{\ell=0}^{k} a_\ell e^{j+\ell} &= h[\Phi(t_j,\dots,t_{j+k},y^j,\dots,y^{j+k},h) \\
&\quad -\Phi(t_j,\dots,t_{j+k},y(t_j),\dots,y(t_{j+k}),h)] + h(\varepsilon^{j+k}(h) - \tau(t_j,y,h)) \\
&=: h[\Phi(t_j,\dots,t_{j+k},y^j,\dots,y^{j+k},h) - \Phi(t_j,\dots,t_{j+k},y(t_j),\dots,y(t_{j+k}),h)] \\
&\quad +h(\varepsilon^{j+k} - \tau^j) \\
&=: g^{j+k}.
\end{aligned} \tag{4.7.22}$$

Mit der Lipschitz–Bedingung (4.7.20) folgt zunächst

$$\begin{aligned}
\|g^{j+k}\| &\le ch\sum_{\ell=0}^{k}\|y^{j+\ell} - y(t_{j+\ell})\| + h\left(\|\varepsilon^{j+k}\| + \|\tau^{j+k}\|\right) \\
&= ch\sum_{\ell=0}^{k}\|e^{j+\ell}\| + h\left(\|\varepsilon^{j+k}\| + \|\tau^{j+k}\|\right).
\end{aligned} \tag{4.7.23}$$

Unser Fernziel ist die Anwendung der diskreten Gronwall–Ungleichung. Wir schreiben dazu die Differenzengleichung $k$ter Ordnung in ein System von k Gleichungen erster Ordnung, d.h. wir definieren Matrizen

$$E^j := \begin{pmatrix} (e^j)^T \\ \vdots \\ (e^{j+k-1})^T \end{pmatrix}, \quad G^{j+k} := \begin{pmatrix} 0 \\ \vdots \\ 0 \\ (g^{j+k})^T \end{pmatrix} \in \mathbb{R}^{k\times n}, \quad A := \begin{pmatrix} 0 & 1 & & & \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ & & & 0 & 1 \\ -a_0 & -a_1 & \cdots & \cdots & -a_{k-1} \end{pmatrix}. \tag{4.7.24}$$

Dann ist (4.7.22) äquivalent zu

$$E^{j+1} = AE^j + G^{j+k}. \tag{4.7.25}$$

Beachte, dass die Eigenwerte von A genau die Nullstellen von $\rho(z)$ sind. Da nach Voraussetzung die Wurzelbedingung (WB) gilt, ist der betragsgrößte Eigenwert $|\lambda_{\max}| = 1$ von A einfach. Also hat $\lambda_{\max}$ nur einen linearen Elementarteiler. Somit existiert nach Lemma 4.7.19 eine Vektornorm $\|\cdot\|$ mit

$$\|A\| = r(A) = 1 \tag{4.7.26}$$

für die zugehörige Matrixnorm. Fasse nun E^j, G^j als Elemente von $\mathbb{R}^{nk}$ auf. Definiere die Norm

$$|\!|\!| E^j |\!|\!| := \max_{i=1,\dots,n} \|E^{j,(i)}\|,$$

wobei $E^{j,(i)}$ die i–te Spalte von E^j bezeichne und $\|\cdot\|$ hier die spezielle oben eingeführte Vektornorm ist. Wegen (4.7.26) gilt dann

$$|\!|\!| AE^j |\!|\!| = \max_{i=1,\dots,n} \|AE^{j,(i)}\| \le \|A\| \max_{i=1,\dots,n} \|E^{j,(i)}\| = \|A\| |\!|\!| E^j |\!|\!| = |\!|\!| E^j |\!|\!|, \tag{4.7.27}$$

d.h. A ist eine schwache Kontraktion. Damit folgt aus (4.7.25)

$$\|\!|E^{j+1}|\!\| \leq \|A\| \|\!|E^j|\!\| + \|\!|G^{j+k}|\!\| = \|\!|E^j|\!\| + \|\!|G^{j+k}|\!\|. \tag{4.7.28}$$

Des Weiteren folgt aus der Äquivalenz von Normen auf endlichdimensionalen Räumen

$$\sum_{\ell=0}^{k} \|e^{j+\ell}\| \leq c \left(\|\!|E^j|\!\| + \|\!|E^{j+1}|\!\| \right),$$

und aus (4.7.23) (mit einer anderen Konstante c)

$$\|\!|G^{j+k}|\!\| \leq ch \left(\|\!|E^j|\!\| + \|\!|E^{j+1}|\!\| + \|\varepsilon^{j+k}\| + \|\tau^j\| \right).$$

Somit folgt für (4.7.28)

$$\|\!|E^{j+1}|\!\| \leq \|\!|E^j|\!\| + ch \left(\|\!|E^j|\!\| + \|\!|E^{j+1}|\!\| + \|\varepsilon^{j+k}\| + \|\tau^j\| \right),$$

d.h. für genügend kleines h (so dass $(1 - ch) > 0$) gilt

$$(1 - ch) \|\!|E^{j+1}|\!\| \leq (1 + ch) \|\!|E^j|\!\| + ch \left(\|\varepsilon^{j+k}\| + \|\tau^j\| \right),$$

und damit

$$\|\!|E^{j+1}|\!\| \leq \frac{1 + ch}{1 - ch} \|\!|E^j|\!\| + \frac{ch}{1 - ch} \left(\|\varepsilon^{j+k}\| + \|\tau^j\| \right).$$

Um die diskrete Gronwall–Ungleichung anwenden zu können, entwickeln wir $\frac{1+ch}{1-ch}$. Es gilt

$$\frac{1 + x}{1 - x} \leq 1 + 4x \text{ für } x \leq \frac{1}{2},$$

denn

$$\begin{aligned} \frac{1 + x}{1 - x} &= (1 + x) \left(\sum_{i=0}^{\infty} x^i \right) = \sum_{i=0}^{\infty} x^i + \sum_{i=0}^{\infty} x^{i+1} = \sum_{i=0}^{\infty} x^i + \sum_{i=1}^{\infty} x^i \\ &= 1 + 2 \sum_{i=1}^{\infty} x^i = 1 + 2 \frac{x}{1 - x} \leq 1 + 4x \quad \text{für } x \leq \frac{1}{2}. \end{aligned}$$

Damit folgt

$$\|\!|E^{j+1}|\!\| \leq (1 + 4ch) \|\!|E^j|\!\| + ch(1 + 4ch) \left(\|\varepsilon^{j+k}\| + \|\tau^j\| \right).$$

Anwendung der diskreten Gronwall–Ungleichung mit

$$z_j := \|\!|E^{j+1}|\!\|, \quad \delta_j := 4ch, \quad \eta_j := ch(1 + 4ch) \left(\|\varepsilon^{j+k}\| + \|\tau^j\| \right)$$

liefert

$$\begin{aligned} \|\!|E^j|\!\| &\leq \left(\|\!|E^0|\!\| + \sum_{\ell=0}^{j-1} ch(1 + 4ch) \left(\|\varepsilon^{\ell+k}\| + \|\tau^\ell\| \right) \right) e^{4chj} \\ &\leq \left(\|\!|E^0|\!\| + c \max_{\ell \leq j} \left(\|\varepsilon^{\ell+k}\| + \|\tau^\ell\| \right) \right) e^{4c(t_N - t_0)}, \end{aligned}$$

da $jh \leq t_N - t_0 = Nh$ für $j \leq N$ und $\sum_{\ell=0}^{j-1} ch(1 + 4ch) = jhc(1 + 4ch) \leq \tilde{c}$ für $j \leq N$ ist. Aufgrund der Konsistenz folgt $\|\!|E^0|\!\| \to 0$ und $\max_{\ell \leq j} \|\tau^\ell\| \to 0$ für $h \to 0$ und jedes j. Ist also $\{\varepsilon^j\}_j$ Nullfolge für $h \to 0$, so folgt insgesamt $\|\!|E^j|\!\| \to 0$, $j \leq N \to \infty$, da $e^{4c(t_N - t_0)}$ konstant ist. Also ist das Mehrschrittverfahren konvergent. ∎

Korollar 4.7.29. Es gelten die Voraussetzungen aus Satz 4.7.21, und das allgemeine Mehrschrittverfahren (4.7.14) sei konsistent von der Ordnung p. Sind auch die Störungen $\varepsilon^j(h)$ von der Ordnung p, so ist das Mehrschrittverfahren *konvergent* von der Ordnung p. ∎

Wenn ρ die Wurzelbedingung (WB) erfüllt, nennt man das Mehrschrittverfahren $\emptyset$*–stabil* (*Null–stabil,* LIPSCHITZ*–stabil, asymptotisch stabil*). Beachte, dass die Stabilität im Wesentlichen von ρ und fast nicht von Φ abhängt. Daher heißt (WB) auch *Stabilitätsbedingung.*

In Satz 4.7.21 wurde gezeigt, dass Konsistenz und Stabilität Konvergenz nach sich ziehen. Wir fragen nun, ob Konvergenz auch immer Konsistenz impliziert. Für lineare Mehrschrittverfahren lässt sich zeigen:

Satz 4.7.30. Konvergiert ein lineares Mehrschrittverfahren für alle Anfangswertaufgaben mit $f \in \mathcal{C}^1$, dann ist es auch konsistent.

Beweis: Betrachte die spezielle Anfangswertaufgabe $y' = 0$ mit $y(t_0) = 1$. Die exakte Lösung lautet $y \equiv 1$. Zu Startwerten $y^j = 1$ für $j = 0, \ldots, k-1$ liefert das Verfahren Werte $y^j = y_h(t_j),\ j \geq k$, mit

$$\sum_{\ell=0}^{k} a_\ell y^{j+\ell} = 0.$$

Wegen $y_h(t_j) \to y(\bar{t})$ für $t_j \to \bar{t}$ (und $y \equiv 1$) folgt

$$\sum_{\ell=0}^{k} a_\ell = 0 = \rho(1).$$

Zum Nachweis der zweiten Bedingung $\rho'(1) = \sigma(1)$ erinnere (4.3.4) und aus dem Beweis von Bemerkung (4.3.6) an die Darstellung des lokalen Diskretisierungsfehlers

$$\tau(t, y, h) = \frac{1}{h}\rho(1)y(t) + y'(t)(\rho'(1) - \sigma(1)) + O(h) = c_1 y'(t) + O(h).$$

Wir haben $c_1 = 0$ zu zeigen. Dazu halten wir fest: nach Satz 4.7.21 erfüllt ρ die Bedingung (WB), d.h. $|z| = 1$ ist nur einfache Nullstelle, woraus $\rho'(1) \neq 0$ folgt. Daher ist

$$R := \frac{\sigma(1)}{\rho'(1)}$$

wohldefiniert. Betrachte nun die spezielle Anfangswertaufgabe $y' = 1$ mit $y(0) = 0$. Lösung ist $y(t) = t$. Startwerte seien $y^j = jhR$ für $j = 0, \ldots, k-1$. Da $y(t) = t$ und insbesondere $y(t_j) = t_j = jh$ folgt für die Startwerte

$$y^j = jhR = jh + jh(R-1) = y(t_j) + \varepsilon^j$$

mit $\varepsilon^j := jh(R-1) \to 0$ für $h \to 0$ und alle $j = 0, \ldots, k-1$. Dazu liefert das Mehrschrittverfahren Werte

$$y^{j+k} + a_{k-1}y^{j+k-1} + \ldots + a_0 y^j = h(b_0 + \ldots + b_k) = h\sigma(1).$$

