Skript zur Vorlesung
Numerical Methods for ODEs

Dr. A. Naumann

gehalten im WS 2020/21
Technische Universität Chemnitz

TECHNISCHE UNIVERSITÄT
CHEMNITZ

Website zur Vorlesung:
https://www.tu-chemnitz.de/mathematik/part_dgl/teaching/WS2020_Numerik_von_ODEs/

Dieses Vorlesungsskript wurde im WS 2006/07 erstellt von Roland Herzog unter Benutzung früherer Vorlesungen und Übungen von Prof. Peter Benner, Prof. Rüdiger Weiner, Prof. Joachim Stöckler, Prof. Hans Josef Pesch und Prof. Rolf Rannacher, denen ich herzlich für das Zurverfügungstellen der Unterlagen danke.

Im WS 2007/08, WS 2012/2013 und WS 2013/2014 wurde es durch Rene Schneider teilweise überarbeitet und erweitert. Im WS 2010/11 hat Herbert Egger die Vorlesung gehalten und viele Hinweise zur Überarbeitung gegeben, die im WS 2011/12 zu einer wesentlichen Verbesserung geführt haben.

Im WS 2018/2019 wurde die Vorlesung von Max Winkler gehalten und das Skript umfangreich überarbeitet. Da das Vorlesungsskripts zu umfangreich war, wurden folgende Aspekte aus dem Skript entfernt:

- Kollokationsverfahren
- Symplektische Verfahren
- Gittersteuerung durch verschiedene Schrittweiten
- Exponentielle Integratoren

Material für: 21–22 Vorlesungen à 90 Minuten

Im WS 2020/2021 wurde die Vorlesung von Andreas Naumann gehalten und das Skript im wesentlichen übernommen. Die Hinweise zu den unstetigen Galerkin Verfahren am Ende von Abschnitt 5 umformuliert. Des Weiteren wurden die Aspekte

- Linear-implizite Runge-Kutta-Verfahren
- Verfahren mit dichter Ausgabe
- Differentiell-algebraische Systeme

für ca 30 Vorlesungen wieder aufgenommen.

Fehler und Kommentare bitte an: andreas.naumann@mathematik.tu-chemnitz.de

Stand: 8. März 2021

Inhaltsverzeichnis

Kapitel 0. Einführung und Motivation 5

§ 1 Aufgabenstellung, Beispiele und Ziele der Vorlesung 5

§ 2 Existenz lokaler Lösungen 9

Kapitel 1. Numerische Lösung von Anfangswertproblemen 11

§ 3 Einschrittverfahren 12

§ 3.1 Grundbegriffe und ein erstes Verfahren 12

§ 3.2 Konvergenzanalysis allgemeiner Einschrittverfahren 17

§ 3.3 Runge-Kutta-Verfahren 25

3.3.1 Allgemeine Ordnungsbedingungen 28

3.3.2 Explizite Runge-Kutta-Verfahren 31

3.3.3 Implizite Runge-Kutta-Verfahren 33

§ 3.4 Stabilitätsbegriffe bei Einschrittverfahren 42

3.4.1 A-Stabilität 42

3.4.2 Steife Differentialgleichungen 47

3.4.3 L-Stabilität 52

§ 3.5 Gittersteuerung durch eingebettete Runge-Kutta-Verfahren 53

§ 3.6 Einschrittverfahren für Dgl zweiter Ordnung 59

§ 4 Mehrschrittverfahren 61

§ 4.1 Einleitung und Grundbegriffe 61

§ 4.2 Konvergenzuntersuchung bei Mehrschrittverfahren 71

§ 4.3 Stabilitätsbegriffe bei Mehrschrittverfahren 83

§ 4.4 Praktische Aspekte bei Mehrschrittverfahren 88

§ 5 Unstetige Galerkin-Verfahren 91

§ 6 Kollokationsverfahren und IRKV 98

§ 7 Linear implizite Runge-Kutta-Verfahren 102

§ 8 Differentiell-algebraische Systeme 106

§ 8.1 Einführung 106

§ 8.2 Eigenschaften linearer DAE-Systeme 108

§ 8.3 Numerische Behandlung linearer DAE-Systeme 117

§ 9 Symplektische Verfahren 121

Literatur 125

Index 127

Kapitel 2. Numerische Lösung von Randwertaufgaben 131

§ 10 Schießmethoden 132

§ 11 Finite-Differenzen-Verfahren 133

§ 11.1 Implementierung 133

§ 11.2 Konvergenz Finiter-Differenzen-Verfahren 135

§ 12 Numerische Methoden für Anfangs-Randwert-Probleme 138

§ 12.1 Finite-Differenzen-Diskretisierung 138

§ 12.2 Zeitdiskretisierung mittels unstetigem Galerkin-Verfahren 140

§ 12.3 Die vertikale Linienmethode 140

KAPITEL 0

Einführung und Motivation

Inhalt

§ 1 Aufgabenstellung, Beispiele und Ziele der Vorlesung 5

§ 2 Existenz lokaler Lösungen 9

§ 1 Aufgabenstellung, Beispiele und Ziele der Vorlesung

Wir betrachten in dieser Vorlesung hauptsächlich **Anfangswertprobleme** für Systeme gewöhnlicher Differentialgleichungen (Dgl) erster Ordnung:

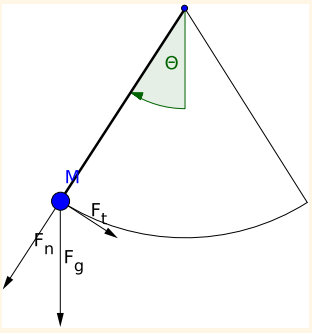
$$\left.\begin{aligned} &\text{Gesucht ist eine differenzierbare Funktion } y : [0,T] \to \mathbb{R}^n, \\ &\text{sodass } y'(t) = f(t, y(t)) \text{ für alle } t \in [0,T] \\ &\text{und } y(0) = y_a \text{ gilt.} \end{aligned}\right\} \qquad (\textbf{AWP})$$

Dabei sind die **rechte Seite** $f : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$, der **Anfangswert** (**AW**) $y_a \in \mathbb{R}^n$, die **Anfangszeit** $t = 0$ und die **Endzeit** $t = T$ gegeben.

Die unabhängige Variable t spielt in vielen, aber nicht in allen Problemen die Rolle der Zeit. Die Ableitung $y'(t)$ wird dann oft mit $\dot{y}(t)$ bezeichnet. Die gesuchte Funktion $y(t)$ heißt auch der **Zustand** der Dgl. Ist die rechte Seite f von t unabhängig, so heißt das Problem **autonom**.

Zahlreiche Fragestellungen führen auf Probleme von diesem Typ.

Beispiel 1.1 (Einfaches Pendel)**.**



Gegeben sei ein einfaches Pendel (siehe Abbildung). Folgende Daten sind gegeben:

- die Länge des Pendels L,
- die Masse m des punktförmigen Objekts,
- die Gravitationskonstante g,
- Anfangsauslenkung Θ_0 und -winkelgeschwindigkeit ω_0.

- **Modellierung:** Die Tangentialkraft ergibt sich aus der Gewichtskraft $F_g = -mg\mathbf{e}_2$ durch $|F_t| = \sin(\Theta)|F_g| = |\sin(\Theta)|mg$. Mit dem Newton'schen Kraftgesetz finden wir eine Beziehung zwischen Tangentialbeschleunigung a_t und Tangentialkraft, nämlich

$$a_t = \frac{1}{m} F_t = -\sin(\Theta) g.$$

Für die Winkelbeschleunigung gilt außerdem $\ddot{\Theta} = \frac{1}{L} a_t$ und somit erhalten wir die Differentialgleichung

$$\ddot{\Theta} + \frac{g}{L} \sin(\Theta) = 0.$$

Ferner erhalten wir aus Messungen eine Anfangsauslenkung $\Theta(0) = \Theta_0$ sowie eine initiale Winkelgeschwindigkeit $\dot{\Theta}(0) = \omega_0$.

- **Auf einen bekannten Fall zurückführen:** Wir können diese DGL 2. Ordnung in System erster Ordnung überführen. Dazu führen wir die Variablen $u = \Theta$, $v = \dot{u} = \dot{\Theta}$ ein. Unser Lösungsvektor lautet $y(t) = (u(t)\ v(t))^\top$. Dies führt auf das System

$$\dot{y}(t) = \begin{pmatrix} \dot{u}(t) \\ \dot{v}(t) \end{pmatrix} = f(t, y(t)) := \begin{pmatrix} v(t) \\ -\sin(u(t)) \frac{g}{L} \end{pmatrix}.$$

- **Konstruktion eines numerischen Verfahrens**: Wir formulieren das einfachste Verfahren, das **explizite Euler-Verfahren**. Die Grundidee besteht darin, dass man eine Näherungslösung sucht, welche die Differentialgleichung näherungsweise in den Punkten eines Gitters $\{t_k\}_{k=0}^N$ mit $t_k = hk$ (äquidistant) erfüllt. Dabei ist h der Schrittweitenparameter (üblicherweise klein). Die Ableitung wird durch einen Differenzenquotienten approximiert, d. h. $\dot{y}(t_k) \approx \frac{1}{h}(y(t_{k+1}) - y(t_k))$. In jedem Gitterpunkt soll die Näherungslösung y_h die daraus folgende Differenzengleichung erfüllen:

$$\begin{aligned} y_h(t_0) &= y_0, \\ y_h(t_{k+1}) &= y_h(t_k) + h\, f(t_k, y_h(t_k)),\ k = 0, 1, \ldots, N-1. \end{aligned}$$

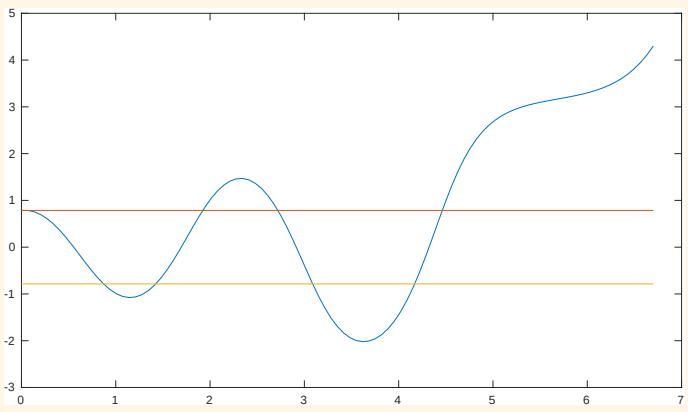
- **Matlab-Implementierung**

```
function [t,y] = euler_expl(f, time, y0, N)
% Function solves the initial value problem
%
%   y'(t) = f(t,(y(t)),   y(ta)=y0
%
% Syntax:
%  y = euler_expl(f, time, y0, N)
% Input parameters:
%  f, handle to the function f(t,y)
%  time=[ta, tb], array containing initial and final time
%  y0 (column vector), initial value at time ta
%  N, number of subintervals
% Output parameters:
%  t, time grid as a column vector
%  y, solution vector containing the entry y(t_k) in the k-th row

    h = (time(2)-time(1))/N;
    t = [time(1):h:time(2)]';
```

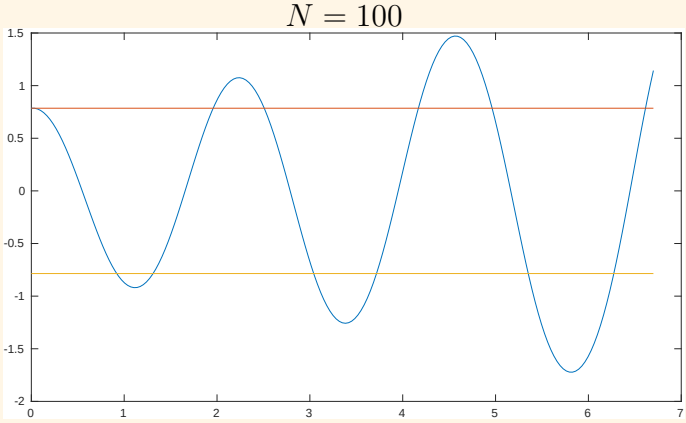
```
    y = zeros(N+1,size(y0,1));
    y(1,:) = y0';

    for k=1:N
       y(k+1,:) = y(k,:) + h*f(t(k),y(k,:))';
    end
end
```

```
% Number of subintervals
N = 200;

% Initial and final time
ta = 0.;
tb = 10.;

% Problem-specific data
y0 = [pi/4; 0];
f = @(t,y) [y(2); -9*sin(y(1))];

% Solution algorithm
[x_expl,y_expl] = euler_expl(f, [ta tb], y0, N);
[x_ode45,y_ode45] = ode45(f, [ta tb], y0);

% Plot solution
figure(1);
plot(x_expl,y_expl(:,1), 'r-', ...
     x_ode45, y_ode45(:,1),'b-');

legend('Our method', 'ODE45');

hold on
plot([ta, tb], [y0(1) y0(1)]);
plot([ta, tb],-[y0(1) y0(1)]);
hold off
```

- **Auswertung:**



$N = 100$



$N = 200$

- **Fazit:** Das Verfahren ist offensichtlich nicht für diese Problemklasse geeignet. Die maximale Auslenkung wird immer größer, was der Physik widerspricht. Selbst bei sehr kleinen Schrittweiten sind immer noch deutliche Abweichungen von der exakten Lösung zu erkennen.

Beispiel 1.2 (Räuber-Beute-Modell)**.** Heuser, 1991, § 59 Wir nehmen an, eine Population y_1 (Beutetiere) lebe von einer ausreichend vorhandenen Nahrungsquelle. Der einzige natürlich Feind ist eine Population y_2 von Raubtieren, die auf y_1 angewiesen sind. Ein einfaches Modell für die Populationsentwicklung ist gegeben durch

$$y_1'(t) = \alpha_1 y_1(t) - \beta_1 y_1(t) y_2(t)$$
$$y_2'(t) = -\alpha_2 y_2(t) + \beta_2 y_1(t) y_2(t).$$

Dabei sind $\alpha_1 > 0$ und $\alpha_2 > 0$ die natürliche Vermehrungs- bzw. Verminderungsraten der Beute bzw. der Räuber in Abwesenheit der jeweils anderen Population. Der Wechselwirkungsterm geht von einer Begegnungshäufigkeit von Räuber und Beute aus, die über Konstanten $\beta_1, \beta_2 > 0$ proportional zu beiden Beständen ist. Ein typischer Lösungsverlauf ist in Abbildung 1.1 zu sehen.

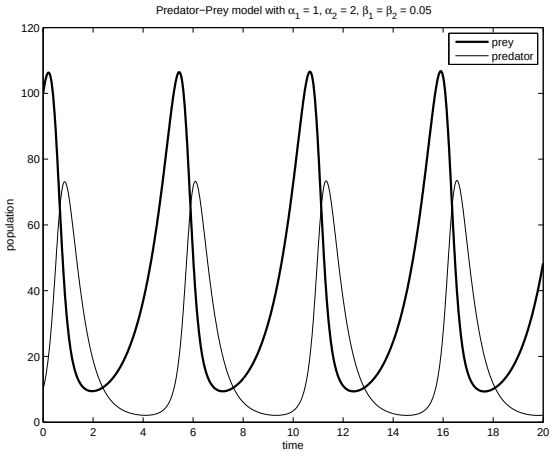


ABBILDUNG 1.1. Verlauf der Lösung aus Beispiel 1.2 (Räuber-Beute-Modell) für die Parameterwahl $\alpha_1 = 1$, $\alpha_2 = 2$, $\beta_1 = \beta_2 = 0.05$ zum Anfangswert $y(0) = (100, 10)^\top$.

Das (**AWP**) wird man nur in seltenen Fällen analytisch lösen können. Wir behandeln deshalb in dieser Vorlesung numerische Verfahren für die näherungsweise Lösung von (**AWP**). Um die Brauchbarkeit der numerischen Lösung beurteilen zu können, benötigen wir Aussagen über ihre Genauigkeit.

Ziele der Vorlesung:

- das Kennenlernen von Verfahren zur numerischen Lösung von (**AWP**),
- die Analysis dieser Verfahren bzgl. Konvergenz und Stabilität,
- deren praktische Umsetzung in Computerprogramme sowie
- Techniken der Schrittweitensteuerung zur effizienten numerischen Lösung.

§ 2 Existenz lokaler Lösungen

Wir wiederholen kurz einige wichtige Ergebnisse aus der Theorie der gewöhnlichen Differentialgleichungen sowie der Analysis. Hier und in Zukunft bedeutet $\|y\|$ für einen Vektor $y \in \mathbb{R}^n$ immer die Euklidische Norm, also $\|y\| = (y^\top y)^{1/2}$. (Wir könnten auch jede andere Norm verwenden.)

Satz 2.1 (Picard-Lindelöf [1])**.** Es sei

$$Q = [a_1, b_1] \times [a_2, b_2] \times \cdots \times [a_n, b_n] \subset \mathbb{R}^n$$

ein kompakter Quader und $y_a \in \operatorname{int}(Q)$. Weiterhin sei $f : [0, T] \times Q \to \mathbb{R}^n$ stetig und bzgl. y (gleichmäßig) Lipschitz-stetig, d. h., es gibt ein $L > 0$, sodass

$$\|f(t, y_1) - f(t, y_2)\| \le L \, \|y_1 - y_2\| \tag{2.1}$$

gilt für alle $t \in [0, T]$ und alle $y_1, y_2 \in Q$. Dann gibt es ein maximales Intervall $[0, T^*] \subset [0, T]$ und darauf genau eine Lösung des (**AWP**), die in Q verläuft.

Satz 2.2 (Stetige Abhängigkeit von den Anfangswerten)**.** Unter den Voraussetzungen von Satz 2.1 gilt für die Lösung $y(t; s)$ des (**AWP**)

$$y'(t) = f(t, y(t)), \quad y(0) = s$$

zu verschiedenen AW $s_1, s_2 \in \operatorname{int}(Q)$ die Abschätzung

$$\|y(t; s_1) - y(t; s_2)\| \le \mathrm{e}^{L\,t} \|s_1 - s_2\| \tag{2.2}$$

für alle $t \ge 0$ aus dem gemeinsamen Intervall der Lösungen zu s_1 und s_2.

Voraussetzung 2.3 (Generalvoraussetzung)**.** Wir betrachten das (**AWP**). Wir wollen im Folgenden stets voraussetzen, dass ...

(a) die rechte Seite $f : [0, T] \times Q \to \mathbb{R}^n$ zumindest auf einem kompakten Quader $[0, T] \times Q$ stetig ist und dort der Lipschitz-Bedingung (2.1) genügt.

(b) für den gegebenen AW $y_a \in Q$ die Lösung auf dem gesamten Intervall $[0, T]$ in Q bleibt. (Gegebenfalls machen wir T kleiner.)

[1] Heuser, 1991, Satz 12.2, Aufgabe 12.6

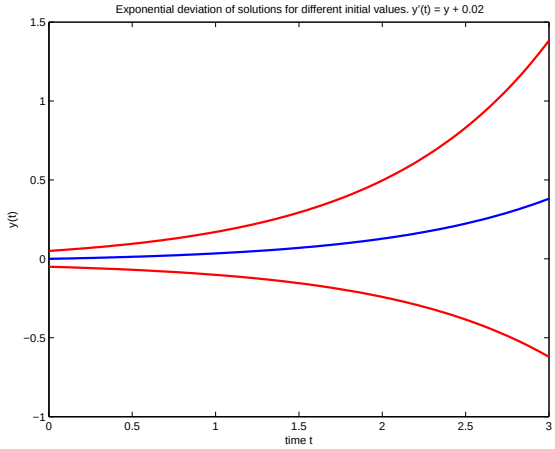


ABBILDUNG 2.1. Illustration der exponentiellen Abweichung (2.2) über die Zeit bei der Lösung eines (**AWP**) zu verschiedenen Anfangswerten.

Bemerkung 2.4 (Hinreichende Bedingung für Lipschitz-Stetigkeit)**.** Es sei $f : [0,T] \times Q \to \mathbb{R}^n$ stetig. Falls f außerdem stetige erste partielle Ableitungen $\partial f/\partial y_i$, $i = 1, \ldots, n$ besitzt, so ist die Lipschitz-Bedingung (2.1) mit

$$L = \max_{(t,y)\in[0,T]\times Q} \|f_y(t,y)\|$$

erfüllt.[2]

[2]Dabei ist $\|f_y(t,y)\|$ die durch die Euklidische Vektornorm induzierte Matrixnorm (Spektralnorm) der Jacobimatrix $f_y(t,y)$.

KAPITEL 1

Numerische Lösung von Anfangswertproblemen

Inhalt

§ 3 Einschrittverfahren 12

§ 3.1 Grundbegriffe und ein erstes Verfahren 12

§ 3.2 Konvergenzanalysis allgemeiner Einschrittverfahren 17

§ 3.3 Runge-Kutta-Verfahren 25
3.3.1 Allgemeine Ordnungsbedingungen 28
3.3.2 Explizite Runge-Kutta-Verfahren 31
3.3.3 Implizite Runge-Kutta-Verfahren 33

§ 3.4 Stabilitätsbegriffe bei Einschrittverfahren 42
3.4.1 A-Stabilität 42
3.4.2 Steife Differentialgleichungen 47
3.4.3 L-Stabilität 52

§ 3.5 Gittersteuerung durch eingebettete Runge-Kutta-Verfahren 53

§ 3.6 Einschrittverfahren für Dgl zweiter Ordnung 59

§ 4 Mehrschrittverfahren 61

§ 4.1 Einleitung und Grundbegriffe 61

§ 4.2 Konvergenzuntersuchung bei Mehrschrittverfahren 71

§ 4.3 Stabilitätsbegriffe bei Mehrschrittverfahren 83

§ 4.4 Praktische Aspekte bei Mehrschrittverfahren 88

§ 5 Unstetige Galerkin-Verfahren 91

§ 6 Kollokationsverfahren und IRKV 98

§ 7 Linear implizite Runge-Kutta-Verfahren 102

§ 8 Differentiell-algebraische Systeme 106

§ 8.1 Einführung 106

§ 8.2 Eigenschaften linearer DAE-Systeme 108

§ 8.3 Numerische Behandlung linearer DAE-Systeme 117

§ 9 Symplektische Verfahren 121

§ 3 Einschrittverfahren

§ 3.1 Grundbegriffe und ein erstes Verfahren

Alle in diesem Kapitel behandelten numerischen Verfahren bestimmen Näherungswerte der exakten Lösung auf einem Punktgitter.

Definition 3.1 (Gitter, Gitterfunktion)**.**

(a) Ein **Gitter** auf $[0,T]$ ist eine endliche Punktmenge

$$\mathcal{T} = \{t_0, t_1, \ldots, t_N\} \quad \text{mit} \quad 0 = t_0 < t_1 < \cdots t_N = T.$$

Die $\{t_i\}$ heißen auch die **Stützstellen**. Die Abstände $h_i = t_{i+1} - t_i$ heißen die **Schrittweiten** des Gitters $\mathcal{T}$.

(b) Für ein gegebenes Gitter bezeichnet

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

die (maximale) **Gitterweite** oder den **Diskretisierungsparameter**.

Viele Konvergenzresultate hängen von h ab. Darum werden wir dort, wo der Diskretisierungsparameter wichtig ist, $\mathcal{T}_h$ anstelle von $\mathcal{T}$ schreiben.

(c) Ist $h_i = h$ für alle i, so heißt das Gitter **äquidistant** (vgl. Beispiel 1.1).

(d) Wir suchen als Näherung der exakten Lösung $y : [0,T] \to \mathbb{R}^n$ des (**AWP**) eine **Gitterfunktion** $y_h : \mathcal{T} \to \mathbb{R}^n$, die nur auf der Punktmenge $\mathcal{T}$ definiert ist.

Der Einfachheit halber werden wir für die Funktionswerte einfach y_i statt $y_h(t_i)$ schreiben.

Die Differentialgleichung

$$y'(t) = f(t, y(t))$$

besagt, dass die gesuchte Lösung $y(t)$ an der Stelle t die Tangentensteigung $f(t, y(t))$ hat. Mit anderen Worten, die Steigung der Lösung $y(t)$ passt sich in das durch $f(t,y)$ definierte Richtungsfeld ein. Für skalare Differentialgleichungen, also im Fall $n = 1$, lässt sich das einfach grafisch veranschaulichen (Abbildung 3.1 links)[1].

Das einfachste numerische Verfahren besteht nun darin, die Lösungskurve $y(t)$ im Intervall $[t, t+h]$ durch ihre Tangente im Punkt $(t, y(t))$ zu approximieren, deren Steigung aus dem Richtungsfeld, d. h. aus der rechten Seite $f(t, y(t))$ abgelesen werden kann.

Auf einem Gitter $\mathcal{T}$ mit den Stützstellen t_k erhalten wir sukzessive die Näherungen $y_k = y_h(t_k)$ für die Werte der exakten Lösung $y(t_k)$ gemäß

$$\begin{aligned} y_h(t_{k+1}) &= y_h(t_k) + h_k\, f(t_k, y_h(t_k)) \\ \text{oder kurz} \qquad y_{k+1} &= y_k + h_k\, f(t_k, y_k), \\ \text{beginnend mit} \qquad y_0 &= y_a. \end{aligned} \tag{3.1}$$

[1] Siehe Matlab-Skript `slopefield_2a.m`

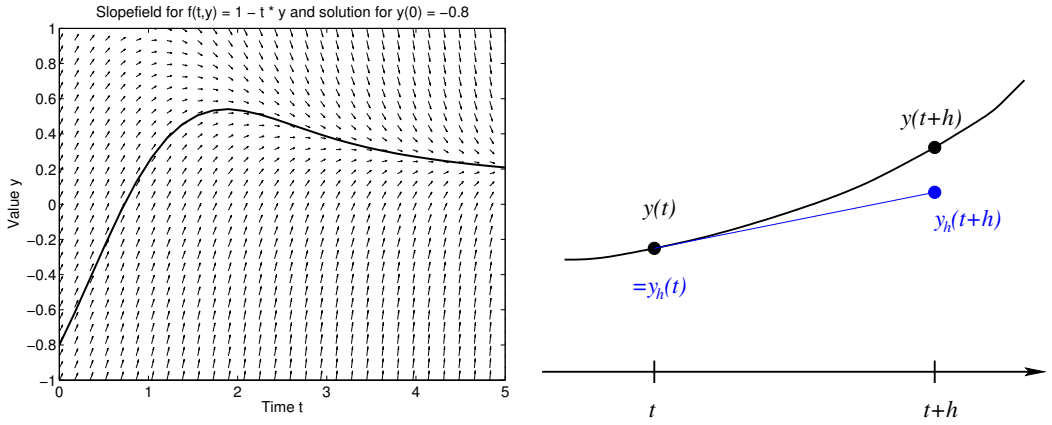


Abbildung 3.1. Richtungsfeld einer Dgl (links) und Konstruktionsidee des expliziten Euler-Verfahrens (Polygonzugmethode) (rechts).

Die Näherungsvorschrift (3.1) heißt auch **Methode von Euler**, **explizites Euler-Verfahren** oder **Polygonzugmethode**. In der englischsprachigen Literatur wird das Verfahren auch als **Vorwärts-Euler-Verfahren** (**Forward Euler**) bezeichnet.

Eine wichtige Fragestellung ist die Untersuchung der Konvergenz und Konvergenzgeschwindigkeit der numerischen Lösungen, wenn die Gitterweite $h \searrow 0$ strebt. Wir wollen dies zunächst exemplarisch für das explizite Euler-Verfahren untersuchen. Dazu bezeichnen wir für ein gegebenes Gitter $\mathcal{T}$ mit

$$e_h(t_k) := e_k := y_h(t_k) - y(t_k), \quad t_k \in \mathcal{T}$$

den **(globalen) Fehler** des Näherungsverfahrens am Gitterpunkt $t_k \in \mathcal{T}$. Auch e_h ist eine Gitterfunktion.

Die Abbildung 3.2 zeigt, dass sich der Fehler e_{k+1} an einem Gitterpunkt t_{k+1} aus zwei Anteilen zusammensetzt:

(a) aus dem Fehler des Eulerschrittes von $(t_k, y(t_k))$ nach (t_{k+1}, z_{k+1}), dem sogenannten **lokalen Diskretisierungsfehler**, und

(b) aus dem Fehler, der sich durch Fortpflanzung des Fehlers e_k ergibt, der bereits am Gitterpunkt t_k vorhanden war.

Wir schätzen beide Anteile ab.

Schritt a): Zur Abschätzung des lokal hinzukommenden Fehlers definieren wir den sogenannten **lokalen Diskretisierungsfehler** d_{k+1} des Euler-Verfahrens an der Stelle t_{k+1} über

$$\begin{aligned} h_k\, d_{k+1} &= y(t_{k+1}) - z_{k+1} \\ &= y(t_{k+1}) - y(t_k) - h_k\, f(t_k, y(t_k)). \end{aligned} \tag{3.2}$$

Man nennt d_{k+1} auch den **Konsistenzfehler** (an der Stelle t_{k+1}), da er sich aus dem Einsetzen der exakten Lösung in das diskrete Schema (hier (3.1)) ergibt.

Um den lokalen Diskretisierungsfehler abzuschätzen, verwenden wir die Taylorentwicklung:

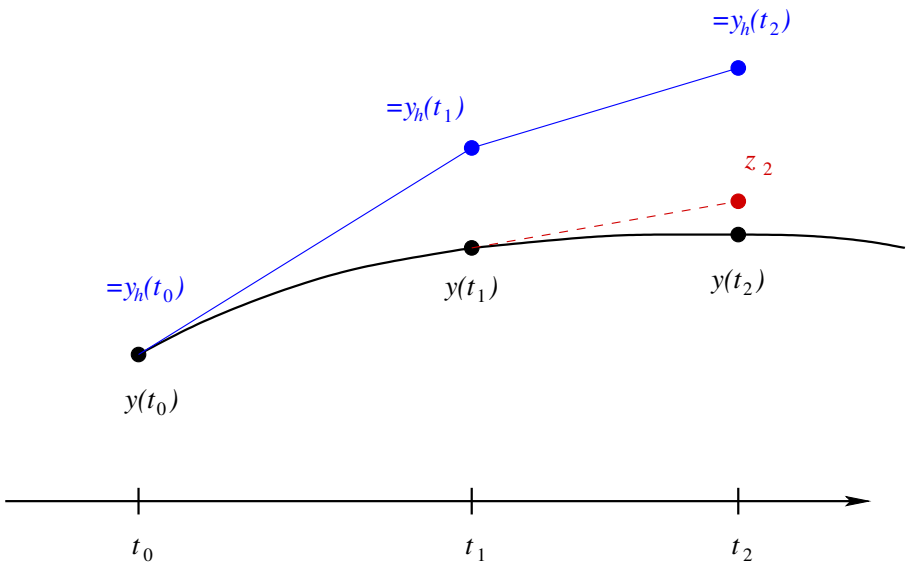


ABBILDUNG 3.2. Fehlerfortpflanzung bei einem Einschrittverfahren.

Satz 3.2 (Satz von Taylor mit Lagrange-Restglied)**.** Sei $I \subset \mathbb{R}$ ein Intervall, $x_0 \in I$ und $f\colon I \to \mathbb{R}$ eine $(m+1)$-mal stetig differenzierbare Funktion. Dann gilt

$$f(x) = \sum_{j=0}^{m} \frac{f^{(j)}(x_0)}{j!}(x - x_0)^j + \frac{f^{(m+1)}(\xi)}{(m+1)!}(x - x_0)^{m+1}$$

für ein ξ zwischen x_0 und x.

Bemerkung 3.3 (Höhere Dimensionen)**.** Der Satz von Taylor lässt sich auch auf Funktionen $f\colon \mathbb{R}^n \supset G \to \mathbb{R}^n$ verallgemeinern:

$$f(x_0 + d) = \sum_{j=0}^{m} \frac{f^{(j)}(x_0)\,[d^{(j)}]}{j!} + \int_0^1 \frac{(1-s)^m}{m!} f^{(m+1)}(x_0 + s\,d)\,[d^{(m+1)}]\,\mathrm{d}s.$$

Die j-ten Ableitungen von f sind dann als Multilinearformen zu interpretieren, d. h. $f^{(j)}(x)[d^{(j)}]$ bedeutet das j-fache Einsetzen der Richtung f in diese multilineare Abbildung. Um die Notation einfach zu halten werden wir die Beweise in dieser Vorlesung lediglich für den skalarwertigen Fall führen.

Betrachten wir nun (3.2). Mit Satz 3.2 erhalten wir eine Darstellung für $y(t_{k+1})$:

$$\begin{aligned} y(t_{k+1}) &= y(t_k) + y'(t_k)\,h_k + \frac{1}{2}y''(\xi)\,h_k^2 \\ &= y(t_k) + f(t_k, y(t_k))\,h_k + \frac{1}{2}y''(\xi)\,h_k^2, \quad \xi \in [t_k, t_{k+1}] \end{aligned}$$

Hierfür setzen wir voraus, dass y in $C^2([0,T];\mathbb{R}^n)$ liegt. Wir setzen dies in (3.2) ein und erhalten die Fehlerdarstellung

$$h_k\,d_{k+1} = \frac{1}{2}y''(\xi)\,h_k^2, \quad \xi \in [t_k, t_{k+1}].$$

Wegen der Beschränktheit von y'' erhalten wir die Abschätzung

$$\|d_{k+1}\| \leq \frac{1}{2} \max_{\xi \in [t_k, t_{k+1}]} \|y''(\xi)\| \, h_k. \tag{3.3}$$

Der lokale Diskretisierungsfehler ist also von der Größenordnung der Schrittweite h_k.