Angenommen, die letzten k Werte haben die Form wie die Startwerte, also

$$y^{j+i} = (j+i)hR, \quad j = 0, \ldots, i,\ i < k.$$

Wir wollen zeigen, dass y^{j+k} dann auch diese Form hat. Durch Aufsplitten und mit $\rho(1) = 0$ ist

$$\begin{aligned} h\sigma(1) &= y^{j+k} + Rh((j+k-1)a_{k-1} + \ldots + ja_0) \\ &= y^{j+k} + Rhj(1 + a_{k-1} + \ldots + a_0) - Rhj + Rh(k + (k-1)a_{k-1} + \cdots + a_1) - kRh \\ &= y^{j+k} + Rh\rho'(1) - (j+k)Rh, \end{aligned}$$

was äquivalent zu

$$y^{j+k} - (j+k)Rh = h(\sigma(1) - R\rho'(1)) = 0$$

ist. Also liefert das Verfahren als Lösung

$$y^j = jhR, \quad j = 0, 1, 2, \ldots$$

Aus der Konvergenz hat man nun $t = y(t) = \lim_{j\to\infty,\ jh=t}(jhR) = tR$, woraus $R = 1$ folgt. Also ist $\rho'(1) - \sigma(1) = 0$, womit sich die Behauptung mit Bemerkung 4.3.6 ergibt. ∎

Korollar 4.7.31. Die auf numerischer Integration beruhenden Verfahren vom Typ (r, ℓ) aus (4.5.8) bis (4.5.11) sind konvergent mit Konsistenzordnung.

Beweis: Diese Mehrschrittverfahren sind alle von der Form

$$y^{j+k} - y^{j+r-\ell} = h \ldots$$

mit $r - \ell < k$. Daraus folgt

$$\rho(z) = z^k - z^{r-\ell},$$

d.h. $a_k = 1$, $a_{r-\ell} = -1$ und $a_i = 0$ sonst, und daher

$$\rho(z) = z^{(r-\ell)}(z^{k-r+\ell} - 1), \quad r - \ell < k.$$

Also ist 0 $(r-\ell)$–fache Nullstelle und die $(k-r+\ell)$–ten Einheitswurzeln sind Nullstellen. Alle betrachteten Fälle sind so, dass $r - \ell = k - 1$ (Adams-Verfahren) oder $r - \ell = k - 2$ (Nyström, Milne–Simpson) gilt, also $\rho(z) = z^{k-1}(z-1)$ oder $\rho(z) = z^{k-2}(z^2-1)$. Also sind alle Nullstellen vom Betrag Eins einfach. Damit sind die Mehrschrittverfahren $\emptyset$–stabil, d.h. (WB) ist erfüllt und Konvergenz liegt somit nach Satz 4.7.21 vor. ∎

Satz 4.7.32. Die auf numerischer Differentiation beruhenden Verfahren aus Kapitel 4.4 erfüllen genau dann die Stabilitätsbedingung, wenn für $r = 0$ (implizite Verfahren) $k \leq 6$ bzw. für $r = 1$ (explizite Verfahren) $k \leq 2$ ist.

Zum Beweis: Zunächst kann man rekursiv eine explizite Form des charakteristischen Polynoms $\rho(z)$ herleiten,

$$\begin{aligned} \nabla z^k &= z^k - z^{k-1} = z^{k-1}(z-1) \\ \nabla^2 z^k &= \nabla(\nabla z^k) = \nabla(z^k - z^{k-1}) \\ &= z^k - z^{k-1} - (z^{k-1} - z^{k-2}) = z^k - 2z^{k-1} + z^{k-2} \\ &= z^{k-2}(z-1)^2. \end{aligned}$$

Über Induktion erhält man allgemein

$$\begin{aligned} \nabla^i z^k &= \nabla^{i-1}(\nabla z^k) \\ &= \nabla^{i-1}(z^k - z^{k-1}) = \nabla^{i-1} z^k - \nabla^{i-1} z^{k-1} \\ &= z^{k-(i-1)}(z-1)^{i-1} - z^{k-1-(i-1)}(z-1)^{i-1} \\ &= z^{k-i}(z-1)(z-1)^{i-1} = z^{k-i}(z-1)^i. \end{aligned}$$

Mit (4.4.7) hat damit das charakteristische Polynom des BDF–Verfahrens die Form

$$\rho(z) = \sum_{i=1}^{k} \frac{1}{i} z^{k-i}(z-1)^i. \tag{4.7.33}$$

Zur weiteren Untersuchung der Nullstellen von ρ mit funktionstheoretischen Methoden siehe [HNW], Chapter III, Theorem 3.4. ∎

In Korollar 4.3.15 wurde gezeigt, dass die maximal erreichbare Ordnung von linearen k–Schrittverfahren $p = 2k$ beträgt. Dies wird allerdings durch die folgende Aussage eingeschränkt.

Satz 4.7.34 (Erste Dahlquist–Schranke)**.** Für die Ordnung p eines *stabilen* linearen k–Schrittverfahrens gilt

$$p \leq \begin{cases} k+2, & \text{falls } k \text{ gerade} \\ k+1, & \text{falls } k \text{ ungerade} \\ k, & \text{falls } \frac{b_k}{a_k} \leq 0, \text{ also insbesondere für explizite Verfahren} \end{cases}$$

Beweis: [HNW], Chapter III, Theorem 3.5. ∎

Ein Beispiel dafür, dass die Grenzen scharf sind, ist das Verfahren von Milne–Simpson: für $k = 2$ gilt $p = 4$ mit $\rho(z) = z^2 - 1$, vergl. Satz (4.5.12) (iv).

4.7.4 Stabilitätsbegriffe für Schrittweiten $h > 0$ und Stabilitätsbereiche

Der bisherige Begriff der $\emptyset$–Stabilität (d.h. (WB) ist für ρ erfüllt) berücksichtigt nicht, dass im praktischen Verfahren die Schrittweite h wegen des Einflusses von Rundungsfehlern nicht beliebig klein gewählt werden kann. Auch ist dieser Stabilitätsbegriff völlig unabhängig vom zugrunde liegenden Problem.

Für praktische Probleme kommt daher auch die rechte Seite Φ ins Spiel. Im Folgenden wird das Verhalten $\emptyset$–stabiler Verfahren für *feste* Schrittweiten $h > 0$ untersucht. Dabei kommen Eigenschaften des spezifischen Problems mit ins Spiel.

Betrachte die *lineare* Form des allgemeinen Mehrschrittverfahrens (4.7.14) mit Störungen für eine Anfansgwertaufgabe in $n = 1$. Wie im Beweis von Satz 4.7.21 gilt für den Fehler

$$e_j := y^j - y(t_j)$$

und

$$\begin{aligned} \sum_{\ell=0}^{k} a_\ell e_{j+\ell} &= h \sum_{\ell=0}^{k} b_\ell (f_{j+\ell} - f(t_{j+\ell}, y(t_{j+\ell}))) + h \text{ rest}_{j+k} \\ &= h \sum_{\ell=0}^{k} b_\ell \left(\frac{\partial f}{\partial y}(t_{j+\ell}, \eta_{j+\ell}) \right) e_{j+\ell} + h \text{ rest}_{j+k}. \end{aligned}$$

Im Folgenden sei vereinfachend $\frac{\partial f}{\partial y} =: \lambda$, $h \text{ rest}_{j+k} =: K =$ konst unabhängig von j und k angenommen. Dann gilt

$$\sum_{\ell=0}^{k} (a_\ell - h\lambda b_\ell) e_{j+\ell} = K. \tag{4.7.35}$$

Wir untersuchen nun das Verhalten des Fehlers. Eine spezielle Lösung der inhomogenen Gleichung (4.7.35) ist

$$g = g_j = -\frac{K}{h\lambda \sum\limits_{\ell=0}^{k} b_\ell} = \text{const.}$$

Die allgemeine Lösung erhält man durch die Superposition $u^j + g_j$, wobei $\{u^j\}_j$ allgemeine Lösung der homogenen Differenzengleichung

$$\sum_{\ell=0}^{k} (a_\ell - h\lambda b_\ell) u^{j+\ell} = 0, \quad j = 0, 1, 2, \ldots \tag{4.7.36}$$

ist. Das Wachstumsverhalten von e_j wird also durch das von u^j bestimmt. Betrachte dazu das *Stabilitätspolynom*

$$\Pi(z, h\lambda) := \sum_{\ell=0}^{k} (a_\ell - h\lambda b_\ell) z^\ell = \rho(z) - h\lambda\sigma(z), \tag{4.7.37}$$

das sich direkt aus der Differenzengleichung (4.7.36) ableiten lässt. Beachte, dass $\lim\limits_{h\to 0} \Pi(z, h\lambda) = \rho(z)$. Wir fragen nun, welche Probleme im Hinblick auf Stabilität auftreten können, vergleiche Beispiel 4.7.1.

Beispiel 4.7.38. Wende die Mittelpunktsregel

$$y^{j+2} - y^j = 2hf_{j+1}$$

auf die Anfangswertaufgabe $y' = \lambda y$ mit festem $\lambda \in \mathbb{R}$ und $y(0) = 1$, dessen Lösung $y(t) = e^{\lambda t}$ ist, an. Hier ist

$$y^{j+2} - y^j = 2h\lambda y^{j+1}. \tag{4.7.39}$$

Die Startwerte seien die exakten Werte $y^0 = 1$ und $y^1 = e^{\lambda h}$. Gemäß des Ansatzes zur Lösung von (4.7.39) berechne die Nullstellen des Stabilitätspolynoms

$$\Pi(z, h\lambda) = z^2 - 2h\lambda z - 1 = \rho(z) - h\lambda\sigma(z).$$

Diese lauten

$$z_{1/2} = h\lambda \pm \sqrt{\lambda^2 h^2 + 1}.$$

Nach Lemma 4.7.7 lassen sich dann alle Lösungen von (4.7.39) als Linearkombination

$$y^j = c_1 (z_1)^j + c_2 (z_2)^j$$

schreiben. Eine Entwicklung von z_1 und z_2 nach den Potenzen von h ergibt

$$\begin{aligned}
z_1 &= 1 + h\lambda + \frac{1}{2}(h\lambda)^2 - \frac{1}{8}(h\lambda)^4 + O(h^6) \\
&= e^{h\lambda}(1 - \frac{1}{6}(h\lambda)^3) + O(h^5), \\
z_2 &= -1 + h\lambda - \frac{1}{2}(h\lambda)^2 + O(h^4) \\
&= -e^{-h\lambda}(1 + \frac{1}{6}(h\lambda)^3) + O(h^4).
\end{aligned}$$

Einsetzen der Anfangsbedingung ergibt

$$\begin{aligned}
c_1 &= \frac{z_2 - e^{h\lambda}}{z_2 - z_1} = 1 + \frac{(h\lambda)^3}{12} + O(h^4) \\
c_2 &= \frac{e^{h\lambda} - z_1}{z_2 - z_1} = \frac{(h\lambda)^3}{12} + O(h^4).
\end{aligned}$$

Für festes $t = jh$, d.h. $y_h(t_j) = y^j$, folgt daher

$$y_h(t) = e^{\lambda t}\left(1 - \frac{\lambda^3}{6}h^2 + \frac{(\lambda h)^3}{12}\right) + (-1)^{t/h}e^{-\lambda t}\left(\frac{\lambda^3 h^3}{12}\right) + O(h^4).$$

Für $\lambda < 0$ fällt der erste Summand entsprechend der Lösung ab, wird aber für $h > \delta$ letztlich vom zweiten Summanden dominiert. Der zweite Term ist für große t stark oszillierend.