Schritt b): Zur Untersuchung der Fortpflanzung des Fehlers e_k müssen wir am Zeitpunkt t_k zwei explizite Eulerschritte betrachten zu den Startwerten $y_h(t_k)$ und $y(t_k)$, die nach Definition gerade um $e_k = y_h(t_k) - y(t_k)$ auseinander liegen:

$$\begin{aligned} y_h(t_{k+1}) &= y_h(t_k) + h_k \, f(t_k, y_h(t_k)) \\ z_{k+1} &= y(t_k) + h_k \, f(t_k, y(t_k)). \end{aligned}$$

Deren Differenz lässt sich abschätzen als:

$$\begin{aligned} \|y_h(t_{k+1}) - z_{k+1}\| &\leq \|y_h(t_k) - y(t_k)\| + h_k \, \|f(t_k, y_h(t_k)) - f(t_k, y(t_k))\| \\ &\leq (1 + h_k \, L) \, \|y_h(t_k) - y(t_k)\| \\ &= (1 + h_k \, L) \, \|e_k\|. \end{aligned} \tag{3.4}$$

Hier setzen wir voraus, dass die rechte Seite f in einer geeigneten Menge Lipschitzstetig bzgl. y mit L-Konstante L ist.[2]

Zusammen erhalten wir mit (3.2) und (3.4) also die Abschätzung

$$\begin{aligned} \|e_{k+1}\| &= \|y_h(t_{k+1}) - y(t_{k+1})\| \\ &\leq \underbrace{\|y_h(t_{k+1}) - z_{k+1}\|}_{b)} + \underbrace{\|z_{k+1} - y(t_{k+1})\|}_{a)} \\ &\leq (1 + h_k \, L) \, \|e_k\| + h_k \, \|d_{k+1}\| = \|e_k\| + h_k \, L \, \|e_k\| + h_k \, \|d_{k+1}\| \end{aligned} \tag{3.5}$$

für den globalen Fehler. Mit (3.5) haben wir eine Rekursion für die Fehlerfortpflanzung von Schritt zu Schritt gewonnen. Wiederholtes Einsetzen von $\|e_j\|, j = k, k-1, \ldots, 1$ und $\|e_0\| = 0$ liefert

$$\|e_n\| \leq \sum_{k=0}^{n-1} h_k \, L \, \|e_k\| + \sum_{k=1}^{n} h_{k-1} \, \|d_k\|. \tag{3.6}$$

Um daraus eine explizite Abschätzung des Fehlers $\|e_n\|$ zu erhalten verwenden wir

Lemma 3.4 (Diskretes Gronwall-Lemma)**.** Gegeben seien Zahlenfolgen $\delta_n > 0$, $a_n \geq 0$ und $b_n \geq 0$, welche die Ungleichung

$$a_n \leq b_n + \sum_{k=0}^{n-1} \delta_k a_k, \qquad n \geq 1,$$

erfüllen. Ferner sei $(b_n)_{n \in \mathbb{N}_0}$ nicht-fallend. Dann gilt die Abschätzung

$$a_n \leq \exp\left(\sum_{k=0}^{n-1} \delta_k \right) b_n$$

für alle $n \geq 1$.

[2] Genauer: Die diskreten Lösungen $\{y_h(t_k)\}$ müssen auch in Q liegen.

Beweis: Wir führen eine Hilfsfolge $\{S_n\}_{n\in\mathbb{N}_0}$ ein:

$$S_0 := b_0 \geq a_0, \qquad n > 0\colon\ S_n := \sum_{k=0}^{n-1} \delta_k a_k + b_n \geq a_n.$$

Man rechnet leicht nach:

$$S_n - S_{n-1} = \delta_{n-1} a_{n-1} + b_n - b_{n-1}.$$

Wir zeigen über Induktion, dass

$$(a_n \leq) S_n \leq b_n \exp\left(\sum_{k=0}^{n-1} \delta_k\right) \tag{3.7}$$

erfüllt ist. Der Induktionsanfang ist klar. Unter der Annahme, dass (3.7) für die Indizes $1, \ldots, n-1$ gilt, zeigen wir:

$$S_n \leq S_{n-1} + \delta_{n-1} \underbrace{a_{n-1}}_{\leq S_{n-1}} + b_n - b_{n-1} \leq (1+\delta_{n-1}) S_{n-1} + b_n - b_{n-1}$$

$$\overset{I.A.}{\leq} \underbrace{(1+\delta_{n-1})}_{\leq e^{\delta_{n-1}}} b_{n-1} \exp\left(\sum_{k=0}^{n-1} \delta_k\right) + b_n - b_{n-1} \leq \underbrace{\exp\left(\sum_{k=0}^{n-1} \delta_k\right)}_{\geq 1} b_{n-1} + (b_n - b_{n-1})$$

$$\leq \exp\left(\sum_{k=0}^{n-1} \delta_k\right) (b_{n-1} + b_n - b_{n-1}).$$

Dies entspricht der Behauptung. □

Nun wenden wir das diskrete Gronwall-Lemma 3.4 auf (3.6) an. Wir wählen dazu

$$a_n := \|e_n\|, \quad \delta_k := h_k L, \quad b_n := \sum_{k=1}^{n} h_k \|d_k\|$$

und erhalten

$$\begin{aligned} \|e_n\| &\leq \sum_{j=1}^{n} h_{j-1} \|d_j\| \exp\Big(\sum_{j=0}^{n-1} h_j\, L\Big) \\ &= \sum_{j=1}^{n} h_{j-1} \|d_j\| \exp(L\, t_n) \quad \text{für } n \in \mathbb{N}_0. \end{aligned}$$

Durch Einsetzen der Abschätzung (3.3) für den lokalen Fehler, also

$$\|d_j\| \leq \frac{1}{2} \max_{\xi\in[t_{j-1},t_j]} \|y''(\xi)\|\, h_{j-1} \leq \frac{1}{2} \max_{\xi\in[0,t_j]} \|y''(\xi)\|\, h$$

ergibt sich unter Beachtung von $\sum_{j=1}^{n} h_{j-1} = t_n$ folgender Konvergenzsatz:

Satz 3.5 (Konvergenz des expliziten Euler-Verfahrens)**.** Die Lösung des (**AWP**) sei $C^2([0,T];\mathbb{R}^n)$. Dann gilt für den Fehler der Näherungslösungen des expliziten Euler-Verfahrens auf einem beliebigen Gitter $\{0 = t_0, t_1, \ldots, t_N = T\}$ der Klasse $\mathcal{T}_h$ die Abschätzung

$$\|e_n\| \leq \frac{t_n}{2} e^{L\, t_n} \max_{\xi\in[0,t_n]} \|y''(\xi)\|\, h \tag{3.8}$$

für alle $n = 0, 1, \ldots, N$ und damit

$$\|e_h\|_{\infty,h} := \max_{0\le n\le N} \|e_n\| \le \frac{T}{2} \mathrm{e}^{L\,T} \max_{\xi\in[0,T]} \|y''(\xi)\|\, h. \tag{3.9}$$

Eine Kurzschreibweise dieser Abschätzung lautet

$$\|e_h\|_{\infty,h} \le Ch \tag{3.10}$$

mit einer positiven Konstanten $C = C(y, f, T)$.

Bemerkung 3.6 (Generische Konstanten)**.** Um die Notation zu vereinfachen verwenden wir häufig eine generische Konstante C. Diese ist stets unabhängig vom Gitter $\mathcal{T}$ und somit von h_i. Falls diese von der Lösung y oder den Daten y_0, f abhängt schreiben wir $C(y)$ bzw. $C(y_0, f)$. Der Wert der Konstanten kann sich auch in jeder Abschätzung ändern. Eine alternative Schreibweise für (3.10) wäre:

$$\|e_h\|_{\infty,h} = \mathcal{O}(h).$$

Bemerkung 3.7. Wesentliche Bestandteile des Konvergenzbeweises am Beispiel des expliziten Euler-Verfahrens sind:

(a) die Abschätzung der lokalen Fehlerbeiträge in jedem Schritt (lokaler Diskretisierungsfehler = Konsistenzfehler) und

(b) der Beweis einer Abschätzung für die Fehlerfortpflanzung (diskrete Stabilität des Schemas).

Im nächsten Abschnitt werden wir dieses Vorgehen auf allgemeine Einschrittverfahren übertragen.

§ 3.2 Konvergenzanalysis allgemeiner Einschrittverfahren

Als Verallgemeinerung von (3.1) betrachten wir im Rest von § 3 Verfahren der Bauart

$$y_{k+1} = y_k + h_k\, \Phi(t_k, h_k, y_k), \tag{3.11}$$

die aus der Kenntnis eines Näherungswertes y_k für $y(t_k)$ und der Schrittweite h_k einen neuen Näherungswert y_{k+1} für y an der Stelle $t_{k+1} = t_k + h_k$ bestimmen.

Definition 3.8 (Ein(zel)schrittverfahren)**.** Ein numerisches Verfahren zur Lösung des (**AWP**), das der Vorschrift (3.11) genügt, heißt **Einschrittverfahren** (ESV). Φ heißt die **Verfahrensfunktion** oder **Inkrementfunktion** des Verfahrens.[3] Das ESV heißt

- **explizit**, falls Φ eine explizite Darstellung besitzt, und andernfalls
- **implizit**.

Wie wir bereits in § 3.1 gesehen haben, spielt der **Konsistenzfehler** eine wesentliche Rolle. Er bezieht sich immer auf das Einsetzen der exakten Lösung in das Verfahren.

[3] Natürlich hängt die Verfahrensfunktion Φ auch von der rechten Seite f ab (sonst würde ja das Verfahren die Dgl gar nicht berücksichtigen), aber diese Abhängigkeit drücken wir nicht explizit aus.

Definition 3.9 (Konsistenzfehler, Konsistenzordnung bei ESV)**.** Es sei $y(\cdot)$ die Lösung der Dgl $y'(t) = f(t, y(t))$ auf dem Intervall $[0, T]$.

(a) Die Größe

$$d(y, t+h, h) := \frac{y(t+h) - y(t)}{h} - \Phi(t, h, y) \tag{3.12}$$

heißt der **Konsistenzfehler** oder **Abschneidefehler** des ESV (3.11) an der Stelle t zur Schrittweite h.

(b) Das ESV heißt **konsistent**, falls gilt:

$$\lim_{h \searrow 0} \sup_{t \in [0, T-h]} \|d(y, t+h, h)\| = 0. \tag{3.13}$$

(c) Das ESV besitzt die **Konsistenzordnung** p für das (**AWP**), falls

$$\sup_{t \in [0, T-h]} \|d(y, t+h, h)\| \leq C\, h^p \tag{3.14}$$

gilt mit einer von h unabhängigen Konstanten C, also kurz:

$$\sup_{t \in [0, T-h]} \|d(y, t+h, h)\| = \mathcal{O}(h^p).$$

Die Konsistenz und Konsistenzordnung beziehen sich dabei jeweils nur auf die Differentialgleichung des (**AWP**) auf $[t, t+h]$, es werden hier exakte Anfangswerte bei t verwendet. Aus (3.12) folgt, dass ein ESV offenbar genau dann konsistent mit dem (**AWP**) ist, wenn gilt:

$$\lim_{h \searrow 0} \Phi(t, h, y) = f(t, y(t)) \quad \text{gleichmäßig in } [0, T-h].$$

Wie bereits beim Euler-Verfahren gesehen, bestimmt man die Konsistenzordnung mittels Taylorentwicklung, vgl. (3.1). Dafür mussten wir hinreichende Glattheit der exakten Lösung y annehmen, also hinreichende Glattheit der rechten Seite f. Allgemeiner geht $\max_{t \in [0,T]} \|y^{(p+1)}(t)\|$ in die Konstante c ein.

Fazit: Ein Verfahren erreicht bei einer konkreten Aufgabenstelle „seine“ maximale Konsistenzordnung nur dann, wenn die rechte Seite hinreichend glatt ist.

Exemplarisch listen wir nun einige weitere explizite und implizite ESV auf.

Bemerkung 3.10 (Herleitung spezieller Verfahren)**.** Wir betrachten das AWP

$$\dot{y}(t) = f(t, y(t))$$

auf einem einzelnen Intervall $t \in [t_k, t_{k+1}]$.

- Approximiere die Ableitung mit dem **Differenzenquotienten**

$$y(t) \approx \frac{y_{k+1} - y_k}{h_k}.$$

Dieser approximiert die Ableitung im Mittelpunkt, also $\dot{y}(t_k + \frac{h_k}{2})$, mit Ordnung 2 (zentraler Differenzenquotient), in allen anderen Punkten $t \in [t_k, t_{k+1}] \setminus \{t_k + h_k/2\}$ mit Ordnung 1.

- Verwende eine **Approximation der rechten Seite**. Diese sollte möglichst nah an $f(t_{k+1/2}, y(t_{k+1/2}))$, $t_{k+1/2} := \frac{1}{2}(t_k + t_{k+1})$, liegen, siehe Abbildung 3.3. Da $y(t_{k+1/2})$ Unbekannt ist verwenden wir Approximationen der Form:

$$f(t_{k+1/2}, y(t_{k+1/2})) \approx f(\xi_k, \eta_k), \quad \xi_k, \in [t_k, t_{k+1}], \ \eta_k \approx y(t_{k+1/2}). \tag{3.15}$$

- Damit folgt die Verfahrensvorschrift

$$y_{k+1} = y_k + h_k f(\xi_k, \eta_k), \quad \xi_k \in [0, 1], \ \eta_k \approx y(t_{k+1/2}).$$

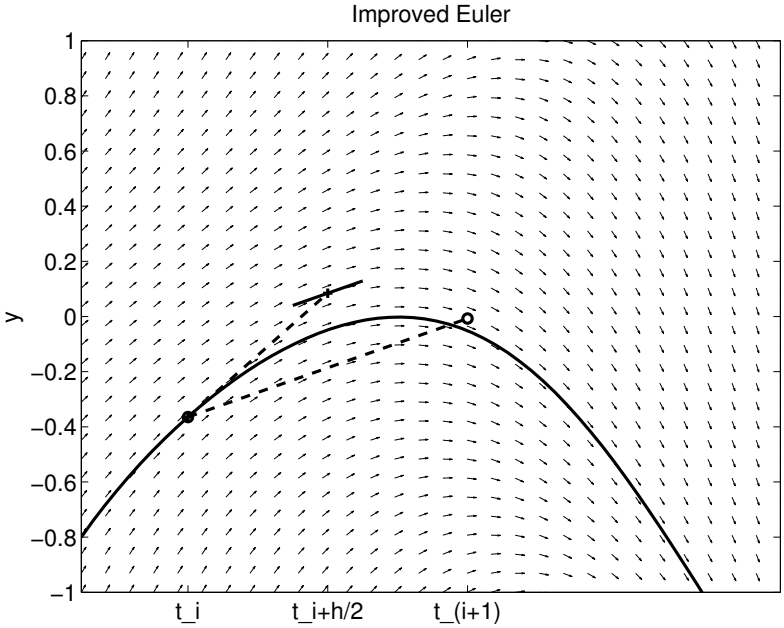


Abbildung 3.3. Geometrische Interpretation des Verbesserten Euler-Verfahrens.

Beispiel 3.11 (Einige explizite und implizite Einschrittverfahren[4])**.**

(a) Die Wahl $\xi_k = t_k$ und $\eta_k = y_k$ in Bemerkung 3.10 liefert das **explizite Euler-Verfahren** (**Forward Euler**). Die Verfahrensfunktion lautet

$$\Phi(t_k, h_k, y_h) = f(t_k, y_k) \quad \text{bzw.} \quad \Phi(t, h, y) = f(t, y).$$

Dies ist ein explizites ESV der Ordnung 1.

(b) Die Wahl $\xi_k = t_{k+1}$ $\eta_k = y_{k+1}$ in Bemerkung 3.10 liefert das **implizite Euler-Verfahren** (**Backward Euler**)

$$y_{k+1} = y_k + h_k f(t_{k+1}, y_{k+1}). \tag{$\star$}$$

In jedem Schritt muss das i. A. nichtlineare Gleichungssystem[5] gelöst werden (z.B. mit Fixpunktiteration, Newton-Verfahren). Die Verfahrensfunktion lautet

$$\Phi(t_k, h_k, y_k) = f(t_{k+1}, y_{k+1}) \overset{(\star)}{=} f(t_k + h_k, y_k + h_k\, \Phi(t_k, h_k, y_k))$$
$$bzw. \quad \Phi(t, h, y) = f(t + h, y + h\, \Phi(t, y, h)).$$

Es handelt sich um ein implizites ESV der Ordnung 1. (Lösbarkeit im Kontext allgemeinerer impliziter Verfahren, § 3.3.3)

(c) Die Wahl $\xi_k = t_{k+1/2}$, $\eta_k = y_k + \frac{h_k}{2} f(t_k, y_k)$ (halber expliziter Euler-Schritt) in Bemerkung 3.10 liefert das **verbesserte Euler-Verfahren**

$$y_{k+1} = y_k + h_k f\left(t_k + \frac{h_k}{2}, y_k + \frac{h_k}{2} f(t_k, y_k)\right)$$

Die Verfahrensfunktion lautet

$$\Phi(t, h, y) = f\left(t + \frac{h}{2}, y + \frac{h}{2} f(t, y)\right).$$

Dieses Verfahren ist ein explizites ESV der Ordnung 2. Hier werden allerdings 2 Funktionsauswertungen pro Zeitschritt benötigt.

(d) Approximiert man $f(t_{k+1/2}, y_{k+1/2})$ durch den Mittelwert $\frac{1}{2}(f(t_k, y_k) + f(t_{k+1}, y_{k+1})$, so erhält man das **Crank-Nicolson-Verfahren**:

$$y_{k+1} = y_k + \frac{h_k}{2} \left(f(t_k, y_k) + f(t_{k+1}, y_{k+1})\right).$$

Die Verfahrensfunktion lautet

$$\Phi(t, h, y) = \frac{1}{2} \left[f(t, y) + f(t + h, y + h\Phi(t, h, y))\right].$$

Dies ist ein implizites Verfahren 2. Ordnung.

(e) Ersetzt man y_{k+1} im Crank-Nicolson-Verfahren durch $y_k + h_k f(t_k, y_k)$ (Approximation durch expliziten Euler-Schritt) erhält man das **Verfahren von Heun** (Heun, 1900) mit

$$\Phi(t, h, y) = \frac{1}{2}\big(f(t, y) + f(t + h, y + hf(t, y))\big) \tag{3.16}$$

Dieses ist explizit mit Ordnung 2 und benötigt 2 f-Auswertungen pro Zeitschritt.

Alle obigen Verfahren sind Beispiele für sogenannte Runge-Kutta-Verfahren, die systematisch in § 3.3 behandelt werden. Die Struktur der mehrfachen (rekursiven) f-Auswertungen pro Zeitschritt ist typisch für diese Verfahren.

Bemerkung 3.12 (Lokaler Diskretisierungsfehler)**.** Der in Definition 3.9 eingeführte Konsistenzfehler wird auch als **lokaler Diskretisierungsfehler** bezeichnet, da wir ihn — wie bereits in § 3.1 gesehen — wie folgt interpretieren können, siehe auch Abbildung 3.4

$$\begin{aligned} y(t + h) - y_h(t + h) &= y(t + h) - y_h(t) - h\,\Phi(t, y_h(t), h) \\ &= y(t + h) - y(t) - h\,\Phi(t, y(t), h) \\ &= h\,d(y(\cdot), t + h, h). \end{aligned}$$

[4] In Sheet 1, Exercise 2 werden die Konsistenzordnungen diese Verfahren nachgerechnet. In Sheet 1, Homework 1 werden diese implementiert.

[5] In Sheet 1, Exercise 3 wird die Fixpunkt-Iteration angegeben und Konvergenz für kleine h gezeigt. Darüber hinaus wird auch die Newton-Iteration aufgestellt und die Lösbarkeit des Newton-Systems überprüft.

Das heißt: Wenn man bei t mit dem exakten Startwert $y_h(t) = y(t)$ startet, so ist der Fehler nach einem einzelnen Integrationsschritt der Länge h gerade $h\, d(y(\cdot), t+h, h)$.

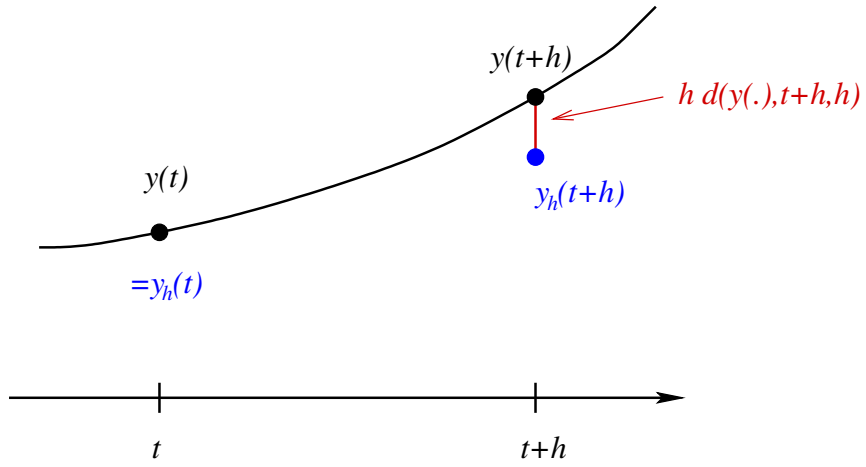


Abbildung 3.4. Veranschaulichung lokaler Diskretisierungsfehler.

Beispiel 3.13 (Konsistenzordnung des Heun-Verfahrens)**.** Wir wollen die Konsistenzordnung des in (3.16) beschriebenen Verfahrens herleiten.
Allgemeine Vorgehensweise: Wir wenden den Satz von Taylor 3.2 auf y und Φ in den Entwicklungspunkten t und $y = y(t)$ an. Diese setzen wir anschließend in (3.12) ein. Um die Notation zu vereinfachen schreiben wir $y := y(t)$, $f := f(t, y(t))$, sowie $f_{,t} := \frac{\partial}{\partial t} f(t, y)$, $f_{,y} := \frac{\partial}{\partial y} f(t, y), \ldots$.

- Eine Taylorentwicklung im Entwicklungspunkt t liefert
$$\frac{1}{h}(y(t+h) - y(t)) = y' + \frac{1}{2}y''h + \frac{1}{6}y'''h^2 + \mathcal{O}(h^3)$$
Ferner gilt:
$$y' = f(t, y(t)) = f$$
$$y'' = \frac{d}{dt} f(t, y(t)) = f_{,t} + f_{,y}\, f$$
$$y''' = \frac{d}{dt}(f_{,t} + f_{,y}\, f) = f_{,tt} + f_{,ty} f + (f_{,ty} + f_{,yy}\, f)\, f + f_{,y}(f_{,t} + f_{,y}\, f)$$
$$= f_{,tt} + 2f_{,ty}\, f + f_{,yy}\, f^2 + f_{,y}\, f_{,t} + f_{,y}^2\, f$$

- Wir verwenden die Taylorformel
$$\begin{aligned} &f(t + \Delta t, y + \Delta y) \\ &= f + f_{,t}\Delta t + f_{,y}\Delta y \\ &+ \frac{1}{2}\left(f_{,tt}\Delta t^2 + 2f_{,ty}\Delta t\, \Delta y + f_{,yy}\Delta y^2\right) + \mathcal{O}((\Delta t + \Delta y)^3). \end{aligned}$$
Mit $\Delta t = h$ und $\Delta y = hf$ folgt dann
$$\Phi = f + \frac{1}{2}h(f_{,t} + f_{,y}\, f) + \frac{1}{4}\left(f_{,tt}h^2 + 2f_{,ty}h^2 f + f_{,yy}h^2 f^2\right) + \mathcal{O}(h^3).$$

- Einsetzen in (3.12) liefert

$$\begin{aligned} d(t,h,y) &= h^2\Big(\left(\frac{1}{6}-\frac{1}{4}\right) f_{,tt} + \left(\frac{1}{3}-\frac{1}{2}\right) f_{,ty}\, f \\ &\quad + \left(\frac{1}{6}-\frac{1}{4}\right) f_{,yy}\, f^2 + \frac{1}{6} f_{,y}\, f_{,t} + \frac{1}{6} f_{,y}^2\, f \Big) + \mathcal{O}(h^3) \\ &= \mathcal{O}(h^2). \end{aligned}$$

  Das Heun-Verfahren ist also konsistent mit der Ordnung 2. Wir haben hier sogar gezeigt, dass es maximal die Ordnung 2 besitzt.

Definition 3.14 (Fehler, Konvergenz, Konvergenzordnung bei ESV)**.** Es sei $y(\cdot)$ die Lösung des **(AWP)** auf dem Intervall $[0,T]$. Weiter sei $y_h(\cdot)$ eine mit einem ESV (3.11) erzeugte Näherungslösung auf einem Gitter $\mathcal{T} = \{0 = t_0 < t_1 < \ldots < t_N = T\}$.

(a) Die Größe
$$e_h(t) := y_h(t) - y(t), \quad t \in \mathcal{T}$$
heißt der **globale (Diskretisierungs-)Fehler** des Verfahrens auf dem Gitter $\mathcal{T}$ an der Stelle $t \in \mathcal{T}$.

(b) Das ESV heißt **konvergent**, falls auf jeder Folge von Gittern $\{\mathcal{T}_h\}$ mit $h \searrow 0$ für die erzeugten Näherungslösungen gilt:
$$\|y_h - y\|_{\infty,h} := \max_{t\in\mathcal{T}_h} \|y_h(t) - y(t)\| \to 0 \quad \text{für } h \searrow 0.$$

(c) Das ESV hat die **Konvergenzordnung** $p \in \mathbb{N}$, falls eine von h unabhängige Konstante $c > 0$ sowie $\overline{h} > 0$ existieren, sodass für die Näherungslösung auf einem beliebigen Gitter der maximalen Gitterweite $h \le \overline{h}$ gilt:
$$\|y_h - y\|_{\infty,h} \le c\, h^p.$$

Beim expliziten Euler-Verfahren hatten wir mit Hilfe einer Stabilitätsabschätzung (Lemma 3.4) nachgewiesen, dass aus der Konsistenz des Schemas die Konvergenz mit derselben Ordnung folgt. Dies wiederholen wir jetzt für ein allgemeines ESV.

Lemma 3.15 (Diskrete Stabilität von ESV)**.** Die Verfahrensfunktion des ESV (3.11) erfülle die **Lipschitz-Bedingung** (**Stabilitätsbedingung**)
$$\|\Phi(t,h,v) - \Phi(t,h,w)\| \le K\, \|v - w\| \tag{3.17}$$
für alle $t \in [0,T]$, $y_1, y_2 \in \mathbb{R}^n$ sowie $h \le \overline{h}$. Es sei $\mathcal{T} = \{0 = t_0, t_1, \ldots, t_N = T\}$ ein Gitter auf $[0,T]$ mit maximaler Schrittweite $h \le \overline{h}$ und v_h und w_h zwei *beliebige* Gitterfunktionen auf $\mathcal{T}$. Dann gilt die **diskrete Stabilitätsabschätzung**
$$\|v_n - w_n\| \le \left(\|v_0 - w_0\| + \sum_{j=1}^{n} h_{j-1}\, \|d_j^v - d_j^w\| \right) \exp\left(K\, t_n\right) \tag{3.18}$$
für alle $n = 0, 1, \ldots, N$. Dabei steht d_j^v als Abkürzung für den Konsistenzfehler (3.12) *zur Gitterfunktion* w_h
$$d_j^v := d(v_h, t_j, h_{j-1}) = \frac{1}{h_j}(v_j - v_{j-1}) - \Phi(t_{j-1}, h_{j-1}, v_{j-1}), \quad j = 1, \ldots, N$$

und analog für d_j^w.

Beweis: Nach Definition von d_j^v und d_j^w erfüllen die Gitterfunktionen die Rekursion

$$v_{n+1} = v_n + h_n\,\Phi(t_n, h_n, v_n) + h_n\,d_{n+1}^v$$
$$w_{n+1} = w_n + h_n\,\Phi(t_n, h_n, w_h) + h_n\,d_{n+1}^w.$$

Die Bedingung (3.17) erlaubt also die Abschätzung

$$\|v_{n+1} - w_{n+1}\| \le \|v_n - w_n\| + h_n\,K\,\|v_n - w_n\| + h_n\,\|d_{n+1}^v - d_{n+1}^w\|.$$

Von hier aus erhalten wir wie in (3.6) durch wiederholtes Einsetzen in den ersten Term die Abschätzung

$$\|v_n - w_n\| \le \|v_0 - w_0\| + \sum_{j=0}^{n-1} h_j\,K\,\|v_j - w_j\| + \sum_{j=1}^{n} h_{j-1}\,\|d_j^v - d_j^w\|$$

und weiter mit dem diskreten Lemma von Gronwall (Lemma 3.4) mit den Setzungen

$$a_n := \|v_n - w_n\|, \quad b_n := \|v_0 - w_0\| + \sum_{j=1}^{n} h_{j-1}\,\|d_j^v - d_j^w\|, \quad \delta_j := h_j\,K$$

die Behauptung. □

Bemerkung 3.16 (zu Lemma 3.15)**.**

Beim expliziten Euler-Verfahren ($\Phi(t,y,h) = f(t,y)$) gilt $K = L$ (L ist die Lipschitz-Konstante von f) für alle $h > 0$. Auch bei allen anderen in dieser Vorlesung behandelten ESV ist die Bedingung (3.17) erfüllt.

Bei impliziten ESV gilt dies nur unter der Zusatzbedingung $h \le \overline{h}$. Die stellt aber für die Konvergenzbetrachtung keine Einschränkung dar. Im Allgemeinen hängt K nur von der Lipschitz-Konstanten der rechten Seite f ab.[6]

Satz 3.17 (Konvergenz von ESV)**.** Sei Φ die Verfahrensfunktion eines ESV der Form (3.11), welches die Lipschitz-Bedingung (3.17) für alle $h \le \overline{h}$ erfüllt. Ferner sei das zugehörige ESV konsistent mit Ordnung $p \in \mathbb{N}$. Dann gilt für die Näherungslösungen auf einem beliebigen Gitter $\{0 = t_0, t_1, \ldots, t_N = T\}$ der Klasse $\mathcal{T}_h$ mit maximaler Schrittweite $h \le \overline{h}$ die Abschätzung

$$\|y_h(t_n) - y(t_n)\| \le C(y)\mathrm{e}^{K\,t_n}t_n\,h^p, \qquad n = 0, 1, \ldots, N, \tag{3.19}$$

vorausgesetzt die Lösung y von (**AWP**) gehört zu $C^{p+1}([0,T];\mathbb{R}^n)$.

Wir Verwenden auch die Kurzschreibweise:

$$\|y - y_h\|_{\infty,h} \le C(y,f,T)h^p.$$

Beweis: Wir gehen vor wie im Beweis von Theorem 3.5. Wir verwenden Lemma 3.15 (diskrete Stabilität) mit

$$v_h = y_h, \qquad w_h = y|_\tau$$

[6] In Sheet 2, Exercise 4 wird die Lipschitz-Stetigkeit für das implizte Euler-Verfahren und das Verfahren von Heun nachgerechnet.

Dann gilt nach Voraussetzung für die jeweiligen lokalen Fehler

$$\begin{aligned} d_j^v &= 0 \\ \|d_j^w\| &= \|d(y(\cdot), t_j, h_{j-1})\| \leq C\, h^p. \end{aligned}$$

Aus Lemma 3.15 ergibt sich nun

$$\|y_h(t_n) - y(t_n)\| \leq \left(\|y_h(0) - y(0)\| + C(y)\, h^p \sum_{j=1}^{n} h_{j-1} \right) \exp\left(K\, t_n\right).$$

Mit $y_h(0) = y(0)$ und $\sum_{j=1}^{n} h_{j-1} = t_n$ folgt die Behauptung. □

Folgerung 3.18. Die Konvergenzordnung eines ESV, das die Bedingung (3.17) erfüllt, entspricht also der Konsistenzordnung. Man spricht deshalb einfach von *der* **Ordnung** eines ESV.

Bemerkung 3.19 (zur Fehlerabschätzung (3.19))**.**

(a) Der globale Fehler kann exponentiell mit der Zeit t anwachsen. Dieses Verhalten entspricht dem der Differenz zweier Lösungen zu verschiedenen Anfangsdaten, siehe Satz 2.1.

(b) Die Konvergenzordnung bleibt erhalten, wenn im Verfahren ungenaue AW verwendet werden, die aber mit der „richtigen“ h-Ordnung gegen $y(0)$ konvergieren, also $\|y_h(0) - y(0)\| = \mathcal{O}(h^p)$.

(c) (3.19) ist eine **a-priori Abschätzung**, d. h. wir können vor dem Berechnen der y_i und ohne Kenntniss der exakten Lösung y eine Aussage über den asymptotischen Verlauf ($h \to 0$) des Verfahrensfehlers treffen.

BSP: Bei einem Verfahren 2. Ordnung wird beim halbieren der Schrittweite ($h \to h/2$) der Fehler geviertelt $\|y - y_{h/2}\| \leq \frac{1}{4}\|y - y_h\| + o(h^2)$.