Obwohl also die Stabilitätsbedingung (WB) erfüllt ist, ist das Verfahren für die Aufgaben nur bedingt geeignet oder sogar *ungeeignet*, für die $\lambda < 0$ und $|\lambda| \gg 1$ ist. (Für das allgemeine Problem $y' = f(t, y)$ wäre das der Fall, wenn f großen Schwankungen unterliegt; $\lambda \approx \frac{\partial f}{\partial y}$.) Der Grund für dieses Stabilitätsproblem besteht in diesem Beispiel darin, dass das charakteristische Polynom $\rho(z) = z^2 - 1$ *zwei* Wurzeln vom Betrag Eins hat, nämlich ± 1. Da für die Wurzeln $z_i(h\lambda)$ des Stabilitätspolynoms

$$\begin{aligned} z_1(h\lambda) &= 1 + h\lambda + O((h\lambda)^2) \\ z_2(h\lambda) &= -1 + h\lambda + O((h\lambda)^2), \end{aligned} \tag{4.7.40}$$

gilt, ist für $\lambda < 0$ zwar $|z_1(h\lambda)| < 1$, aber $|z_2(h\lambda)| > 1$. ∎

Dies motiviert folgende

Definition: Alle Wurzeln von $\rho(z)$, die von $z = 1$ verschieden sind, heißen *parasitäre* Wurzeln. Die Wurzel $z = 1$ heißt *Hauptwurzel*. Ein lineares Mehrschrittverfahren heißt *stark stabil*, falls $\rho(1) = 0$ (Konsistenz) gilt, und alle anderen Nullstellen $|z| < 1$ erfüllen (d.h. $z = -1$ ist *keine* Nullstelle). Ein lineares MSV heißt *schwach stabil*, falls (WB) gilt, aber mehrere Wurzeln den Betrag Eins haben (z.B. Mittelpunktsregel, Milne–Simpson, denn die Verwendung *gerader* Differenzen impliziert die Existenz zweier Wurzeln vom Betrag Eins). ∎

Allgemein gilt

Bemerkung 4.7.41. Gegeben sei die Anfangswertaufgabe $y' = \lambda y$ mit $y(0) = 1$, die die Lösung $y = e^{\lambda t}$ besitzt. Das lineare Mehrschrittverfahren habe die Ordnung p und sei konvergent. Dann gilt für die Hauptwurzel von $\Pi(z, h, \lambda)$

$$z_1(h\lambda) = e^{h\lambda} + O(h^{p+1}).$$

Beweis: Aus der Konvergenz der Ordnung p folgt Konsistenz mit Ordnung p. Einsetzen der exakten Lösung in Definition (4.2.1) ergibt

$$h\tau(t, y, h) = \sum_{\ell=0}^{k}(a_\ell - h\lambda b_\ell)e^{\lambda(t+\ell h)} = O(h^{p+1}),$$

d.h.

$$e^{\lambda t}\sum_{\ell=0}^{k}(a_\ell - h\lambda b_\ell)e^{\lambda \ell h} = O(h^{p+1}),$$

also

$$\Pi(e^{h\lambda}, h\lambda) = O(h^{p+1}).$$

Nun gilt andererseits

$$\Pi(e^{h\lambda}, h\lambda) = (a_k - h\lambda b_k)(e^{h\lambda} - z_1(h\lambda)) \cdots (e^{h\lambda} - z_k(h\lambda)).$$

Aus der Konvergenz folgt die $\emptyset$–Stabilität des Mehrschrittverfahrens. Daher ist $z = 1$ einfache Nullstelle von $\rho(z)$, womit

$$e^{h\lambda} - z_i(h\lambda) \not\to 0 \text{ für } h \to 0, \quad i > 1,$$

gilt, es also keine weitere Nullstelle $z = 1$ gibt. Wegen $\Pi(e^{h\lambda}, h\lambda) = O(h^{p+1})$ gilt aber insgesamt

$$e^{h\lambda} - z_1(h\lambda) = O(h^{p+1}).$$

■

Je nach dem asymptotischen Verhalten der Lösung y der Anfangswertaufgabe $y' = \lambda y$ ergeben sich unterschiedliche *Anforderungen* an das numerische Verfahren.

Bemerkung: Sei $n = 1$. Die Lösung y von $y' = \lambda y$ mit $y(0) = 1$ erfüllt

$$\lim_{t\to\infty} y(t) = \begin{cases} = 0, & \mathrm{Re}(\lambda) < 0 \ \ (\text{bzw. } \mathrm{Re}\frac{\partial f}{\partial y} < 0) \\ < \infty, & \mathrm{Re}(\lambda) \leq 0 \\ = \infty, & \mathrm{Re}(\lambda) > 0 \end{cases}.$$

Definition 4.7.42. Ein Mehrschrittverfahren heißt *absolut stabil* für $h\lambda$, falls

$$|z_i(h\lambda)| \leq 1, \quad \text{für jedes } i = 1, \ldots, m \quad \text{und } |z_i(h\lambda)| < 1 \text{ für mehrfache Nullstellen}$$

ist, andernfalls *absolut instabil*. Die Menge

$$D_\lambda := \{h\lambda : \ |z_i(h\lambda)| \leq 1 \ \text{ für } i = 1, \ldots, m \ \text{ und } |z_i(h\lambda)| < 1 \text{ für } z_i(h\lambda) \text{ mehrfache Nullstelle}\}$$

heißt *Bereich absoluter Stabilität*. ■

Bemerkung 4.7.43. Wegen Bemerkung 4.7.41 ist *jedes* konsistente, asymptotisch stabile (d.h. $\emptyset$–stabile) Mehrschrittverfahren für *kleine positive* $h\lambda$ absolut *instabil*. ■

Es wird sich zeigen, dass das Konzept der absoluten Stabilität wichtig für Probleme mit Re $\lambda < 0$ („steife“ Probleme) ist. Man kann zeigen, dass optimale Mehrschrittverfahren (d.h. konsistente und $\emptyset$–stabile Verfahren mit Ordnung $p = k + 2$ (4.7.34)) *keine* Bereiche absoluter Stabilität haben. Man erwartet, dass stark stabile Verfahren ($|z| < 1$ für alle parasitären Nullstellen) stets *nichtleere* Bereiche absoluter Stabilität haben (für $\mathrm{Re}(\lambda) < 0$). Falls bei einem Verfahren alle parasitären Wurzeln im Ursprung liegen, erwartet man beträchtliche Bereiche absoluter Stabilität, so z.B. beim Adams–Verfahren mit

$$\rho(z) = z^k - z^{k-1} = z^{k-1}(z - 1).$$

Als nächstes folgen ein paar Beispiele zu *Bereichen absoluter Stabilität*. Sei im Folgenden stets Re $\lambda < 0$.

Beispiel 4.7.44. Gegeben sei das implizite Euler–Verfahren

$$y^{j+1} = y^j + hf(t_{j+1}, y^{j+1}).$$

Das zugehörige Stabilitätspolynom lautet

$$\Pi(z, h\lambda) = z - 1 - (h\lambda)z,$$

d.h. $\Pi(z, h\lambda) = 0$ genau dann, wenn $z(1 - h\lambda) = 1$ oder äquivalent dazu $z = \frac{1}{1-h\lambda}$ ist, also hat man

$$D_\lambda = \{h\lambda : \ \frac{1}{|1 - h\lambda|} < 1\} = \{h\lambda : \ |h\lambda - 1| \geq 1\} \supseteq \{h\lambda : \ \mathrm{Re}(\lambda) \leq 0\} = \mathbb{C}_-,$$

siehe Figur 10. Unabhängig von der Größe von λ ist also h beliebig wählbar, man bleibt immer im Bereich der absoluten Stabilität.

Im Unterschied dazu betrachte das explizite Euler–Verfahren

$$y^{j+1} = y^j + hf_j.$$

Hier ist

$$\Pi(z, h\lambda) = z - 1 - h\lambda,$$

mit Nullstelle

$$z = 1 + h\lambda,$$

also

$$D_\lambda = \{h\lambda : \ |1 + h\lambda| \leq 1\}.$$

Für Re $\lambda > 0$ ist das Verfahren absolut instabil wegen Bemerkung (4.7.43), für Re $\lambda \leq 0$ ist D_λ die Kreisscheibe um -1 mit Radius 1.

Die schraffierten Flächen folgender Grafiken stellen die Stabilitätsbereiche für das explizite und das implizite Euler–Verfahren dar.

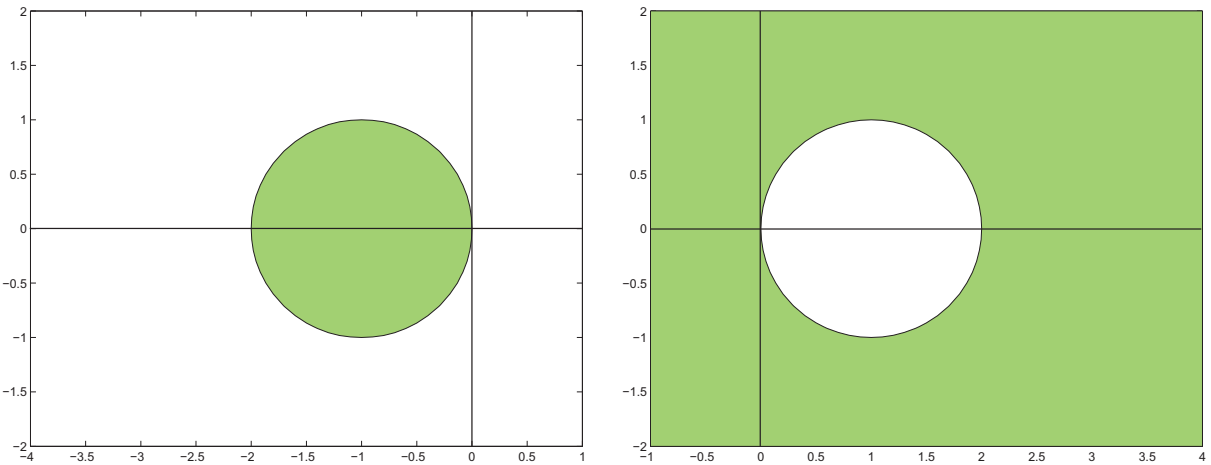


Abbildung 10: Schraffiert sind die Bereiche der absoluten Stabilität für das explizite (links) und das implizite Euler–Verfahren (rechts)

Hierbei sieht man schon, dass *implizite* Verfahren erheblich größere Bereiche absoluter Stabilität als *explizite* Verfahren besitzen. Setzt man im Verfahren von Adams–Moulton $r = k = 1$, so erhält man die Trapezregel

$$y^{j+1} - y^j = \frac{h}{2}(f_j + f_{j+1})$$

mit dem Stabilitätspolynom

$$\Pi(z, h\lambda) = z - 1 - \frac{h\lambda}{2}(z + 1) = z\left(1 - \frac{h\lambda}{2}\right) - \left(1 + \frac{h\lambda}{2}\right).$$

Die Nullstelle lautet

$$z = \frac{1 + h\lambda/2}{1 - h\lambda/2}.$$

Also ist

$$D_\lambda = \{h\lambda : \left|\frac{1+h\lambda/2}{1-h\lambda/2}\right| \leq 1\} \supseteq \{h\lambda : \ \mathrm{Re}(\lambda) < 0\}.$$

■

Die Bereiche absoluter Stabilität sind für $k = 1, \ldots, 6$ für die expliziten und impliziten ADAMS–Verfahren in Figur 11 und 12 aufgetragen.