(d) Die Konstante C des lokalen Fehlers (die $\max_{t\in[0,T]}\|y^{(p+1)}(t)\|$ enthält) und die Lipschitz-Konstante K sind im Allgemeinen nicht bekannt. Daher ist (3.19) nicht geeignet um eine Schrittweitensteuerung zu realisieren. Hierfür sind *a-posteriori* Schranken besser geeignet.

Um diesen Abschnitt abzuschließen diskutieren wir noch die in Satz 3.17 auftretende Voraussetzung $y \in C^{p+1}([0,T];\mathbb{R}^n)$:

Satz 3.20 (Maximale Regularität)**.** Es sei $y : [0,T] \to Q \subset \mathbb{R}^n$ eine Lösung des **(AWP)** auf $[0,T]$. Falls $f \in C^k([0,T] \times Q)$ ist für ein $k \in \mathbb{N}_0$, dann gilt $y \in C^{k+1}([0,T];\mathbb{R}^n)$.

Beweis: Wir verwenden *Bootstrapping*-Argumente:

$$\begin{aligned} k = 0: \quad & y'(t) = f(t, y(t)) && \Rightarrow \quad y \in C^1([0,T];\mathbb{R}^n) \\ k = 1: \quad & y''(t) = f_{,t}(t, y(t)) + f_{,y}(t, y(t))\, y'(t) && \Rightarrow \quad y \in C^2([0,T];\mathbb{R}^n) \end{aligned}$$

usw. durch Induktion. □

Ist y also nur von geringer Glattheit (z.B. wenn f geringe Glattheit aufweist), so bringt die Verwendung eines Verfahrens höherer Ordnung keine Verbesserung der Konvergenzrate, da diese durch die Regularität der Lösung limitiert ist. [7]

In diesen Fällen lässt sich die Konvergenzrate aber durch eine adaptive Schrittweitensteuerung verbessern.

§ 3.3 Runge-Kutta-Verfahren

Wir haben bisher einige Beispiele expliziter und impliziter Verfahren der Ordnungen 1 und 2 kennengelernt. In diesem Abschnitt beschäftigen wir uns mit der systematischen Konstruktion von ESV höherer Ordnung $p \geq 2$. Dabei beschränken wir uns der Einfachheit halber auf äquidistante Schrittweiten, was aber eigentlich nicht notwendig ist. Zum Einstieg betrachten wir folgenden Ansatz für eine explizite[8] Verfahrensfunktion (vgl. Beispiel 3.11):

$$\Phi(t, y, h) = \gamma_1 f(t, y) + \gamma_2 f(t + \alpha h, y + \beta h f(t, y)) \tag{3.20}$$

mit vier Parametern α, β, γ_1 und γ_2 (vergleiche Beispiel 3.11 d) und e).

Ziel: Wähle α, β, γ_1, γ_2 so, dass das Verfahren eine möglichst hohe Konsistenzordnung besitzt.

Vorgehensweise: Analog zu Beispiel 3.13 ermitteln wir die Ordnung des Konsistenzfehlers

$$d(y, t + h, h) := \frac{y(t + h) - y(t)}{h} - \Phi(t, h, y)$$

über Taylor-Entwicklungen von Φ und $y(t + h)$:

- Für den Differenzenquotienten gilt:

$$\begin{aligned}\frac{1}{h}(y(t + h) - y(t)) &= y' + \frac{1}{2} h\, y'' + \mathcal{O}(h^2) \\ &= f + \frac{1}{2} h\, (f_{,t} + f_{,y} f) + \mathcal{O}(h^2).\end{aligned}$$

- Für die Verfahrensfunktion gilt:

$$\begin{aligned}\Phi(t, h, y(t)) &= \gamma_1 f + \gamma_2 f(t + \alpha h, y + \beta h f) \\ &= \gamma_1 f + \gamma_2\, (f + f_{,t}\, \alpha h + f_{,y}\, \beta h f) + \mathcal{O}(h^2) \\ &= (\gamma_1 + \gamma_2) f + \gamma_2 h\, (\alpha f_{,t} + \beta f_{,y}\, f) + \mathcal{O}(h^2).\end{aligned}$$

- Einsetzen in $d(y, t + h, h)$ ergibt:

$$\begin{aligned}d(y, t + h, h) = (1 - \gamma_1 - \gamma_2) f \\ + \left(\left(\frac{1}{2} - \gamma_2 \alpha \right) f_{,t} + \left(\frac{1}{2} - \gamma_2 \beta \right) f_{,y}\, f \right) h \\ + \mathcal{O}(h^2).\end{aligned}$$

Nach Koeffizientenvergleich erhalten wir:

[7] In Sheet 2, Homework 4 werden das explizite und verbesserte Eulerverfahren auf ein Problem angewendet, dessen exakte Lösung einen Knick hat. Dort sieht man auch, dass man im Fall, dass der Knick auf einem Gitterpunkt liegt, Glück hat.

[8] beachte: kein Lösen von Gleichungssystemen, nur 2 f-Auswertungen

- Ein Verfahren der Ordnung $p = 1$ unter der Bedingung:
$$\gamma_1 + \gamma_2 = 1. \tag{3.21}$$
- Ein Verfahren der Ordnung $p = 2$, falls zusätzlich gilt:
$$\alpha\gamma_2 = 1/2, \qquad \beta\gamma_2 = 1/2. \tag{3.22}$$
Diese Bedingung impliziert $\alpha = \beta$. Wir haben somit eine ganze Verfahrensklasse hergeleitet (3 Unbekannte $(\gamma_1, \gamma_2, \alpha)$ für 2 Bedingungen).
- Die Ordnung 3 ist mit dem expliziten Ansatz (3.20) nicht erreichbar.[9]

Beispiel 3.21 (vgl. Beispiel 3.11)**.** Ansatz (3.20) enthält die folgenden schon bekannten expliziten Verfahren:

(a) Mit $\gamma_1 = 1$ und $\gamma_2 = 0$ ergibt sich das explizite Euler-Verfahren (Ordnung 1).

(b) Mit $\gamma_1 = 0$, $\gamma_2 = 1$ und $\alpha = \beta = 1/2$ ergibt sich das verbesserte Euler-Verfahren (Ordnung 2).

(c) Mit $\gamma_1 = \gamma_2 = 1/2$ und $\alpha = \beta = 1$ ergibt sich das Verfahren von Heun (Ordnung 2).

Die Idee (3.20) lässt sich verallgemeinern:

Definition 3.22 (Allgemeines r-stufiges Runge-Kutta-Verfahren)**.** Ein Verfahren, welches sich aus der Funktion
$$\Phi(t_k, h_k, y_k) = \sum_{j=1}^{r} \gamma_j f_{jk}$$
mit
$$f_{jk} = f\left(t_k + \alpha_j h_k, y_k + h_k \sum_{i=1}^{r} \beta_{ji}\, f_{ik}\right), \quad j = 1, \ldots, r,$$
heißt **r-stufiges Runge-Kutta-Verfahren**. Dabei sind γ_j, α_j, β_{ji} reelle Parameter. Das Verfahren heißt

- **explizit**, falls $\beta_{ji} = 0$ für $i \geq j - 1$,
- und andernfalls **implizit**.

Bemerkung 3.23 (zu Definition 3.22)**.** Bei expliziten Verfahren geht die Summe zur Berechnung der f_{jk} bis $j - 1$. Dann kann man nacheinander $f_{1k}, f_{2k}, \ldots, f_{rk}$ explizit berechnen. Die Realisierung impliziter Verfahren ist deutlich komplizierter.

Ein RKV ist durch die Angabe der Vektoren
$$a = (\alpha_1, \ldots, \alpha_s)^\top, \qquad c = (\gamma_1, \ldots, \gamma_s)^\top$$
und der Matrix
$$B = (\beta_{j\ell})_{j,\ell=1,\ldots,s}$$

[9] Wenn man die Taylor-Entwicklung eine Ordnung weiter ausführt, sieht man, dass beim Koeffizientenvergleich Bedingungen entstehen, die mit den vier Koeffizienten nicht alle zu erfüllen sind.

eindeutig bestimmt. Man notiert es kurz als **Butcher-Diagramm** bzw. **Butcher-Tableau**:

$$\begin{array}{c|c} a & B \\ \hline & c^\top \end{array} \quad \text{bzw.} \quad \begin{array}{c|ccc} \alpha_1 & \beta_{11} & \cdots & \beta_{1r} \\ \vdots & \vdots & & \vdots \\ \alpha_r & \beta_{r1} & \cdots & \beta_{rr} \\ \hline & \gamma_1 & \cdots & \gamma_r \end{array}$$

Zur Herleitung spezieller Runge-Kutta-Verfahren verwenden wir den Hauptsatz der Differential- und Integralrechnung und erhalten

$$y(t_{k+1}) - y(t_k) = \int_{t_k}^{t_{k+1}} y'(t)\mathrm{d}t = \int_{t_k}^{t_{k+1}} f(t, y(t))\mathrm{d}t.$$

Verwendet man die Approximation $y_k \approx y(t_k)$ und eine Quadraturformel (z.B. die implizite Trapezregel) für das Integral über f erhält man

$$y_{k+1} = y_k + h_k \frac{1}{2} \left(\underbrace{f(t_k, y_k)}_{f_{1k}} + \underbrace{f(t_{k+1}, y_{k+1})}_{=f_{2k}} \right). \tag{*}$$

Die Steigungen ergeben sich aus

$$\begin{aligned} f_{1k} &= f(t_k + 0\, h_k, y_k) \\ f_{2k} &= f(t_{k+1}, y_{k+1}) \overset{(*)}{=} f\left(t_k + 1\, h_k, y_k + h_k \left[\frac{1}{2} f_{1k} + \frac{1}{2} f_{2k} \right] \right). \end{aligned}$$

Ablesen der Koeffizienten liefert das Butcher-Tableau:

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

Durch die Wahl anderer Quadraturformeln lassen sich viele weitere Verfahren konstruieren.

Beispiel 3.24 (Einige ESV und ihre Butcher-Diagramme)**.** Alle in Beispiel 3.11 vorgestellten Verfahren sind RKV:

(a) explizites Euler-Verfahren (**Vorwärts-Euler-Verfahren**)

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

(b) implizites Euler-Verfahren (**Rückwärts-Euler-Verfahren**)

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

(c) verbessertes Euler-Verfahren (**explizite Mittelpunktsregel**)

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

(d) Verfahren von Heun

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

Dazu kommen die weiteren Verfahren:

(e) **implizite Mittelpunktsregel**

$$\begin{array}{c|c} 1/2 & 1/2 \\ \hline & 1 \end{array}$$

(f) **implizite Trapezregel**

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

(g) **das Verfahren aus** (3.20)

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

Es ist zu erkennen, dass bei expliziten Verfahren alle Einträge in B auf und oberhalb der Hauptdiagonalen verschwinden. Bei semi-impliziten Verfahren ist die Hauptdiagonale besetzt.

§ 3.3.1 Allgemeine Ordnungsbedingungen

Wir diskutieren nun Bedingungen, die die Konsistenzordnung und damit die Konvergenzordnung eines RKVs festlegen.

Definition 3.25. Eine Differentialgleichung heißt **autonom**, falls diese nicht explizit von der unabhängigen Variablen t abhängt. Eine explizite autonome DGL 1. Ordnung hat also die Gestalt

$$\dot{y}(t) = f(y(t)), \qquad t \in [0, T]$$

mit $f\colon \mathbb{R}^n \to \mathbb{R}^n$.

Eine nicht-autonome Differentialgleichung kann man durch Hinzunahme einer weiteren Unbekannten y_{n+1} mit

$$\dot{y}_{n+1} = 1, \qquad y_{n+1} = 0$$

autonom machen. Diese Bedingung ist äquivalent zu $y_{n+1} = t$. Das daraus resultierende System mit Lösung $\tilde{y}(t) = (y(t), y_{n+1}(t))^\top$ lautet

$$\dot{\tilde{y}} = \begin{pmatrix} \dot{y} \\ \dot{y}_{n+1} \end{pmatrix} = \tilde{f}(\tilde{y}), \quad \tilde{f}(\tilde{y}) = \begin{pmatrix} f(y_{n+1}, y) \\ 1 \end{pmatrix} \tag{3.23}$$

mit Anfangswert $\tilde{y}(0) = (y_0, 0)^\top$. Für ein sinnvolles Runge-Kutta-Verfahren fordern wir, dass dieses **autonomie-invariant** ist, d. h. angewendet auf (**AWP**) und (3.23) die selbe Lösung liefert.

Satz 3.26 (Autonomie-Invarianz allgemeiner RKV)**.** Ein RKV liefert dieselben Näherungswerte für die autonome und nicht-autonome Form jeder Dgl, wenn gilt:

$$\alpha_j = \sum_{i=1}^{r} \beta_{ji}, \quad j = 1, \dots, r\,, \tag{3.24a}$$

$$1 = \sum_{j=1}^{r} \gamma_j\,. \tag{3.24b}$$

Beweis: Wir betrachten einen beliebigen Zeitschritt $t_k \rightsquigarrow t_{k+1} := t_k + h_k$, und es gelte:

$$\tilde{y}_k = (y_k, t_k), \quad \text{d. h.} \quad [y_k]_{n+1} = t_k.$$

Vergleicht man jeweils einen RKV-Schritt für das autonome und nicht-autonome System erhält man

$$\tilde{y}_{k+1} = \tilde{y}_k + h_k \sum_{j=1}^{r} \gamma_j \tilde{f}_{jk} \qquad \text{(n+1 Komponenten)}$$

$$y_{k+1} = y_k + h_k \sum_{j=1}^{r} \gamma_j f_{jk} \qquad \text{(n Komponenten).}$$

Aus der Forderung $[\tilde{y}_{k+1}]_{1,\dots,n} = y_{k+1}$ folgt

$$\begin{aligned} f_{jk} &= f\left(t_k + \alpha_j h_k, y_k + h_k \sum_{i=1}^{r} \beta_{ji} f_{ik}\right) \\ \overset{!}{=} [\tilde{f}_{jk}]_{1,\dots,n} &= f\left(\underbrace{[y_k]_{n+1}}_{=t_k} + h_k \sum_{i=1}^{r} \beta_{ji} \underbrace{[\tilde{f}_{ik}]_{n+1}}_{=1}, y_k + h_k \sum_{i=1}^{r} \beta_{ji} [\tilde{f}_{ik}]_{1,\dots,n}\right). \end{aligned}$$

Dies führt auf die Bedingung

$$\alpha_j = \sum_{i=1}^{r} \beta_{ji}, \qquad j = 1, \dots, r,$$

so dass $f_{ij} = [\tilde{f}_{ij}]_{1,\dots,n}$ erfüllt ist. Die zweite Bedingung bekommen wir aus der Forderung $[y_{k+1}]_{n+1} = t_{k+1} = t_k + h_k$. Diese führt auf die Gleichung

$$t_k + h_k = [y_k]_{n+1} + h_k \sum_{j=1}^{r} \gamma_j \underbrace{[\tilde{f}_{jk}]_{n+1}}_{=1}$$

und wegen $[y_k]_{n+1} = t_k$ erhalten wir (3.24b). □

Beachte: (3.24a) bedeutet, dass die Zeilensummen der Matrix B gerade den Koeffizienten von α entsprechen. Wenn man auf die Autonomie-Invarianz verzichtet, so ist (3.24b) gerade die Bedingung für die Konsistenzordnung 1.

Wir geben nun die Ordnungsbedingungen von RKV bis zur Ordnung $p = 4$ an.

Satz 3.27 (Ordnungsbedingungen allgemeiner RKV[10])**.** Wir betrachten ein RKV, das die Bedingung (3.24) erfüllt.

(a) Ein solches RKV besitzt mindestens die Konsistenzordnung 1.

(b) Es besitzt Ordnung 2, falls zusätzlich gilt:

$$\sum_{j=1}^{r} \gamma_j \alpha_j = \frac{1}{2}.$$

(c) Es besitzt Ordnung 3, falls zusätzlich gilt:

$$\sum_{j,i=1}^{r} \gamma_j \alpha_i \beta_{ji} = \frac{1}{6} \quad \text{und} \quad \sum_{j=1}^{r} \gamma_j \alpha_j^2 = \frac{1}{3}.$$

(d) Es besitzt Ordnung 4, falls zusätzlich gilt:

$$\sum_{j=1}^{r} \gamma_j \, \alpha_j^3 = \frac{1}{4}, \qquad \sum_{j,i=1}^{r} \gamma_j \, \alpha_j \, \beta_{ji} \, \alpha_i = \frac{1}{8},$$

$$\sum_{j,i=1}^{r} \gamma_j \, \beta_{ji} \, \alpha_i^2 = \frac{1}{12}, \qquad \sum_{i,j,k=1}^{r} \gamma_k \, \beta_{kj} \, \beta_{ji} \, \alpha_i = \frac{1}{24}.$$

Beweis: Wir gehen vor, wie am Anfang von § 3.3 beschrieben: In der Verfahrensfunktion

$$\Phi(t_k, h_k, y_h) = \sum_{j=1}^{r} \gamma_j \, f_{kj}$$

werden die Terme f_{kj} an der Stelle $(t, y(t))$ Taylor-entwickelt. Dies wird dann in die Gleichung für den Konsistenzfehler eingesetzt.