Definition: Ist $D_\lambda \supseteq \{h\lambda : \ \mathrm{Re} \leq 0\}$, so heißt das Mehrschrittverfahren *A–stabil*. Also sind das implizite EULER–Verfahren und die Trapezregel A–stabil.

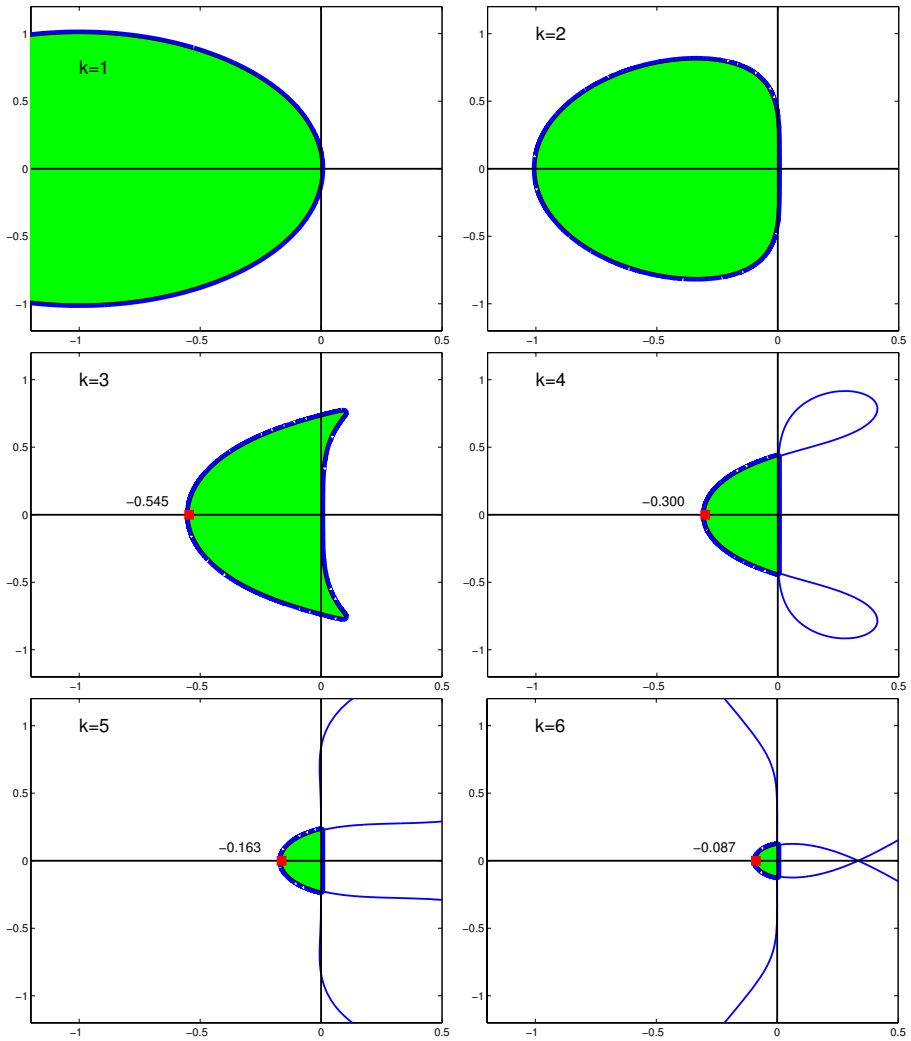


Abbildung 11: Bereiche absoluter Stabilität für die expliziten ADAMS–Methoden

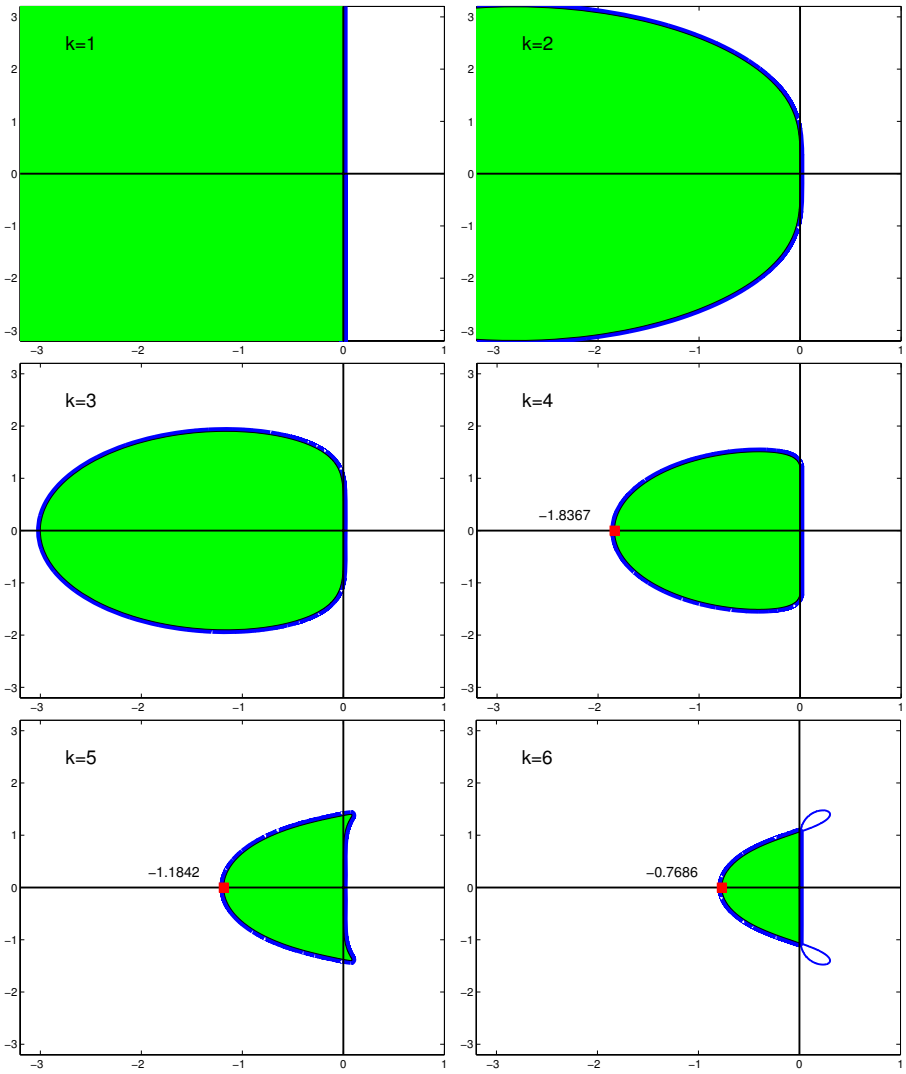


Abbildung 12: Bereiche absoluter Stabilität für die impliziten ADAMS–Methoden

Relative Stabilität

Sei wieder $y' = \lambda y$ mit $y(0) = 1$ gegeben. Im Fall $\mathrm{Re}\lambda > 0$ wächst die Lösung. Bemerkung 4.7.41 besagt, dass

$$(z_1(h\lambda))^j = (e^{h\lambda})^j + O(h^{p+1}) = e^{\lambda(t-t_0)} + O(h^{p+1})$$

für $t - t_0 = jh$ ist, d.h. der durch z_i erzeugte Fehleranteil wächst höchstens wie die Lösung. Ein Mehrschrittverfahren heißt daher *relativ stabil* für $h\lambda$, falls

$$|z_i(h\lambda)| < |z_1(h\lambda)| \quad \text{für alle restlichen } i = 2, \ldots, m$$

gilt. Entsprechend sind Bereiche relativer Stabilität definiert.

Beispiel: Wir betrachten die Mittelpunktsregel (Sonderfall von NYSTRÖM)

$$y^{j+1} = y^{j-1} + 2hf_j.$$

Aus der $\emptyset$–Stabilität und der Konsistenz folgt die absolute Instabilität für $\mathrm{Re}(\lambda) > 0$ wegen Bemerkung 4.7.41. Es ist $\Pi(z, h\lambda) = z^2 - 1 + 2h\lambda = 0$ genau dann, wenn

$$z_{1/2} = h\lambda \pm \sqrt{(h\lambda)^2 + 1}.$$

Es gilt $|z_2(h\lambda)| = |h\lambda - \sqrt{(h\lambda)^2 + 1}| < |z_1(h\lambda)|$, also ist das Verfahren relativ stabil für alle $h\lambda$ mit $\mathrm{Re}(h\lambda) > 0$.

Abschließend halten wir fest, dass die verwendeten Stabilitätskonzepte von den Aufgaben abhängen. Im Fall $\mathrm{Re}(h\lambda) > 0$, sind $\emptyset$–stabile, konsistente Verfahren absolut instabil für alle $h\lambda$ wegen Bemerkung 4.7.41. Relevant in diesem Fall ist dann das Konzept der relativen Stabilität.

Das Konzept absoluter Stabilität ist wiederum interessant für die im nächsten Kapitel behandelten steifen Probleme, bei denen $\mathrm{Re}(h\lambda) < 0$ ist.

5 Steife Probleme

Beispiele für das Auftreten steifer Probleme sind Kontrollprozesse, chemische Reaktionen mit *stark unterschiedlichen* Abklingraten oder Diffusionsprozesse (Wärmeleitungsgleichung $u_t - u_{xx} = f$). Dabei ist die Steifheit oft versteckt. Für ein erstes **Beispiel** sei die Anfangswertaufgabe

$$y' = -50(y - \cos t), \quad y(0) = 0$$

gegeben. Die exakte Lösung ist

$$y(t) = \frac{2500}{2501}\cos t + \frac{50}{2501}\sin t - \frac{2500}{2501}e^{-50t}.$$

Auffällig ist der explosionsartige Anstieg nahe $t = 0$, der für steife Probleme typisch ist. Wir verwenden einmal das implizite, einmal das explizite EULER–Verfahren. In letzterem ist wegen $\lambda = -50$ der Bereich der absoluten Stabilität durch $D_\lambda = \{h : |1 - 50h| \leq 1\}$ gegeben und die Schrittweite h somit durch $0 \leq h \leq \frac{2}{50} = 0.04$ eingeschränkt. Die Abbildungen weiter unten zeigen jeweils die exakte Lösung auf dem Intervall $[0, 1.5]$. In Abbildung 5 ist die Lösung mit dem impliziten EULER–Verfahren berechnet. Hier liefert selbst eine verhältnismäßig große Schrittweite von $h = 0.5$ eine passable Annäherung. In Abbildung 5 sind zusätzlich zur exakten Lösung die diskretisierten Lösungen des expliziten EULER–Verfahrens zu den Schrittweiten $h = 1.974/50 = 0.03948$ (38 Schritte) und $h = 1.875/50 = 0.0375$ (40 Schritte) aufgetragen. Ist die Schrittweite schon fast am Rand des Stabilitätsgebiets, so kommt es bereits zu starken Oszillationen, was das schlechtere Abschneiden für Schrittweite $h = 0.03948$ erklärt. Außer, dass sich das Verfahren auf den ersten Blick wegen der großen Oszillationen als ungeeignet zur Lösung dieses Problems herausstellt, fällt auf, dass eine minimale Änderung der Schrittweite eine stark veränderte approximative Lösung hervorbringt. Im Folgenden wird sich zeigen, dass implizite Verfahren bei steifen Problemen den expliziten stets vorzuziehen sind.

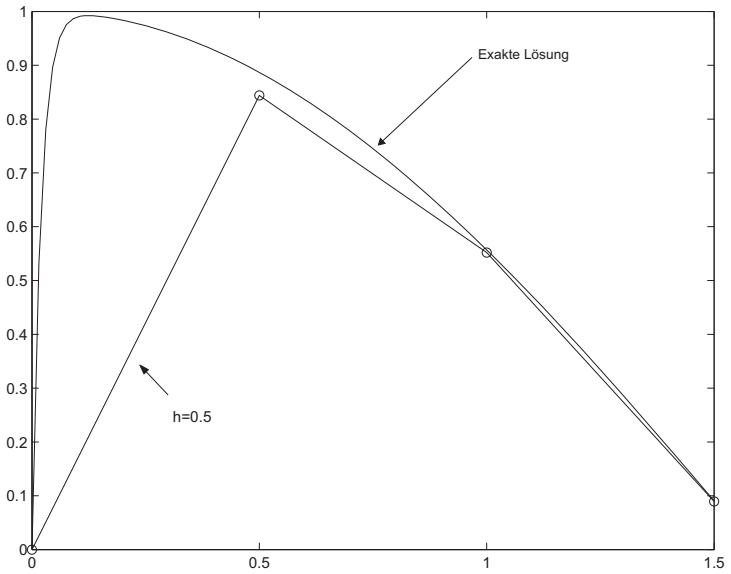


Abbildung 13: Implizites EULER–Verfahren für $y' = -50(y - \cos t), \quad y(0) = 0$

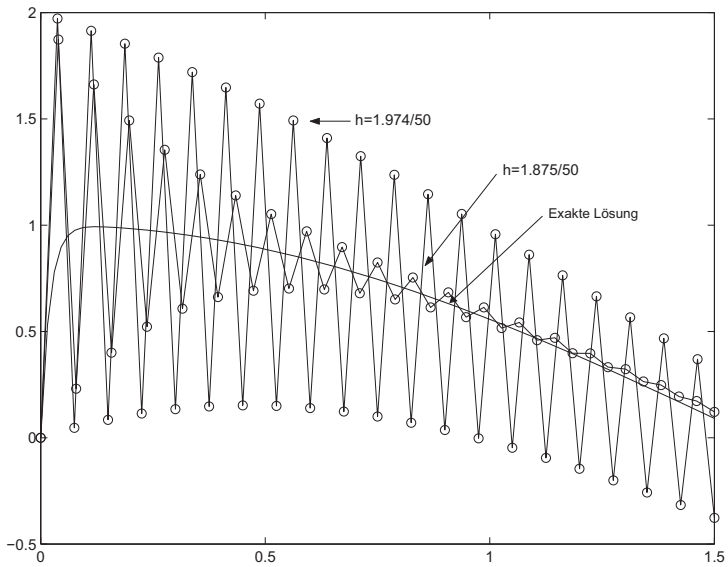


Abbildung 14: Explizites EULER–Verfahren für $y' = -50(y - \cos t), \quad y(0) = 0$

Beispiel 5.1. Gegeben sei das Differentialgleichungssystem

$$\begin{aligned} y_1' &= -100y_1 \\ y_2' &= -2y_2 - y_1 \end{aligned}$$

mit den Anfangsdaten $y_1(0) = y_2(0) = 1$. Die Lösung der ersten Gleichung $y_1(t) = e^{-100t}$ fällt sehr schnell ab, während sich die Lösung von $y_2'(t) = -2y_2 - e^{-100t}$ sehr viel langsamer ändert. Bis $t = 0.01$ klingt y_1 um den Faktor e^{-1} ab, bei $t = 0.1$ ist er um $e^{-10} \approx 10^{-4}$ reduziert und spielt für größere t keine Rolle mehr. Dennoch beeinflusst dieser Term die Rechnung erheblich! Wendet man nämlich das explizite EULER–Verfahren auf

$$y_1' = -100y_1, \quad y_1(0) = 1,$$

an, erhält man die Rekursion

$$y_1^{j+1} = y_1^j - 100hy_1^j = (1 - 100h)y_1^j.$$

Für $h = \frac{1}{100}$ folgt mit $y_1^0 = 1$

$$y_1^{j+1} = \frac{1}{2}y_1^j = \ldots = 2^{-j-1}.$$

Für $j = 20$, also bei $t_{20} = 0.1$, ergibt sich

$$|y_1^{21}| = 2^{-21} < 10^{-6},$$

wie bei der exakten Lösung spielt y_1^j nach $t = 0.1$ keine Rolle mehr. Rechnet man nun mit $h = \frac{1}{2}$ weiter, um y_2 angemessen zu integrieren, erhält man aber $1 - 100 \cdot \frac{1}{2} = -49$, also

$$y_1^{j+1} = -49y_1^j,$$

was zu einem explosionsartigen Anwachsen von y_1^j im Lauf der Rechnungen führt. ■

Definition 5.2. Ein System $y' = Ay + B(t)$ bzw. $y' = f(t, y)$ heißt *steif*, wenn

i) $\mathrm{Re}(\lambda_i) < 0$ für alle Eigenwerte λ_i von $A \in \mathbb{R}^{n\times n}$ bzw. $\frac{\partial f}{\partial y}$;

ii) $\max_i |\mathrm{Re}(\lambda_i)| \gg \min_i |\mathrm{Re}(\lambda_i)|$.

Der Quotient

$$q := \frac{\max\limits_i |\mathrm{Re}(\lambda_i)|}{\min\limits_i |\mathrm{Re}(\lambda_i)|}$$

heißt *Steifigkeitsquotient* (stiffness quotient). Die relativ stark abklingende Komponente heißt *steife Komponente.* ■

In obigen Beispiel beträgt mit $\lambda_1 = -1$ und $\lambda_2 = -100$ der Steifigkeitsquotient $q = 100$. Es gibt aber auch Beispiele für $q = 10^{16}$! Der Grund für die Schwierigkeit in obigem Beispiel 5.1 ist, dass für $h = \frac{1}{2}$ der Term $h\lambda$ für $\lambda = -1, -100$ nicht im Bereich der absoluten Stabilität des expliziten Euler–Verfahrens liegt. Dies ist nur für $h < \frac{2}{100}$ der Fall. Aber zu kleine Schrittweiten sind für y_2 wegen der Anhäufung von Rundungsfehlern nicht angemessen. Die generelle Schwierigkeit liegt also darin, $h\lambda$ für $\lambda = -100$ in den Bereich der absoluten Stabilität zu bringen. Beachte, dass diese Restriktion an h von *dem* Eigenwert herrührt, der für die praktische Lösung irrelevant ist! Obige Schwierigkeit, $h\lambda$ in den Bereich der absoluten Stabilität zu bringen, fällt weg, wenn der Bereich der absoluten Stabilität die gesamte Halbebene $\mathbb{C}_-$ umfasst, also wenn das Verfahren A–stabil ist. Leider gelten für A-stabile Verfahren die folgenden Einschränkungen.

Satz 5.3 (Dahlquist)**.**

i) Ein explizites Mehrschrittverfahren ist niemals A–stabil.

ii) Die Ordnung eines impliziten, A–stabilen, *linearen* Mehrschrittverfahrens ist höchstens zwei.

iii) Das A–stabile, lineare, implizite Mehrschrittverfahren mit der kleinsten Fehlerkonstante nach Definition (4.3.13) ist die Trapezregel

$$y^{j+1} - y^j = \frac{h}{2}(f_{j+1} + f_j).$$

■

Eine Abschwächung von A–Stabilität ist die **A**(α)–**Stabilität**. Hiermit ist gemeint, dass der Bereich der absoluten Stabilität einen Kegel in $\mathbb{C}_-$ mit der Spitze im Ursprung und Winkel α zwischen der reellen Achse und dem oberen Kegelschenkel umfasst, $0° \le \alpha \le 90°$. Für $\alpha = 90°$ ist $A(90°)$–Stabilität gerade wieder die A–Stabilität. Bei A$(0°)$–Stabilität ist der Bereich der absoluten Stabilität nur die reelle Halbachse, womit noch Anfangswertaufgaben $y' = \lambda y$ mit negativem reellen λ stabil gelöst werden können.

Satz 5.4 (Einschrittverfahren)**.**

(i) Explizite Runge–Kutta–Verfahren haben *kleine* Bereiche absoluter Stabilität.

(ii) Implizite Runge–Kutta–Verfahren sind A–stabil. ■

Dazu betrachten wir den Fall $m = 1$, also das einstufige Runge–Kutta–Verfahren (3.5.1) in der Gauss–Form (mit $p = 2$), also

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

für die Anfangswertaufgabe $y' = \lambda y$ und $y(0) = 1$. Das Verfahren lautet

$$y^{j+1} = y^j + hk_1(t_j, y^j),$$

mit

$$k_1(t_j, y^j) = f\left(t_j + \frac{1}{2}, y^j + \frac{h}{2}k_1(t_j, y^j)\right) = \lambda\left(y^j + \frac{h}{2}k_1(t_j, y^j)\right),$$

was äquivalent zu

$$\left(1 - \frac{h\lambda}{2}\right)k_1 = \lambda y^j$$

bzw.

$$k_1 = \frac{1}{1 - \frac{h\lambda}{2}}\lambda y^j$$

ist. Daher lautet das Verfahren

$$y^{j+1} = y^j + \frac{h\lambda}{1 - \frac{h\lambda}{2}}y^j = \frac{1 + h\lambda/2}{1 - h\lambda/2}\,y^j.$$

Das Stabilitätspolynom lässt sich analog wie bei den linearen MSV in (4.7.37) hier als

$$\Pi(z, h\lambda) = z - \frac{1 + h\lambda/2}{1 - h\lambda/2}$$

angeben und hat die Nullstelle

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

Das Verfahren ist absolut stabil, falls $|z(h\lambda)| \leq 1$, also

$$|z(h\lambda)| = \left|\frac{1 + h\lambda/2}{1 - h\lambda/2}\right| \leq 1$$

gilt. Für $\mathrm{Re}(\lambda) \leq 0$ entspricht dies der gesamten linken Halbebene. Also ist das implizite einstufige Runge–Kutta–Verfahren A–stabil. Die Rechnung für $m > 1$ läuft analog. Für $m = 2$ gilt etwa für die Gauss–Form

$$y^{j+1} = \frac{1 + \frac{h\lambda}{2} + \frac{(h\lambda)^2}{12}}{1 - \frac{h\lambda}{2} + \frac{(h\lambda)^2}{12}}\,y^j,$$

und $|z(h\lambda)| < 1$ gilt für $\mathrm{Re}(\lambda) < 0$ [Gr], Kapitel 3.5.