Idee: Wir können uns dabei auf autonome Dgl $y'(t) = f(y(t))$ beschränken. Dadurch erhalten wir zunächst nur Bedingungen an die Koeffizienten $\beta_{j\ell}$ und γ_j des Verfahrens. Aus (3.24a) können wir dann die Koeffizienten α_j, die für die nichtautonome Version benötigt werden, bestimmen. Mit diesem Trick sparen wir uns bei der ohnehin unübersichtlichen Taylor-Entwicklung die Ableitungen von f, in denen t vorkommt. □

Bemerkung 3.28 (zu RKV)**.**

(a) Die Ordnungsbedingungen sind nichtlineare algebraische Gleichungen in den Koeffizienten des Verfahrens. Das Aufstellen der Bedingungen für Verfahren höherer Ordnung ist sehr aufwändig.

(b) Das Computer-Algebra-System "Mathematica" besitzt eine Routine zum Aufstellen der Ordnungsbedingungen für RKV:

```
<< NumericalDifferentialEquationAnalysis`
RungeKuttaOrderConditions[4]
```

Das Argument der Funktion ist die angestrebte Ordnung des Verfahrens.

(c) Bei der Anwendung auf die triviale Differentialgleichung $y'(t) = f(t)$ reduziert sich *jedes* (explizite oder implizite) RKV auf eine Quadraturformel[11]

$$y_{k+1} = y_k + h_k \sum_{j=1}^{r} \gamma_j \, f(t_k + \alpha_j \, h_k)$$

mit k Stützstellen und Gewichten zur näherungsweisen Berechnung des Integrals $\int_{t_k}^{t_k+h_k} f(s)\,\mathrm{d}s$. Besitzt das RKV die Konsistenzordnung $p \in \mathbb{N}$, so ist der **Genauigkeitsgrad** des Quadraturverfahrens $p-1$, d. h. Polynome $t \mapsto f(t)$ bis zum Grad $p-1$ werden exakt integriert. [12]

[10] In Sheet 2, Homework 3 wird für das implizite Euler-Verfahren, verbesserte Euler-Verfahren und das Verfahren von Heun anhand der hier gegeben Bedinungen die jeweilige Ordnung der Verfahren überprüft.

§ 3.3.2 Explizite Runge-Kutta-Verfahren

Im expliziten Fall lassen sich die Steigungen f_{jk}, $j = 1, \dots, r$ sukzessive und ohne die Lösung von Gleichungssystemen berechnen. Explizite RKV sind also explizite ESV im Sinne unserer früheren Definition 3.8. Jedes f_{jk} benötigt lediglich die Kenntnis der f_{ik}, $i < j$, und eine f-Auswertung. Der Aufwand pro Zeitschritt ist also etwa proportional zur Stufenzahl r. Damit bieten ERKV zunächst wesentliche algorithmische Vorteile gegenüber impliziten RKV.

Es ergibt sich folgender Algorithmus für ein ERKV [13] zu den Schrittweiten h_k:

Algorithmus 3.29 (Durchführung eines ERKV)**.**
Eingabe: Anfangswert y_0, Schrittweiten $\{h_k\}$, Koeffizienten eines ERKV $a = (\alpha_j)$, $B = (\beta_{ji})$, $c = (\gamma_j)$
Ausgabe: Näherungslösung $y_h(\cdot)$ des (**AWP**) als $y_0, \dots, y_k = y_h(t_k), \dots, y_N$

1: **for** $k = 0, 1, 2, \dots, (N-1)$ **do**
2: **for** $j = 1, 2, \dots, r$ **do**
3: Berechne die Steigung f_{jk} gemäß

$$f_{jk} = f(t_k + \alpha_j\, h_k, y_k + h_k \sum_{i=1}^{j-1} \beta_{ji}\, f_{ik})$$

4: **end for**
5: Setze

$$y_{k+1} = y_k + h_k \sum_{j=1}^{r} \gamma_j\, f_{jk}$$

6: Setze $t_{k+1} := t_k + h_k$
7: **end for**

Die Ordnungsbedingungen ergeben sich aus Satz 3.27 unter Beachtung der strikten unteren Dreiecksstruktur von B. Wir betrachten als Beispiel das 3-stufige explizite RKV mit dem Butcher-Diagramm

$$\begin{array}{c|ccc} \alpha_1 & & & \\ \alpha_2 & \beta_{21} & & \\ \alpha_3 & \beta_{31} & \beta_{32} & \\ \hline & \gamma_1 & \gamma_2 & \gamma_3 \end{array}$$

mit 9 unbekannten Parametern. Wir fordern die Autonomie-Invarianz (3.24). Dann hat das Verfahren bereits mindestens die Ordnung 1 und $\alpha_1 = 0$. Es besitzt Ordnung 2, falls zusätzlich

$$\gamma_2\, \alpha_2 + \gamma_3\, \alpha_3 = \frac{1}{2} \tag{3.25}$$

gilt, und Ordnung 3, falls zusätzlich noch

$$\gamma_2\, \alpha_2^2 + \gamma_3\, \alpha_3^2 = \frac{1}{3} \qquad \text{und} \qquad \gamma_3\, \alpha_2\, \beta_{32} = \frac{1}{6} \tag{3.26}$$

[11] **Beachte:** Es kommen dann nur die α- und γ-Koeffizienten vor.
[12] Sheet 2, Exercise 6; siehe auch Hanke-Bourgeois, 2006, Satz 76.4
[13] Dieser wird in Sheet 4, Homework 5 implementiert und die Konvergenzordnung des expliziten Euler-Verfahren, verbesserten Euler-Verfahren und dem Verfahren von Heun am Beispiel verifiziert.

gelten.

Man kann zeigen, dass die Ordnung 4 mit einem expliziten 3-stufigen RKV nicht erreicht werden kann, da die Taylor-Entwicklung des lokalen Fehlers im Koeffizienten von h^4 einen Term enthält, der von allen Parametern unabhängig ist.

Unser 3-stufiges explizites Verfahren wird durch 9 Parameter definiert, die folgende 7 Bedingungen erfüllen sollen:

„Autonomie“: 4 Bedingungen (3.24) (eine davon: $\alpha_1 = 0$)
Ordnung 2: 1 Bedingung für Ordnung 2 (3.25)
Ordnung 3: 2 Bedingungen für Ordnung 3 (3.26).

Wenn keine Widersprüche oder redundante Bedingungen darin enthalten sind, können wir eine zwei-parametrige Schar von solchen Verfahren dritter Ordnung erwarten.

Beispiel 3.30 (Einige explizite 3-stufige RKV dritter Ordnung)**.**

(a) Die **Methode von Heun dritter Ordnung** ist gegeben durch

$$\begin{array}{r|lll} 0 & & & \\ 1/3 & 1/3 & & \\ 2/3 & 0 & 2/3 & \\ \hline & 1/4 & 0 & 3/4 \end{array}.$$

(b) Die **Methode von Kutta dritter Ordnung** [14] ist gegeben durch

$$\begin{array}{r|lll} 0 & & & \\ 1/2 & 1/2 & & \\ 1 & -1 & 2 & \\ \hline & 1/6 & 2/3 & 1/6 \end{array}.$$

Der Ansatz für ein explizites 4-stufiges RKV enthält analog zu den Betrachtungen oben 14 Parameter: $\alpha_1, \ldots, \alpha_4$, $\gamma_1, \ldots, \gamma_4$ sowie die sechs Einträge $\beta_{j\ell}$ der strikten unteren Dreiecksmatrix B. Aus den 5 „ Autonomie“-Bedingungen (3.24) und den 7 weiteren Bedingungen aus Satz 3.27 ergibt sich ein System von 12 nichtlinearen Gleichungen als Bedingung für ein Verfahren vierter Ordnung. Es bleibt wiederum eine zweiparametrige Lösungsschar übrig.

Beispiel 3.31 (Einige explizite 4-stufige RKV vierter Ordnung)**.**

(a) *Das* klassische **Verfahren von Runge-Kutta vierter Ordnung** ist gegeben durch

$$\begin{array}{r|llll} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ 1 & 0 & 0 & 1 & \\ \hline & 1/6 & 1/3 & 1/3 & 1/6 \end{array}.$$

[14] Diese Methode entsteht, indem im Integrations-Ansatz zur Herleitung von RKV die Simpsonregel als Quadraturformel verwendet wird.

(b) Das folgende Verfahren entsteht aus der sogenannten **3/8-Regel** als Quadraturformel:

$$\begin{array}{r|cccc} 0 & & & & \\ 1/3 & 1/3 & & & \\ 2/3 & -1/3 & 1 & & \\ 1 & 1 & -1 & 1 & \\ \hline & 1/8 & 3/8 & 3/8 & 1/8 \end{array} .$$

Stufenzahl r	1	2	3	4–5	6	7–8	9–10	11	12–16	17
$2r + \frac{1}{2}r(r-1)$ Parameter	2	5	9	14–20	27	35–44	54–65	77	90–152	170
$r+1$ Bedingungen (3.24)	2	3	4	5–6	7	8–9	10–11	12	13–17	18
max. Ordnung p	1	2	3	4	5	6	7	8	9	10
Ordnungsbedingungen q	0	1	3	7	16	36	84	199	485	1204

Tabelle 3.1. Zusammenhang Stufen, Ordnung, Anzahl Ordnungsbedingungen für explizite RKV

Der Zusammenhang von Ordnung, Parametern und Bedingungen wird in Tabelle 3.1 verdeutlicht. Die Anzahl Parameter eines r-stufigen ERKV ist $2\,r + \frac{1}{2}\,r\,(r-1)$. Es gelten $r+1$ „Autonomie"-Bedingungen (3.24). Die Anzahl übriger Bedingungen für die Ordnung *allgemeiner* RKV ist Hermann, 2004, Tabelle 2.13, S. 36 entnommen. Offenbar sind einige der Ordnungsbedingungen bei ERKV höher Ordnung redundant. Für die angegebene maximale Ordnung sind jeweils auch Verfahren bekannt, siehe z. B. Hairer, Nørsett, Wanner, 1993, Section II.5.

Satz 3.32 (Butcher-Barrieren)**.**

(a) Die Ordnung $p \geq 5$ kann nicht mit einem expliziten RKV der Stufenzahl $r \leq p$ erreicht werden (Butcher, 1965).

(b) Die Ordnung $p \geq 7$ kann nicht mit einem expliziten RKV der Stufenzahl $r \leq p+1$ erreicht werden (Butcher, 1965).

(c) Die Ordnung $p \geq 8$ kann nicht mit einem expliziten RKV der Stufenzahl $r \leq p+2$ erreicht werden (Butcher, 1985).

Beweis: Teil (a) Siehe Hairer, Nørsett, Wanner, 1993, Theorem II.5.1. Für den Rest stehen Literaturangaben in Hairer, Nørsett, Wanner, 1993, Section II.5. □

§ 3.3.3 Implizite Runge-Kutta-Verfahren

Es stellt sich die Frage, wie sich die Betrachtung impliziter RKV (IRKV) angesichts ihres erheblich größeren Aufwandes rechtfertigen lässt. Dazu wenden wir als Beispiel zunächst das explizite Euler-Verfahren auf die sog. **Dahlquist'sche Testgleichung**

$$y'(t) = \lambda\, y, \qquad y(0) = y_a \in \mathbb{R} \tag{3.27}$$

mit $\lambda < 0$ an. Die exakte Lösung $y(t) = y_a\, \mathrm{e}^{\lambda t}$ ist bekannt.[15]

[15] Ein Demo-Programm ist in Matlab/dahlquist_test.m verfügbar.

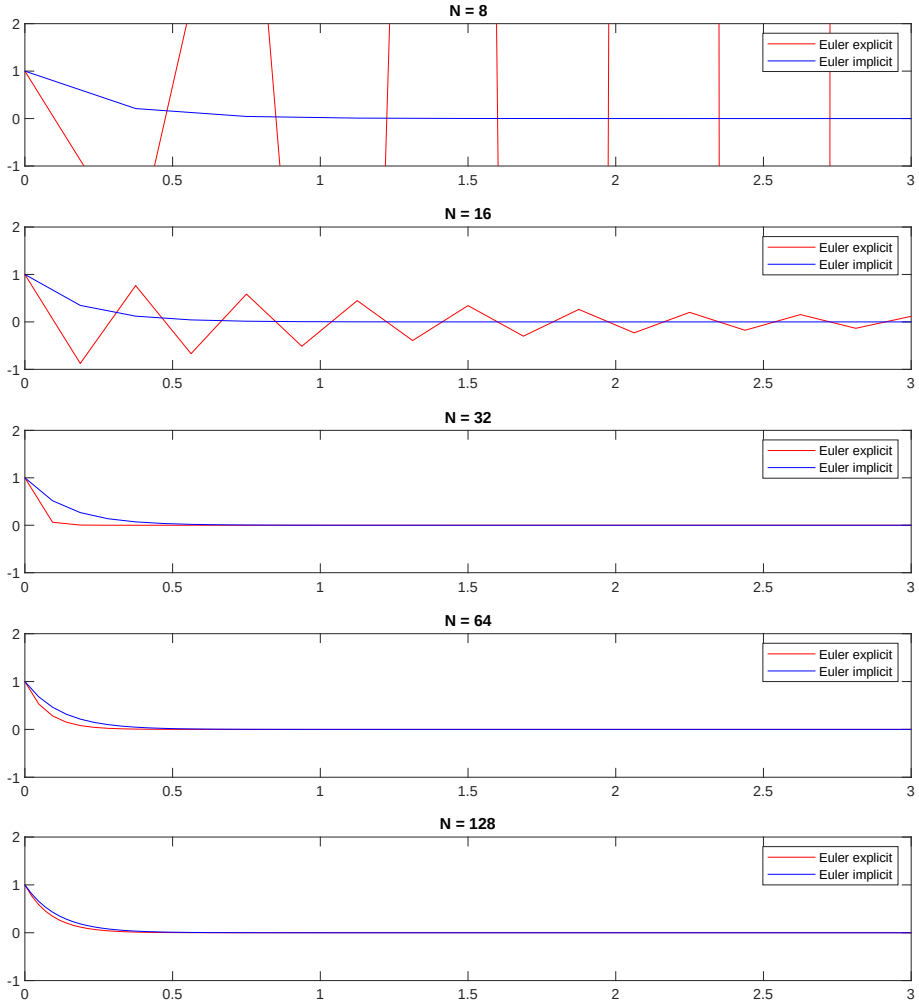


ABBILDUNG 3.5. Vergleich der Lösungen berechnet mit dem expliziten und impliziten Euler-Verfahren zum AWP (3.27) mit $y_a = 1$ und $\lambda = -100$. N ist die Anzahl der Gitterpunkte.

Beobachtung: Die numerische Lösung konvergiert, wie aufgrund von Satz 3.17 zu erwarten war, mit Ordnung 1. Jedoch ist die Lösung für moderate Schrittweiten h praktisch unbrauchbar. Beim impliziten Euler-Verfahren ist die numerische Lösung auch für große Schrittweiten vernünftig (siehe Abbildung 3.5).

Die Näherungslösungen des expliziten Verfahrens erfüllen

$$\begin{aligned} y_{k+1} &= (1 + h_k\,\lambda)\,y_k, \quad k = 0, 1, \ldots, \\ y_0 &= y_a. \end{aligned} \tag{3.28}$$

Damit $y_h(t)$ wie die exakte Lösung $y(t) = y_a\,\mathrm{e}^{\lambda t}$ bei $\lambda < 0$ zumindest vom Betrag her monoton nicht-wachsend ist, muss die Schrittweite h die Bedingung

$$|1 + h\,\lambda| \leq 1$$

erfüllen, also

$$0 \leq h \leq \frac{2}{-\lambda} \quad \text{für } \lambda < 0. \tag{3.29}$$

Etwa im Fall $\lambda = -1000$ ergibt dies die Forderung $h \leq 2 \cdot 10^{-3}$.

Für $\lambda \geq 0$ ergibt sich aus der analogen Forderung $|1 + h\,\lambda| \geq 1$ für betragsmäßiges Wachstum der Näherungslösungen keine Einschränkung an h.

Fazit: Für das AWP (3.27) mit $\lambda \ll 0$ wird die Schrittweite h im expliziten Euler-Verfahren durch die „Stabilitätsanforderung“ (3.29) stark eingeschränkt.