Bemerkung: Bei impliziten Runge–Kutta–Verfahren der Stufe m in der Gauss–Form hat die Nullstelle des Stabilitätspolynoms die Form

$$z(h\lambda) = \frac{1 + \frac{h\lambda}{2} + \frac{(h\lambda)^2}{12} + \ldots + c_m(h\lambda)^m}{1 - \frac{h\lambda}{2} + \frac{(h\lambda)^2}{12} - \ldots - c_m(h\lambda)^m},$$

so dass sich auch hier $|z(h\lambda)| < 1$ für $\mathrm{Re}(\lambda) < 0$ zeigen lässt, das Verfahren also A–stabil ist. Die Nullstellen sind rationale Funktionen in $h\lambda$. Deren besondere Form lässt sich auch als Padé–Approximation (Approximation durch rationale Funktionen) von $e^{h\lambda}$ interpretieren [Gr], Kapitel 3.5.

Bei den *expliziten* RUNGE–KUTTA–Verfahren der Ordnung p hat demgegenüber die Nullstelle

$$z(h\lambda) = 1 + h\lambda + \frac{(h\lambda)^2}{2} + \ldots + \frac{(h\lambda)^p}{p!}$$

die Form des TAYLOR–Polynoms vom Grad p von $e^{h\lambda}$. Die Stabilitätsbereiche dieser Verfahren für die Ordnungen $p = 1, \ldots, 4$ zeigt die nebenstehende Grafik. Die Bereiche der absoluten Stabilität für die expliziten RUNGE–KUTTA–Verfahren der Ordnung $p = 1, \ldots, 4$ liegen innerhalb der geschlossenen Linien. Beachte, dass aufgrund der unterschiedlichen Skalierung der Achsen es sich im Fall $p = 1$ um die Scheibe um -1 mit Radius 1 handelt.

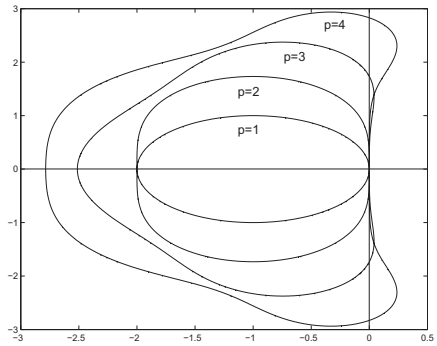


Abbildung 15: Stabilitätsbereiche der expliziten RUNGE–KUTTA–Verfahren $p = 1, \ldots, 4$

Zusammenfassend lässt sich sagen, dass die impliziten RUNGE–KUTTA–Verfahren zur angemessenen Lösung steifer Probleme gut geeignet sind.

Steife Probleme und lineare Mehrschrittverfahren

Bei den impliziten RUNGE–KUTTA–Einschrittverfahren hatte die Nullstelle des Stabilitätspolynoms

$$\Pi(z, h\lambda) = \rho(z) - h\lambda\sigma(z)$$

die Form $z = \frac{1+h\lambda+\ldots}{1-h\lambda+\ldots}$, was für $\mathrm{Re}(\lambda) < 0$ A–stabile Verfahren lieferte.

Eine Übertragung auf k–Schrittverfahren liefert die Forderung

$$z_i(h\lambda) \to 0, \quad \text{für } h\lambda \to -\infty$$

für alle Nullstellen. Aus $\Pi(z, h\lambda) = \sum_{\ell=0}^{k}(a_\ell - h\lambda b_\ell)z^\ell = 0$ folgt (mit $a_k - h\lambda b_k \neq 0$)

$$z^k + \frac{(a_{k-1} - h\lambda b_{k-1})}{a_k - h\lambda b_k} z^{k-1} + \cdots + \frac{a_0 - h\lambda b_0}{a_k - h\lambda b_k} = 0$$

und obiger Forderung folgt $b_{k-1} = \ldots = b_0 = 0$. Damit ist das Verfahren von der Form

$$\sum_{\ell=0}^{k} a_\ell y^{j+\ell} = h b_k f_{j+k},$$

also ein implizites Verfahren vom Typ der BDF–Verfahren (4.4.7). Dieses sind A(α)–stabile Verfahren (4.4.7). Die Stabilitätsbereiche der jeweiligen Verfahren sind in Figur 16 dunkel eingefärbt. Die zugehörigen Stabilitätswinkel sind

k	1	2	3	4	5	6
α	90°	90°	86.03°	73.35°	51.84°	17.84°

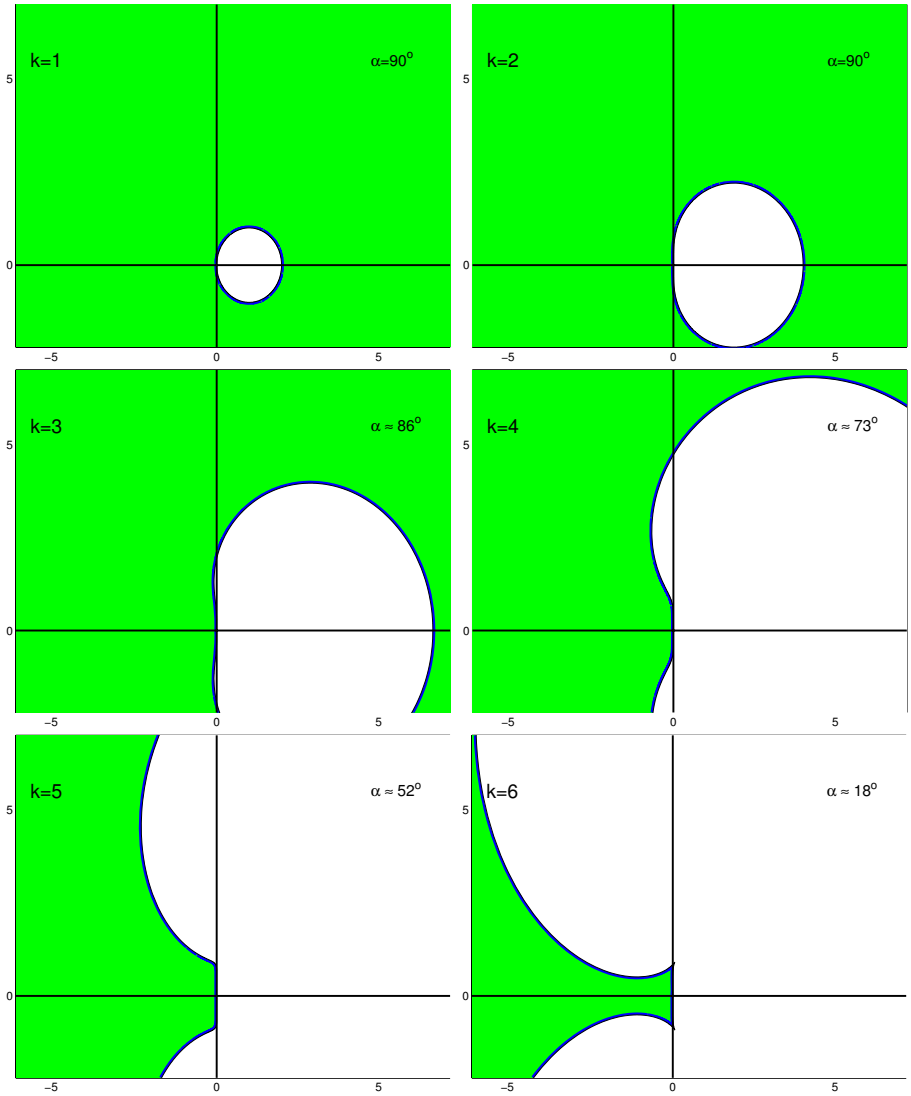


Abbildung 16: Bereiche der absoluten Stabilität für die impliziten BDF–Verfahren für $k = 1, \ldots, 6$

Pragmatische Definition [Curtis und Hirschfelder] 1952: (vgl. Definition 5.2) Ein System heißt *steif*, wenn gewisse implizite Methoden (speziell das BDF–Verfahren) erheblich bessere Ergebnisse als explizite Verfahren liefern.

Steifheit hängt nicht nur von den Eigenwerten ab. Die Verfahrensklasse definiert den Typ des kontinuierlichen Problems. Eine gewöhnliche Differentialgleichung ist nicht per se steif. Daher ist der Begriff „steife Differentialgleichung“ eigentlich unpassend. Strategien zur automatischen Entdeckung von Steifheit findet man in [Br].

6 Randwertprobleme

6.1 Einleitung

Klassische Beispiele für Randwertprobleme sind Schwingungs–, und Eigenwertprobleme. Gegeben sei das Modellproblem zweiter Ordnung

$$y'' = f(x, y, y') \quad \text{mit } y(a) = \alpha \text{ und } y(b) = \beta, \tag{6.1}$$

dessen Lösung $y = y(x)$ auf dem Intervall (a, b) zu finden ist.

Beispiel 6.1a. Betrachte $y'' = -y$ auf $(0, b)$ mit den Randbedingungen $y(0) = \alpha$ und $y(b) = \beta$. Die allgemeine Lösung lautet

$$y(x) = c_1 \sin x + c_2 \cos x.$$

Einsetzen der Randbedingung $y(0) = \alpha$ ergibt $c_2 = \alpha$ und aus $y(b) = \beta$ folgt $c_1 \sin b + c_2 \cos b = \beta$, was äquivalent zu

$$\begin{pmatrix} 0 & 1 \\ \sin b & \cos b \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

ist. Die Determinante der Matrix verschwindet nicht, falls $\sin b \neq 0$, was genau dann der Fall ist, wenn $b \neq k\pi$ mit $k \in \mathbb{Z}$. D.h. das System ist nur eindeutig lösbar für $b \neq k\pi$. Folgende Fälle sind möglich:

- Eindeutigkeit liegt vor, falls $b \neq k\pi$, $k \in \mathbb{Z}$;
- Keine Lösung existiert z.B. wenn $b = \pi$ (aus $y(b) = \beta$ folgt $y(b) = y(\pi) = c_1(\sin \pi) + \alpha(\cos \pi) = \beta$, d.h. keine Lösung, wenn $-\alpha \neq \beta$);
- mehrere Lösungen.

Dies ist ein Beispiel dafür, dass die Lösungsverhältnisse nicht so klar wie bei Anfangswertproblemen sind. ∎

Zur Existenz und Eindeutigkeit von Randwertproblemen zweiter Ordnung kann man zeigen:

Satz 6.2. Betrachte die Randwertaufgabe (6.1). Sei $f(x, u, v)$ bzgl. $u,\ v$ differenzierbar mit

$$\frac{\partial f}{\partial u}(x, u, v) \geq 0$$

für alle $x \in [a, b],\ u,\ v$. Dann besitzt (6.1) eine eindeutige Lösung. ∎

Beachte: in Beispiel 6.1a ist $\frac{\partial f}{\partial u} = -1$, man kann also nicht auf Eindeutigkeit einer Lösung schließen.