Beim impliziten Euler-Verfahren sind dagegen die Näherungslösungen gegeben durch

$$\begin{aligned} & y_{k+1} = y_k + h_k\, f(t_{k+1}, y_{k+1}) = y_k) + h_k\,\lambda\, y_{k+1}, \\ \Rightarrow \quad & y_{k+1} = \frac{1}{1 - h_k\,\lambda}\, y_k, \quad k = 0, 1, \ldots, \\ & y_0 = y_a. \end{aligned} \tag{3.30}$$

Diese sind betragsmäßig fallend für *jede* Schrittweite $h > 0$, siehe Abbildung 3.6. [16]

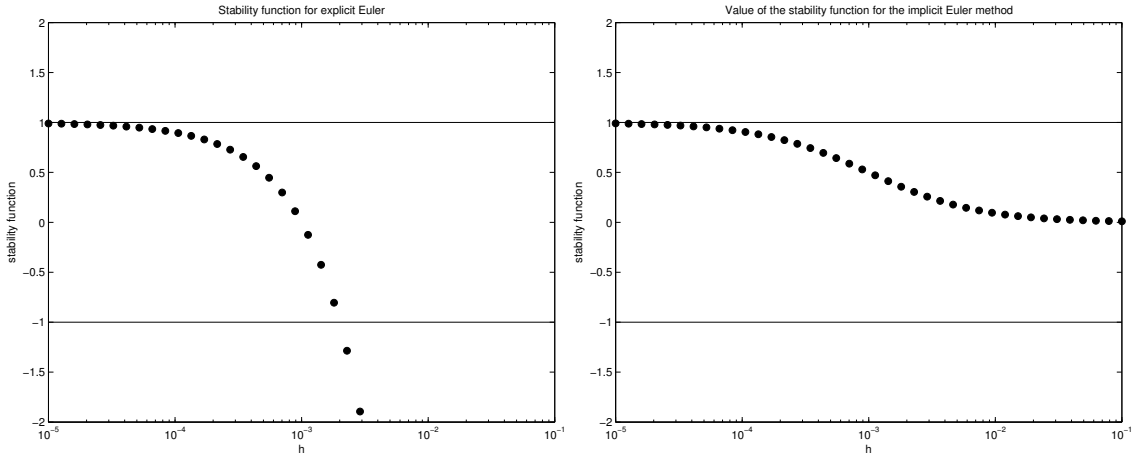


Abbildung 3.6. Darstellung der Funktion $h \mapsto 1 + h\,\lambda$ (links: explizites Euler-Verfahren) und $h \mapsto 1/(1 - h\,\lambda)$ (rechts: implizites Euler-Verfahren) bei $\lambda = -1000$ für verschiedene Schrittweiten h in einer einfach logarithmischen Darstellung.

Die starke Einschränkung der Schrittweitenwahl für explizite RKV bei Gleichungen vom Typ (3.27) mit $\lambda \ll 0$ (fehlende A-Stabilität, siehe § 3.4) motiviert die Betrachtung impliziter RKV trotz ihres höheren Aufwandes.

Falls B keine strikte untere Dreiecksmatrix ist, sind die Steigungen $f_{jk} \in \mathbb{R}^n$ voneinander abhängig, sodass in jedem Zeitschritt von t_k nach t_{k+1} ein i. A. nichtlineares Gleichungssystem der Dimension $n\,r$ gelöst werden muss:

$$f_{jk} = f(t_k + \alpha_j\, h_k, y_k + h_k \sum_{i=1}^{r} \beta_{ji}\, f_{ik}), \quad j = 1, \ldots, r\,. \tag{3.31}$$

Dies ersetzt die Schritte 2–4 im Algorithmus Algorithmus 3.29 für ein ERKV.

Falls f affin-linear in y ist, also $f(t, y) = A(t)\, y + b(t)$, dann ist (3.31) ein lineares Gleichungssystem (LGS).[17]

Wir klären zunächst die Frage nach der eindeutigen Lösbarkeit von (3.31), vgl. Sheet 1, Exercise 3.

[16] Die Implizitheit des Verfahrens versteckt sich bei unserer linearen skalaren Dgl (3.27) in der „Invertierung“ der Zahl $1 - h\,\lambda$.

[17] Dieses Gleichungssystem wird in Aufgabe Sheet 2, Exercise 5 hergeleitet.

Satz 3.33 (Lösbarkeit von (3.31))**.**
Es sei f Lipschitz-stetig bzgl. y, also

$$\|f(t,y_1)-f(t,y_2)\|_\infty \le L\,\|y_1-y_2\|_\infty$$

für alle $t\in[0,T]$ und $y_1,y_2\in\mathbb{R}^n$. Dann ist das System (3.31) für alle Schrittweiten

$$h < \frac{1}{L\,\max\limits_{j=1,\dots,r}\sum_{i=1}^r|\beta_{ji}|}$$

eindeutig lösbar.

Beachte: Im Nenner steht die Zeilensummennorm $\|B\|_\infty$ der Matrix B, also die durch die Maximum-Norm auf $\mathbb{R}^n$ induzierte Matrixnorm.

Beweis: Wir verstehen (3.31) als ein großes System

$$v = F(v)$$

mit dem Vektor der Unbekannten $v := (f_{1k}^\top,\dots,f_{rk}^\top)^\top\in\mathbb{R}^{n\cdot r}$ und

$$F_j(v) := \underbrace{f(t_k+\alpha_j\,h_k,\,y_k+h_k\sum_{i=1}^r\beta_{ji}\,f_{ik})}_{\text{rechte Seite in (3.31)}},\quad j=1,\dots,r. \tag{3.32}$$

Idee: Benutze den Banachschen Fixpunktsatz.

Wir verwenden auf $\mathbb{R}^{n\,r}$ die Maximumnorm. Dann ist

$$\begin{aligned}
&\|F(v)-F(\widetilde v)\|_\infty\\
&= \max_{j=1,\dots,r}\Big\|f(t_k+\alpha_j\,h_k,\,y_k+h_k\sum_{i=1}^r\beta_{ji}\,f_{ik})-f(t_k+\alpha_j\,h_k,\,y_k+h_k\sum_{i=1}^r\beta_{ji}\widetilde f_{ik})\Big\|_\infty\\
&\le L\,\max_{j=1,\dots,r}h\,\Big\|\sum_{i=1}^r\beta_{ji}\,(f_{ik}-\widetilde f_{ik})\Big\|_\infty\\
&\le L\,h\,\max_{j=1,\dots,r}\sum_{i=1}^r|\beta_{ji}|\,\|f_{ik}-\widetilde f_{ik}\|_\infty\\
&\le L\,h\,\max_{j=1,\dots,r}\sum_{i=1}^r|\beta_{ji}|\,\|v-\widetilde v\|_\infty.
\end{aligned}$$

Falls also h wie gefordert klein ist, so ist F eine Kontraktion, und die Behauptung folgt aus dem Banachschen Fixpunktsatz. □

Bemerkung 3.34. Achtung, in Satz 3.33 setzen wir die globale Lipschitz-Stetigkeit von f bzgl. y voraus. Wenn man dies auf Q einschränkt, braucht man noch den Nachweis der Selbstabbildung, siehe auch Langer, 1996, S.94.

Aus dem Fixpunktsatz geht auch hervor, dass die Steigungen $v := (f_{1k}^\top,\dots,f_{rk}^\top)^\top$ für hinreichend kleines h über die Fixpunktiteration

$$v^{(\ell+1)} := F(v^{(\ell)}),\quad \ell=0,1,2,\dots\,.$$

bestimmt werden können. Eine Alternative ist das Newton-Verfahren:

Im allgemeinen Fall, dass B eine vollbesetzte Matrix ist, hat man in jedem Newton-Schritt zur Lösung von $v - F(v) = 0$ das LGS

$$\big(I_{nr\times nr} - F'(v^{(\ell)})\big)\,\Delta v = -(v^{(\ell)} - F(v^{(\ell)})) \tag{3.33}$$

zu lösen. Anschließend wird $v^{(\ell+1)} := v^{(\ell)} + \Delta v$ gesetzt. Dabei hat die Matrix $F'(k)$ die Block-Struktur

$$F'(v) = \begin{pmatrix} \boxed{\frac{\partial F_1}{\partial v_1}} & \boxed{\frac{\partial F_1}{\partial v_2}} & \cdots & \boxed{\frac{\partial F_1}{\partial v_r}} \\ \boxed{\frac{\partial F_2}{\partial v_1}} & & & \vdots \\ \vdots & & & \vdots \\ \boxed{\frac{\partial F_r}{\partial v_1}} & \cdots & \cdots & \boxed{\frac{\partial F_r}{\partial v_r}} \end{pmatrix},$$

wobei in (3.33) alle Blöcke an der Stelle $v^{(\ell)}$ (der aktuellen Iterierten) ausgewertet werden. Nach Kettenregel, angewendet auf (3.32), gilt für jeden Ableitungsblock

$$\frac{\partial F_j(v)}{\partial v_m} = \underbrace{f_y(t_k + \alpha_j\, h_k, y_k + h_k \sum_{i=1}^{r} \beta_{ji}\, v_i)}_{\in \mathbb{R}^{n\times n}}\ \underbrace{h_k\,\beta_{jm}}_{\in\mathbb{R}} \in \mathbb{R}^{n\times n}. \tag{3.34}$$

In *jedem Schritt* des Newton-Verfahrens (3.33) muss man

- $F(v)$ auswerten, d. h. r Auswertungen von f vornehmen (wie bei ERKV),
- $F'(v)$ auswerten, d. h. die Jacobimatrix $f_y(\cdots)$ an r Stellen bestimmen (pro Blockzeile von $F'(v)$ an einer Stelle, indiziert durch j),
- die Matrix $I_{nr\times nr} - F'(v)$ faktorisieren (Aufwand der LR-Zerlegung $\sim (nr)^3$).

In der Praxis verwendet man oft ein vereinfachtes Newton-Verfahren, in dem $F'(k)$ z. B. nur in der ersten Iteration $v^{(0)}$ ausgewertet und dann in allen folgenden Newton-Schritten desselben Zeitschrittes weiterverwendet wird.[18]

Bemerkung 3.35. Die LGS (3.33) sind für h klein genug immer eindeutig lösbar und die Kondition der Systemmatrix geht gegen 1 für $h \searrow 0$, da

$$\big(I_{nr\times nr} - F'(v^{(\ell)})\big) = \big(I_{nr\times nr} - hQ\big)$$

mit einer festen Matrix $Q \in \mathbb{R}^{nr\times nr}$, siehe (3.34).

Definition 3.36 (Klassifikation von IRKV)**.** Ein implizites RKV heißt

(a) **diagonal-implizit** (DIRK), falls B eine untere Dreiecksmatrix ist,

(b) **einfach diagonal-implizit** (SDIRK), falls B eine untere Dreiecksmatrix ist und alle Hauptdiagonaleinträge identisch sind,

(c) **voll implizit**, falls B keine untere Dreiecksmatrix ist.

[18] Dadurch wird die q-quadratische Konvergenz des Newton-Verfahrens auf q-lineare Konvergenz reduziert. Praktisch ist man jedoch oft mit der Startnäherung so nah an der Lösung, dass auch mit dem vereinfachten Verfahren nur 2–3 Iterationen benötigt werden.

Ist B eine untere Dreiecksmatrix (DIRK), so hat $F'(k)$ die Struktur

$$F'(v) = \begin{pmatrix} \boxed{\frac{\partial F_1}{\partial v_1}} & & & \\ \boxed{\frac{\partial F_2}{\partial v_1}} & \boxed{\frac{\partial F_2}{\partial v_2}} & & \\ \vdots & & \ddots & \\ \boxed{\frac{\partial F_r}{\partial v_1}} & \boxed{\frac{\partial F_r}{\partial v_2}} & \cdots & \boxed{\frac{\partial F_r}{\partial v_r}} \end{pmatrix}.$$

Das Newtonsystem (3.33) zerfällt dann in r lineare Gleichungssysteme der Dimension $n \times n$ mit den Matrizen

$$I_{n\times n} - \frac{\partial F_j(v)}{\partial v_j} = I_{n\times n} - f_y(t_k + \alpha_j\, h_k, y_k + h_k \sum_{i=1}^{j} \beta_{ji}\, v_i)\, h_k\, \beta_{jj} \quad j = 1, \ldots, r. \tag{3.35}$$

Der Aufwand der LR-Zerlegung sinkt auf $\sim n^3 r$.

Beispiel 3.37 (SDIRK-Verfahren der Ordnung 3)**.** Wir konstruieren als Beispiel alle SDIRK-Verfahren der Ordnung $p = 3$ mit minimaler Stufenzahl r. Wieviele Stufen werden benötigt?

Autonomieinvarianz	$r + 1$ Bedingungen (3.24)
Ordnung 2:	1 Bedingung Satz 3.27 (b)
Ordnung 3:	2 Bedingungen Satz 3.27 (c).

Bei $r = 2$ Stufen haben wir also 6 Parameter und 6 Gleichungen. Diese Bedingungen für das Butcher-Diagramm

$$\begin{array}{c|cc} \alpha_1 & \beta & \\ \alpha_2 & \beta_{21} & \beta \\ \hline & \gamma_1 & \gamma_2 \end{array}$$

lauten:

$$\gamma_1 + \gamma_2 = 1, \qquad \alpha_1 = \beta,$$
$$\gamma_1\alpha_1 + \gamma_2\alpha_2 = \frac{1}{2}, \qquad \alpha_2 = \beta_{21} + \beta,$$
$$\gamma_1\alpha_1^2 + \gamma_2\alpha_2^2 = \frac{1}{3}, \qquad \gamma_1\alpha_1\beta + \gamma_2\alpha_1\beta_{21} + \gamma_2\alpha_2\beta = \frac{1}{6}.$$

Die Lösungen dieses nichtlinearen Systems sind

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

mit $\omega = \frac{1}{2}\left(1 \pm \frac{\sqrt{3}}{3}\right)$. Für die Wahl „−“ sind alle Koeffizienten positiv.

Frage: Welche Ordnung ist mit allgemeinen r-stufigen IRKV erreichbar?

Lemma 3.38 (Maximale Ordnung von IRKV)**.** Für die Ordnung p eines r-stufigen RKV gilt $p \le 2r$.

Beweis: Ein r-stufiges RKV der Ordnung p erzeugt bei Anwendung auf $y'(t) = f(t)$ eine r-stufige Quadraturformel vom Genauigkeitsgrad $q = p - 1$ (Bemerkung 3.28).

Für den Genauigkeitsgrad einer r-stufigen Quadraturformel gilt aber $q \leq 2r-1$, siehe Numerik-Grundvorlesung. Daraus folgt die Behauptung. □

Zur Erinnerung: Für die Ordnung *expliziter* RKV gilt $p \leq r$, und Gleichheit ist nur bis $r = 4$ erreichbar, siehe Tabelle am Ende von § 3.3.2.

Zum Abschluss dieses Abschnitts untersuchen wir noch einige voll-implizite Verfahren.

Verfahren der Ordnung 2r

Zunächst wenden wir das IRKV angewendet auf die triviale DGL

$$y'(t) = f(t) \qquad t \in [t_k, t_{k+1}]$$

an. Dies ist eine Quadraturaufgabe und es hat sich gezeigt, dass durch Gauß-Quadratur die höchstmögliche Approximation des Integrals erreicht wird

$$\int_{t_k}^{t_{k+1}} f(t)\mathrm{d}t \approx h_k \sum_{j=1}^{r} \gamma_j f(t_k + \alpha_j\, h_k),$$

wobei die Stützstellen α_j, $j = 1, \ldots, r$, die Nullstellen des (verschobenen) **Legendre-Polynoms**

$$\widehat{\phi}_r(t) := \frac{1}{r!}\frac{\mathrm{d}^r}{\mathrm{d}t^r}\left(t^r(t-1)^r\right) \tag{3.36}$$

vom Grad r sind. Die Quadraturformel ist vom Exaktheitsgrad $2r-1$. Die restlichen Koeffizienten β_{ij} und γ_j können eindeutig aus den Ordnungsbedingungen bestimmt werden. Die damit konstruierten RKV, welche auch **Gauß-Verfahren** oder **Gauß-IRKV** genannt werden, besitzten die Ordnung $p = 2r$ (vgl. Sheet 2, Exercise 6).

Beispiel 3.39 (Gauß-Verfahren der Ordnungen 2, 4, 6)**.**

(a) Das Gauß-Verfahren der Ordnung 2 ist die implizite Mittelpunktsregel

$$\begin{array}{c|c} 1/2 & 1/2 \\ \hline & 1 \end{array}\,.$$

(b) Das Gauß-Verfahren der Ordnung 4 ist gegeben durch

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

(c) Das Gauß-Verfahren der Ordnung 6 ist gegeben durch

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

Verfahren der Ordnung $2r-1$

Indem wir eine geringere Ordnung fordern, stehen uns zusätzliche Freiheitsgrade in den Koeffizienten zur Verfügung, welche wir für andere wünschenswerte Eigenschaften ausnutzen können.

Idee: Wähle α_j als die Nullstellen des Polynoms

$$\widehat{\phi}_r(t) + \xi\, \widehat{\phi}_{r-1}(t) \quad \text{für ein } \xi \in \mathbb{R}$$

mit $\widehat{\phi}_r$ aus (3.36).