Zum Beweis: Zurückführung auf eine Anfangswertaufgabe. Dann folgen Existenz und Eindeutigkeit mit der Differenzierbarkeit und Lipschitz–Stetigkeit von f. ∎

6.2 Das einfache Schießverfahren

Die Idee des einfachen Schießverfahrens (Simple Shooting Method) besteht darin, anstelle des Randwertproblems (6.1) auf (a, b) das Anfangswertproblem

$$y'' = f(x, y, y'), \quad y(a) = \alpha, \quad y'(\alpha) = s \tag{6.3}$$

für ein zu bestimmendes s zu betrachten. Aus der LIPSCHITZ–Stetigkeit von f folgt die eindeutige Lösbarkeit von (6.3). Die Lösung von (6.3) sei mit $y(x; s)$ bezeichnet. Bestimme nun s so, dass $y(b; s) = \beta$. Das bedeutet, $y(x; s)$ löst die Randwertaufgabe (6.1) genau dann, wenn $y(b; s) = \beta$ gilt. (6.1) ist also äquivalent zur Lösung der Nullstellenaufgabe (nichtlinear in s)

$$F(s) := y(b; s) - \beta = 0. \tag{6.4}$$

Jede Auswertung von $F : \mathbb{R} \to \mathbb{R}$ verlangt die numerische Lösung der Anfangswertaufgabe (6.3). In Kapitel 2 aus dem ersten Teil wurden verschiedene Möglichkeiten zur Lösung von (6.4) vorgestellt (Bisektions–, NEWTON–Verfahren, ...). Wir wenden unter der Annahme, dass $\frac{\partial}{\partial s} y(b; s)$ (und damit $F'(s)$) existiert, das NEWTON–Verfahren an:

1.) Wähle einen Startwert $s^{(0)}$.

2.) Für $i = 0, 1, 2, \ldots$ setze $s^{(i+1)} = s^{(i)} - \frac{F(s^{(i)})}{F'(s^{(i)})}$.

In jedem Schritt werden $F(s^{(i)})$ und $F'(s^{(i)})$ benötigt.

Zur Bestimmung von $F(s^{(i)})$ löse das Anfangswertproblem

$$y'' = f(x, y, y'), \quad y(a) = \alpha, \quad y'(a) = s^{(i)} \tag{6.5}$$

und werte dessen Lösung $y(x; s^{(i)})$ an der Stelle $x = b$ aus. Dann ist $F(s^{(i)}) = y(b; s^{(i)}) - \beta$.

Zur Bestimmung von $F'(s^{(i)}) = \frac{\partial}{\partial s} y(b; s)\Big|_{s=s^{(i)}}$ setze $v(x) := \frac{\partial}{\partial s} y(x; s)$ und differenziere die Randwertaufgabe (6.1) nach s,

$$\begin{aligned} v''(x) = \frac{\partial}{\partial s}\left(\frac{\partial^2}{\partial x^2} y(x, s)\right) &= \frac{\partial}{\partial s}\big(f(x, y(x; s), y'(x; s))\big) \\ &= \frac{\partial f}{\partial y}(x, y(x; s), y'(x; s))\frac{\partial y}{\partial s} + \frac{\partial f}{\partial y'}(x, y(x; s), y'(x, s))\frac{\partial y'}{\partial s} \\ &= \frac{\partial f}{\partial y}(x, y(x; s), y'(x; s))v + \frac{\partial f}{\partial y'}(x, y(x; s), y'(x; s))v' \end{aligned}$$

und

$$\begin{aligned} v(a) &= \frac{\partial}{\partial s} y(a; s) = \frac{\partial}{\partial s}(\alpha) = 0, \\ v'(a) &= \frac{\partial}{\partial s} y'(a) = \frac{\partial}{\partial s} s = 1. \end{aligned}$$

Damit ergibt sich $v(x) = \frac{\partial}{\partial s} y(x; s)$ als Lösung von

$$v'' = \frac{\partial}{\partial y} f(x, y, y')v + \frac{\partial}{\partial y'} f(x, y, y')v', \quad v(a) = 0, \quad v'(a) = 1,$$

wobei f als Argument die Lösung $y = y(x; s^{(i)})$ der Anfangswertaufgabe (6.5) enthält. Auswertung von $v(x)$ an $x = b$ ergibt $v(b) = \frac{\partial}{\partial s} y(b; s)\Big|_{s=s^{(i)}} = F'(s^{(i)}$.

Insgesamt ist dies mit einem erheblichen Aufwand verbunden, da pro Iterationsschritt des NEWTON–Verfahrens zwei Anfangswertaufgaben zu lösen sind. In vielen Fällen sollte man daher das GAUSS–NEWTON–Verfahren (vgl. Kapitel 4, 1. Teil) anwenden; weiteres in [SB].

6.3 Die Mehrziel(–schieß–)methode

Eine hauptsächliche Schwierigkeit beim einfachen Schießverfahren ist, dass die Lösung $y(x;s)$ von $y' = f(t,y)$ mit $y(t_0) = s$ sehr empfindlich von s abhängen kann. Aufgrund der stetigen Abhängigkeit von den Anfangswerten gilt mit L als LIPSCHITZ–Konstante von f

$$|y(t,s) - y(t,\overline{s})| \leq e^{L(t-t_0)}|s - \overline{s}|,$$

d.h. es gibt beträchtliche Ungenauigkeiten in $y(x,s)$, falls s ungenau und $t \gg t_0$ ist. (Vergleiche dies etwa mit dem Problem, einen Satelliten in die Erdumlaufbahn zu schießen. Weiterhin gilt die LIPSCHITZ–Bedingung für f häufig nicht global, sondern nur lokal in einer Umgebung der Lösung. Falsche Startwerte $s^{(0)}$ tragen diese dann möglicherweise aus diesem Bereich heraus, was zum Verlust der eindeutigen Lösbarkeit führt.

Die Idee der Mehrziel(–schieß–)methode (Multiple Shooting Method) besteht darin, das Intervall (a,b) in Teilintervalle zu zerlegen und das einfache Schießverfahren pro Teilintervall anzuwenden. Dies führt zu

$$y' = f(x,y) \text{ auf } (a,b), \quad r(y(a),y(b)) = 0. \tag{6.5}$$

Angenommen, es existiere eine eindeutige Lösung von (6.5). Sei $a = x_1 < \ldots < x_m = b$ und $s_k = y(x_k)$ für $k = 1,\ldots,m$. Unser Ziel ist die *gleichzeitige* Berechnung aller Unbekannter s_k. Dazu bezeichne $y(x;x_k,s_k)$ die Lösung der Anfangswertaufgabe $y' = f(x,y)$ mit $y(x_k) = s_k$. Für jedes Teilintervall erhält man Teillösungen. Das Ziel ist, dass die Gesamtlösung keine Sprünge enthält. Beachte: Sind die s_k so, dass

$$z(x) := \begin{cases} y(x;x_k,s_k), & x \in [x_k,x_{k+1}) \\ s_m, & x = b \end{cases} \tag{6.6}$$

stetig auf $[a,b]$ ist und $r(z(a),z(b)) = 0$ gilt, dann ist z die Lösung von (6.5). (Ist z Lösung der Anfangswertaufgabe auf $[x_k,x_{k+1}]$, dann ist f mit Ansatz (6.6) stetig und damit auch z'.) Daher fordern wir

$$y(x_{k+1};x_k,s_k) = s_{k+1}, \quad k = 1,\ldots,m-1, \text{ und } r(s_1,s_m) = 0, \tag{6.7}$$

d.h. der Endwert des einen Stücks muss mit dem Anfangswert des nächsten Stücks übereinstimmen.

Dies führt auf ein hochdimensionales Nullstellenproblem $F(\mathbf{s}) = 0$, wobei $F : \mathbb{R} \to \mathbb{R}^m$ mit

$$F(\mathbf{s}) := \begin{pmatrix} F_1(s_1,s_2) \\ F_2(s_2,s_3) \\ \vdots \\ F_{m-1}(s_{m-1},s_m) \\ F_m(s_1,s_m) \end{pmatrix} := \begin{pmatrix} y(x_2;x_1,s_1) - s_2 \\ y(x_3;x_2,s_2) - s_3 \\ \vdots \\ y(x_m;x_{m-1},s_{m-1}) - s_m \\ r(s_1,s_m) \end{pmatrix}$$

z.B. mit dem NEWTON–Verfahren. Der Rechenaufwand ist wie beim einfachen Schießverfahren, nur jetzt für ein größeres Problem.

6.4 Differenzenverfahren

Betrachte auf (a,b) die Randwertaufgabe

$$-y'' + q(x)y = g(x), \quad y(a) = \alpha, \quad y(b) = \beta, \tag{6.8}$$

mit $q,\ g \in \mathcal{C}([a,b])$ und $q(x) \geq 0$ für $x \in [a,b]$. Wegen $f(x,y,y') = q(x)y - g(x)$ folgt $\frac{\partial f}{\partial y} = q(x) \geq 0$ und damit hat (6.8) nach Satz 6.2 eine eindeutige Lösung. Die Idee der numerischen Lösung besteht darin, y'' durch Differenzen zu ersetzen. Seien dazu $a = x_0 < x_1 < \ldots < x_{N+1} = b$ mit $x_j = a + jh,\ h = \frac{b-a}{N+1}$. Damit ist

$$y''(x_j) \approx \frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} =: \Delta^2 u_i, \quad u_j \approx y(x_j).$$

Unter der Annahme $y \in \mathcal{C}^4([a,b])$ folgt mit TAYLOR–Entwicklung für den lokalen Abbruchfehler

$$\tau_i := y''(x_i) - \Delta^2 y_i = -\frac{h^2}{12}(y^{(4)}(x_i + \theta_i h))$$

mit $|\theta| < 1$. Also $\tau_i = O(h^2)$. In der Differentialgleichung (approximiert und ausgewertet an $x = x_j$) gilt entsprechend

$$\begin{aligned} u_0 &= \alpha \\ -\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} + q(x_j)u_j &= g(x_j) \\ u_{N+1} &= \beta \end{aligned}$$

für $j = 1, \ldots, N$. Beachte, dass die exakte Lösung $y_j = y(x_j)$

$$A\mathbf{y} = \mathbf{b} + \tau$$

mit $\tau := (\tau_1, \ldots, \tau_N)^T$ erfüllt. Zur numerischen Lösung löst man das lineare Gleichungssystem in $\mathbf{u} = (u_1, \ldots, u_N)^T$

$$A\mathbf{u} = \mathbf{b} \tag{6.9}$$

mit

$$\begin{aligned} A &:= \frac{1}{h^2}\begin{pmatrix} 2+q_1h^2 & -1 & & 0 & \\ -1 & 2+q_2h^2 & -1 & & \\ & -1 & 2+q_3h^2 & -1 & \\ & & \ddots & \ddots & \ddots \\ & 0 & & -1 & 2+q_Nh^2 \end{pmatrix} \\ &= \frac{1}{h^2}\begin{pmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix} + \operatorname{diag}(q_1, \ldots, q_N) =: B + Q \end{aligned}$$

und

$$\mathbf{b} := \begin{pmatrix} g_1 + \frac{\alpha}{h^2} \\ g_2 \\ \vdots \\ g_{N-1} \\ g_N + \frac{\beta}{h^2} \end{pmatrix}.$$

Zur Lösbarkeit von (6.9) betrachten wir die Eigenschaften von A.

Satz 6.10. Ist $q(x_i) =: q_i \geq 0$ für $i = 1, \ldots, N$, so sind A und B positiv definit, und es gilt $0 \leq A^{-1} \leq B^{-1}$ komponentenweise.

Beweis: Betrachte die symmetrische Matrix $\tilde{B} := h^2 B$. Nach dem Satz von GERSCHGORIN 1.4.4 erfüllen die (nur reellen) Eigenwerte von $\tilde{B}$

$$|\lambda_k - 2| \leq 2,$$

d.h. $0 \leq \lambda_k \leq 4$. Weiter sind alle $\lambda_k > 0$. Denn wäre ein $\lambda_k = 0$ für ein k, so wäre $\det(\tilde{B} - \lambda_k I) = \det \tilde{B} = 0$ im Widerspruch zu $\det \tilde{B} = N + 1$. Also $0 < \lambda_k \leq 4$ für alle k. Damit ist $\tilde{B}$ positiv definit, d.h. $v^T \tilde{B} v > 0$ für alle $v \not\equiv 0$. Also ist B positiv definit. Weiter gilt für die Eigenwerte λ_k von $\tilde{B}$: $0 < \lambda_k < 4$, da sonst $\tilde{B} - 4I$ singulär wäre. Es gilt aber

$$\tilde{B} - 4I = \begin{pmatrix} -2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & -2 \end{pmatrix} = -D\tilde{B}D^{-1}$$

mit $D = \operatorname{diag}(1, -1, 1, \ldots)$, was im Widerspruch zur Nichtsingularität steht. Die positive Definitheit von A folgt nun aus

$$v^T A v = v^T (B + Q) v = v^T B v + v^T Q v > 0.$$

Also ist A positiv definit und damit auch invertierbar. Noch zu zeigen ist $0 \leq A^{-1} \leq B^{-1}$. Betrachte dazu $h^2 A = D(I - J)$ mit $D = \operatorname{diag}(2 + h^2 q_1, \ldots, 2 + h^2 q_N)$ und

$$J = \begin{pmatrix} 0 & \frac{1}{2+h^2 q_1} & & \\ \frac{1}{2+h^2 q_2} & \ddots & \ddots & \\ & \ddots & \ddots & \frac{1}{2+h^2 q_{N-1}} \\ & & \frac{1}{2+h^2 q_N} & 0 \end{pmatrix}$$

sowie $h^2 B = \hat{D}(I - \hat{J})$ mit $\hat{D} = 2I$ und

$$\hat{J} = \begin{pmatrix} 0 & \frac{1}{2} & & \\ \frac{1}{2} & \ddots & \ddots & \\ & \ddots & \ddots & \frac{1}{2} \\ & & \frac{1}{2} & 0 \end{pmatrix}.$$

Da $q_i \geq 0$ für alle i gilt $0 \leq \hat{D} \leq D$ komponentenweise und $0 \leq J \leq \hat{J}$. Wegen $\hat{J} = \frac{1}{2}(-\tilde{B} + 2I)$ und $0 < \lambda_k < 4$ für λ_k im Spektrum von $\tilde{B}$ folgt $-1 < \mu_i < 1$ für die Eigenwerte μ_i von $\hat{J}$, also ist der Spektralradius $\rho(\hat{J}) < 1$. Damit konvergiert die Reihe $0 \leq I + \hat{J} + \hat{J}^2 + \ldots = (I - \hat{J})^{-1}$ und wegen $0 \leq J \leq \hat{J}$ gilt dies auch für J, $0 \leq I + J + J^2 + \ldots = (I - J)^{-1} \leq (I - \hat{J})^{-1}$. Schließlich folgt

$$0 \leq (h^2 A)^{-1} = (D(I - J))^{-1} = (I - J)^{-1} D^{-1} \leq (I - \hat{J})^{-1} \hat{D}^{-1} = (h^2 B)^{-1}$$

und damit die Behauptung. ∎

Zur Abschätzung des globalen Fehlers

Satz 6.11. Die Lösung der Randwertaufgabe (6.8) erfülle $y \in \mathcal{C}^4([a, b])$ und $\|y^{(4)}\|_\infty \leq M$. Sei $q(x) \geq 0$ für alle $x \in [a, b]$ und sei $\mathbf{u} = (u_1, \ldots, u_N)^T$ Lösung von (6.9). Dann gilt

$$|y(x_i) - u_i| \leq M \frac{h^2}{4} (x_i - a)(b - x_i) \tag{6.12}$$

für alle $i = 1, \ldots, N$. D.h. die Konvergenzordnung entspricht der Ordnung des lokalen Abbruchfehlers $\tau_i = O(h^2)$ und Konvergenz liegt für $h \to 0$ vor.

Beweis: Sei $\mathbf{y}$ die exakte Lösung und $\mathbf{u}$ die Lösung von (6.9). Dann gilt $A(\mathbf{y}-\mathbf{u}) = \tau$, d.h. $\mathbf{y}-\mathbf{u} = A^{-1}\tau$. Es sei $|\mathbf{y}| := (|y_1|, \ldots, |y_N|)^T$. Mit dieser Notation gilt

$$|\mathbf{y}-\mathbf{u}| = |A^{-1}\tau| \leq B^{-1}|\tau| \leq M\frac{h^2}{12}B^{-1}\mathbf{e}$$

mit $\mathbf{e} = (1, \ldots, 1)^T$. Schätze nun $B^{-1}\mathbf{e}$ weiter ab. Betrachte dazu das spezielle Randwertproblem $-y'' = 1$ mit $y(a) = y(b) = 0$. Lösung ist $y(x) = \frac{1}{2}(x-a)(b-x)$. Da $y^{(4)} \equiv 0$, ist $\tau_i \equiv 0$ für alle i, d.h. $u_i = y(x_i)$ für alle i; die diskrete Lösung stimmt mit der exakten überein. Hier gilt speziell $B = A$ und $\mathbf{b} = \mathbf{e}$, also $\mathbf{u} = A^{-1}\mathbf{e}$, womit (6.12), also

$$|y(x_i) - u_i| \leq M\frac{h^2}{12}\frac{1}{2}(x_i - a)(b - x_i)$$

für alle $i = 1, \ldots, N$ folgt. ∎

Bemerkung: Obiger Satz ist (wie bei Anfangswertaufgaben) ein Beispiel des Typs von Aussagen

$$\text{Konsistenz} \;+\; \text{Stabilität} \;\Longrightarrow\; \text{Konvergenz.}$$

Das Verfahren $A_h u_h = b_h$ zusammen mit dem lokalen Abbruchfehler $\tau_h = A_h y_h - b_h$ ($y_h = y|_{I_h}$) und dem globalen Fehler $e_h := y_h - u_h$ ergibt

$$\tau_h = A_h y_h - b_h = A_h y_h - A_h u_h = A_h e_h,$$

also $e_h = A_h^{-1}\tau_h$, und damit

$$\|e_h\|_\infty \leq \|A_h^{-1}\|_\infty \|\tau_h\|_\infty.$$

Es gilt $\|A_h^{-1}\| \leq M$ für alle h (Stabilität), und $\|\tau_h\|_\infty = O(h^p)$, was Konsistenz bedeutet.

Schlussbemerkungen

Weitergehende Verfahren sind Variationsmethoden, unter Verwendung von Finiten Elementen und B–Splines. Diese werden allgemein in der Vorlesung über Numerik partieller Differentialgleichungen für multivariate Probleme diskutiert. Einen Vergleich der Verfahren für Randwertprobleme (von gewöhnlichen Differentialgleichungen) findet man in [SB]. Mit der Mehrzielmethode ist das Erreichen einer höheren Ordnung leichter, ist aber wegen des zu lösenden hochdimensionalen Nullstellenproblems insgesamt sehr aufwendig. Das Differenzenverfahren hat grundsätzlich Ordnung $p = 2$. Höhere Genauigkeit erhält man nur durch Verkleinerung der Schrittweite. Dabei schaukeln sich allerdings Rundungsfehler nicht auf.

Literatur

[A] H. Arndt, *Gewöhnliche Differentialgleichungen*, Vorlesungsskript, Institut für Angewandte Mathematik, Universität Bonn, 2000.

[Br] D. Braess, *Finite Elemente*, 3. Auflage 2003, Springer, Berlin et al.

[C] P. G. Ciarlet, *Introduction to Numerical Linear Algebra and Optimisation*, 1. Auflage 1989, Cambridge University Press.

[DB] P. Deuflhard, F. Bornemann, *Numerische Mathematik 2. Gewöhnliche Differentialgleichungen*, 2. Auflage 2002, de Gruyter Lehrbuch, Berlin et al.

[DH] P. Deuflhard, A.Hohmann, *Numerische Mathematik 1. Eine algorithmisch orientierte Einführung*, 3. Auflage 2002, de Gruyter Lehrbuch, Berlin et al.

[DR] W. Dahmen, A. Reusken, *Numerik für Ingenieure und Naturwissenschaftler*, 2. Auflage 2008, Springer.

[GL] G.H. Golub, C.F. van Loan, *Matrix Computations*, 3rd Edition 1996, Johns Hopkins University Press, Baltimore et al.

[Gr] R.D. Grigorieff, *Numerik gewöhnlicher Differentialgleichungen*, Band 1, 1972, Teubner.

[H] H. Heuser, *Gewöhnliche Differentialgleichungen*, 5. Auflage 2006, Wiesbaden.

[HB] M. Hanke–Bourgeois, *Grundlagen der Numerischen Mathematik und des wissenschaftlichen Rechnens*, 2. Auflage 2006, Teubner, Stuttgart et al.

[HH] G. Hämmerlin, K.-H. Hoffmann, *Numerische Mathematik*, 4. Auflage 1994, Springer, Berlin et al.

[HNW] E. Hairer, S.P. Nørsett, G. Wanner, *Solving Ordinary Differential Equations I: Nonstiff Problems*, 2nd Edition 1993, Springer, Berlin et al.

[KR] B. Kernighan, D. Ritchie, *Programmieren in C*, 2. Ausgabe 1990, Carl Hanser Verlag, München et al.

[NM] M. Nowak, R.M. May, *Virus Dynamics: Mathematical Principles of Immunology and Virology*, 2000, Oxford University Press.

[OR] J. M. Ortega, W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, 1970, Academic Press.

[S] J. Stoer, *Numerische Mathematik 1*, 10. Auflage 2007, Springer, Berlin et al.

[SB] J. Stoer, R. Bulirsch, *Numerische Mathematik 2*, 5. Auflage 2005, Springer, Berlin et al.

[SB2] J. Stoer, R. Bulirsch, *Introduction to Numerical Analysis*, 3nd Edition 2002, Springer, Berlin et al.

[SK] H.R. Schwarz, N. Köckler, *Numerische Mathematik*, 5. Auflage 2004, Teubner, Wiesbaden.

[W] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, 1999, Oxford University Press, Oxford.

[Wa] W. Walter, *Gewöhnliche Differentialgleichungen*, 7. Auflage 2000, Springer, Berlin et al.