Man unterscheidet:

- **Radau-I-Verfahren**:
  Wähle $\xi = 1$, daraus ergibt sich immer $\alpha_1 = 0$.
- **Radau-II-Verfahren**:
  Wähle $\xi = -1$, daraus ergibt sich immer $\alpha_r = 1$.

Durch die Ordnungsbedingungen und zusätzliche Annahmen sind die Radau-Verfahren gegebener Ordnung eindeutig.

Beispiel 3.40 (Radau-I- und Radau-II-Verfahren)**.**

Beachte: Diese Verfahren sind nicht alle autonom-invariant.

(a) Das Radau-I-Verfahren der Ordnung 1 ist gegeben durch

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

(b) Das Radau-I-Verfahren der Ordnung 3 ist gegeben durch

$$\begin{array}{c|cc} 0 & \frac{1}{4} & -\frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array} .$$

(c) Das Radau-I-Verfahren der Ordnung 5 ist gegeben durch

$$\begin{array}{r|lll} 0 & \frac{1}{9} & \frac{-1-\sqrt{6}}{18} & \frac{-1+\sqrt{6}}{18} \\ \frac{6-\sqrt{6}}{10} & \frac{1}{9} & \frac{88+7\sqrt{6}}{360} & \frac{88-43\sqrt{6}}{360} \\ \frac{6+\sqrt{6}}{10} & \frac{1}{9} & \frac{88+43\sqrt{6}}{360} & \frac{88-7\sqrt{6}}{360} \\ \hline & \frac{1}{9} & \frac{16+\sqrt{6}}{36} & \frac{16-\sqrt{6}}{36} \end{array} .$$

(d) Das Radau-II-Verfahren der Ordnung 1 ist das implizite Euler-Verfahren

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

(e) Das Radau-II-Verfahren der Ordnung 3 ist gegeben durch

$$\begin{array}{r|ll} \frac{1}{3} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{3}{4} & \frac{1}{4} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array} .$$

(f) Das Radau-II-Verfahren der Ordnung 5 ist gegeben durch

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

Verfahren der Ordnung $2r-2$

Idee: Wähle α_j als die Nullstellen eines Polynoms

$$\widehat{\phi}_r(t) + \xi\,\widehat{\phi}_{r-1}(t) + \mu\,\widehat{\phi}_{r-2}(t) \quad \text{für } \xi, \mu \in \mathbb{R}$$

mit $\widehat{\phi}$ aus (3.36).

Durch die Wahl $\xi = 0$ und $\mu = -1$ entstehen sogenannte Lobatto-III-Verfahren mit $\alpha_1 = 0$ und $\alpha_r = 1$. Man unterscheidet Lobatto-IIIA-, -IIIB- und -IIIC-Verfahren.

Beispiel 3.41 (Lobatto-IIIA- und Lobatto-IIIB-Verfahren)**.**

(a) Das Lobatto-IIIA-Verfahren der Ordnung 4 ist

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

(b) Das Lobatto-IIIB-Verfahren der Ordnung 4 ist

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

§ 3.4 Stabilitätsbegriffe bei Einschrittverfahren

Wir untersuchen in diesem Abschnitt die Eignung expliziter und impliziter RKV für bestimmte Typen von (**AWP**). Diese Eignung führt zu entsprechenden „Stabilitätsbegriffen“, nicht zu verwechseln mit der diskreten Stabilität (Lemma 3.15), die alle RKV (ggf. für $h \leq \overline{h}$) besitzen.

§ 3.4.1 A-Stabilität

In § 3.3.3 hatten wir (als Motivation für implizite RKV) das explizite und implizite Euler-Verfahren auf das AWP (3.27) angewendet. Wir betrachten hier zunächst erneut die skalare **Dahlquist'sche Testgleichung**

$$y'(t) = \lambda\, y, \quad y(0) = 1 \tag{3.37}$$

etwas allgemeiner mit $\lambda \in \mathbb{C}$. Für die exakte Lösung ist $y(t) = y_a\, \mathrm{e}^{\lambda t}$ gilt

$$|y(t)| \to \begin{cases} 0, & \text{falls } \mathrm{Re}(z) < 0, \\ \infty, & \text{falls } \mathrm{Re}(z) > 0. \end{cases}$$

Wünschenswert wäre es, wenn unser numerisches Verfahren diese Eigenschaft bei äquidistanten Schrittweiten ($h_k = h$, $k \in \mathbb{N}_0$) erhält. Insbesondere fordern wir zunächst folgende Monotonie-Eigenschaft

$$|y_k| \leq |y_{k-1}|, \quad \text{für alle } k = 1, 2, \ldots . \tag{3.38}$$

Definition 3.42 (A-Stabilität)**.**

(a) Ein numerisches Verfahren zur Lösung von **(AWP)** heißt **A-stabil** oder **absolut stabil**, falls (3.38) erfüllt ist.

Beispiel 3.43 (Stabilität des expliziten Euler-Verfahrens)**.** Angewendet auf die Testgleichung (3.37) ergibt sich

$$y_k = (1+\lambda h)y_{k-1} = \ldots = (1+\lambda h)^k \underbrace{y_0}_{=1}.$$

Somit ist dieses Verfahren A-stabil für

$$|1+\lambda| \le 1 \quad \text{(Kreis mit Radius 1 im Mittelpunkt } (-1,0).$$

Dies stellt eine Bedingung an h in Abhängigkeit von λ dar:

$$\begin{aligned} & |1+\lambda h| \le 1 \\ \iff \quad & (1+h\,\mathrm{Re}(\lambda))^2 + h^2\,\mathrm{Im}(\lambda)^2 \le 1 \\ \iff \quad & 2h\,\mathrm{Re}(\lambda) + h^2(\mathrm{Re}(\lambda)^2 + \mathrm{Im}(\lambda)^2) \le 0 \\ \iff \quad & 0 < h \le -\frac{2\mathrm{Re}(\lambda)}{|\lambda|^2}. \end{aligned}$$

Wir erhalten also eine obere Schranke für die Wahl des Schrittweitenparameters.

Um Stabilität nachzuweisen, wenden wir also das betrachtete ESV auf (3.37) an und bringen die Vorschrift eines Einzelschrittes auf die Gestalt

$$y_{k+1} = R(h\,\lambda)\,y_k, \quad k = 0,1,2,\ldots. \tag{3.39}$$

Definition (Fortsetzung Definition 3.42)**.**

(b) Die Funktion $R\colon \mathbb{C}^- \supset D \to \mathbb{C}$, $0 \in D$, aus (3.39) heißt **Stabilitätsfunktion**.

(c) Die Menge

$$S = \{z := h\,\lambda \in \mathbb{C} : \text{(3.38) ist erfüllt}\}$$

heißt **Stabilitätsbereich**.

(d) A-Stabilität ist dann Äquivalent zur Forderung

$$|R(h\,\lambda)| \le 1 \tag{3.40}$$

(e) Ein ESV heißt **unbedingt A-stabil**, falls es A-stabil für alle $z \in \mathbb{C}$ mit $\mathrm{Re}(z) < 0$ ist. Andernfalls heißt es **A-stabil unter der Bedingung** $z \in S$.

Für unbedingt A-stabile Verfahren ergibt sich also unabhängig von $\lambda \in \mathbb{C}^-$ *keine* Einschränkung der Schrittweite h, für andere Verfahren können Einschränkungen entstehen. Dies hatten wir bereits in § 3.3.3 beobachtet.

Beispiel 3.44 (Beispiele für Stabilitätsfunktionen[19]).

Verfahren	Stabilitätsfunktion $R(z)$
explizites Euler-Verfahren	$1+z$
verbessertes Euler-Verfahren	$1+z+\frac{1}{2}z^2$
implizites Euler-Verfahren	$\frac{1}{1-z}$
implizite Mittelpunktsregel	$\frac{1+z/2}{1-z/2}$
implizite Trapezregel	dito

Der folgende Satz zeigt, dass jedes RKV eine SF besitzt und welche Gestalt diese hat.

Satz 3.45 (Stabilitätsfunktion von RKV[20]). Ein RKV mit Butcher-Diagramm $\begin{array}{c|c} a & B \\ \hline & c^\top \end{array}$ besitzt die SF

$$R(z) = 1 + z\, c^\top (I_{r\times r} - zB)^{-1}\, \vec{1} \tag{3.41a}$$

$$= \frac{\det(I_{r\times r} - zB + z\,\vec{1}\, c^\top)}{\det(I_{r\times r} - zB)}. \tag{3.41b}$$

mit $\vec{1} = (1,1,\ldots,1)^\top \in \mathbb{R}^s$. Die SF ist definiert für alle $z \in \mathbb{C}$, sodass $1/z$ kein Eigenwert von B ist.

Beachte: Die SF ist eine gebrochen rationale Funktion. Zähler und Nenner in (3.41b) sind Polynome in z mit reellen Koeffizienten vom Höchstgrad r mit Funktionswert 1 bei $z = 0$. Es gehen nur die Koeffizienten B und c ein.

Beweis: Die Anwendung des RKV auf die skalare Testgleichung (3.37) ergibt die Steigungen

$$f_{jk} = f(t_k + \alpha_j\, h, y_k + h\sum_{i=1}^{r} \beta_{ji}\, f_{ik}) = \lambda\left(y_k + h\sum_{i=1}^{r} \beta_{ji}\, f_{ik}\right) \in \mathbb{C}$$

bzw. als ein LGS für $v = (f_{1k},\ldots,f_{rk})^\top \in \mathbb{C}^r$:

$$\begin{aligned} & v = \lambda\, y_k\, \vec{1}_r + h\,\lambda\, B\, v \\ \Rightarrow \quad & v = \lambda\, y_k \left(I_{r\times r} - h\,\lambda\, B\right)^{-1} \vec{1}_r, \end{aligned}$$

[19] In Sheet 3, Exercise 8 werden die angegebenen SF bestätigt sowie die SF der Taylorreihen-Methode hergeleitet.

[20] Hermann, 2004, Satz 4.1

falls die Matrix invertierbar ist, d. h., falls $1/(h\lambda)$ kein Eigenwert von B ist. Der Schritt zu y_{k+1} liefert also

$$\begin{aligned} y_{k+1} &= y_k + h \sum_{j=1}^{r} \gamma_j f_{jk} \\ &= y_k + h\, c^\top v \\ &= y_k + h\lambda y_k\, c^\top \left(I_{r\times r} - h\lambda B\right)^{-1} \vec{1}_r \\ &= \left[1 + h\lambda\, c^\top \left(I_{r\times r} - h\lambda B\right)^{-1} \vec{1}_r\right] y_k. \end{aligned}$$

Damit ist (3.41a) gezeigt:

$$R(z) = 1 + z\, c^\top (I_{r\times r} - zB)^{-1} \vec{1}.$$

Die Darstellung (3.41b) wird in Sheet 3, Exercise 9 gezeigt. □

Folgerung 3.46 (Stabilitätsfunktionen von ERKV)**.** Für r-stufige ERKV gilt

$$R(z) = 1 + z\, c^\top \sum_{j=0}^{r-1} (zB)^j \vec{1},$$

d. h., die SF ist ein Polynom vom Höchstgrad r. Daher ist in diesem Falle $R(z)$ definiert für alle $z \in \mathbb{C}$.

Beweis: Stelle die Inverse in (3.41a) über die Neumannsche Reihe dar:

$$(I - zB)^{-1} = \sum_{j=0}^{\infty} (zB)^j = \sum_{j=0}^{r-1} (zB)^j.$$

Die Reihe konvergiert, sie hat sogar nur endlich viele Summanden, weil B als strikte untere Dreiecksmatrix nilpotent ist ($B^r = 0$).

Da B strikte untere Dreiecksmatrix ist, hat B nur Null als Eigenwert (algebraische Vielfachheit r). Daher kann $\frac{1}{z}$ für kein $z \in \mathbb{C}$ ein Eigenwert von B sein. □

Folgerung 3.47 (Aussagen über ERKV)**.**

Konsistente ERKV sind niemals unbedingt A-stabil.

Beweis: Konsistente RKV haben bereits mindestens die Ordnung 1. Die SF $R(z)$ eines expliziten RKV der Ordnung $p \geq 1$ ist nach Folgerung 3.46 ein Polynom vom Grad $\geq p$ und damit insbesondere unbeschränkt auf $\mathbb{C}^-$. □

Frage: Wie groß sind die Stabilitätsgebiete bekannter Verfahren?

Es gilt

$$\begin{aligned} &\{z \in \mathbb{C} : |R(z)| \leq 1\} = S = \overline{S} \\ &\{z \in \mathbb{C} : |R(z)| < 1\} \subset \operatorname{int} S \\ \Rightarrow\quad &\partial S = \overline{S} \setminus \operatorname{int} S \subset \{z \in \mathbb{C} : |R(z)| = 1\}. \end{aligned}$$

Den Rand des (nicht notwendig zusammenhängenden) Stabilitätsgebietes S kann man also bestimmen, indem man für $\varphi \in [0, 2\pi]$ die Lösungen z der Gleichung

$$R(z) = \mathrm{e}^{\mathrm{i}\varphi}$$

berechnet, da sich jedes $z \in \mathbb{C}$ mit $|z| = 1$ als $z = \mathrm{e}^{\mathrm{i}\varphi}$ darstellen lässt.

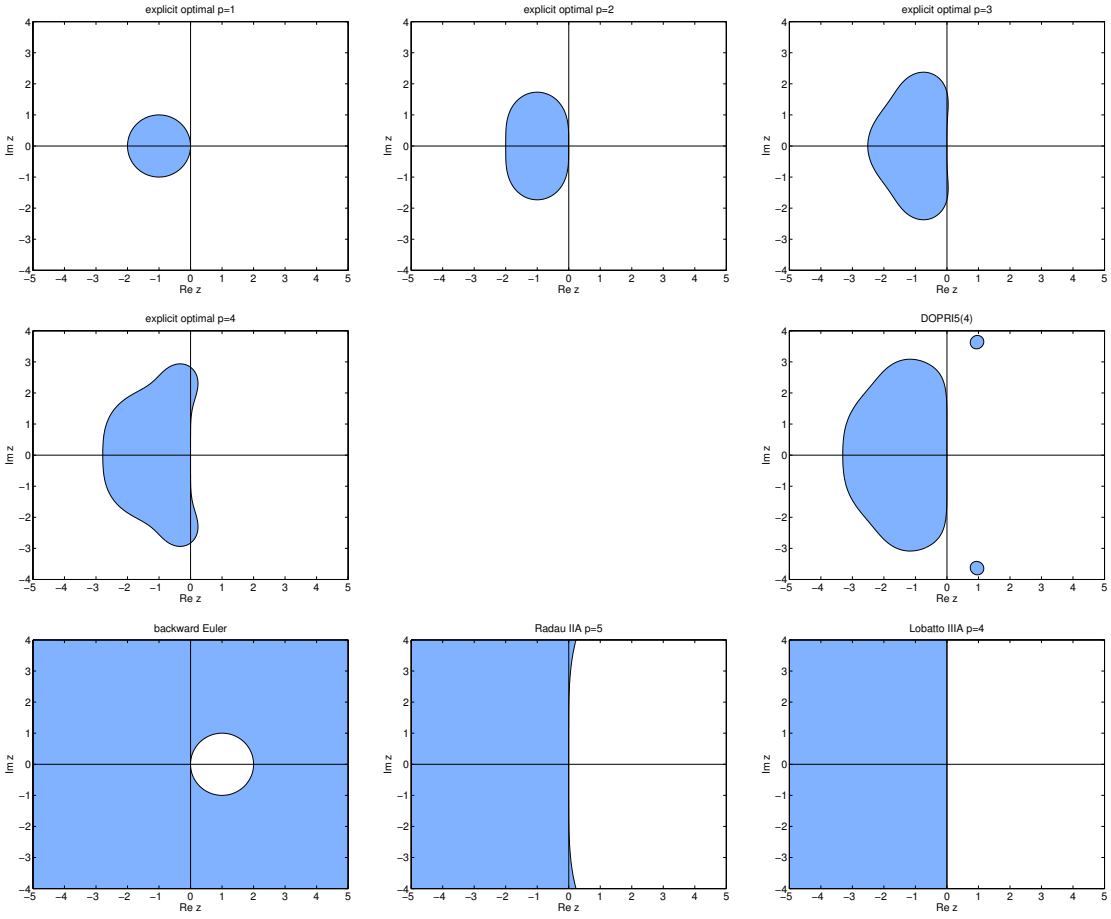


ABBILDUNG 3.7. Stabilitätsgebiete einiger RKV.

Beispiel 3.48 (Stabilitätsgebiete von ERKV)**.** Die Stabilitätsgebiete der expliziten RKV der optimalen Ordnung $p = s \leq 4$ sind in Abbildung 3.7 dargestellt.[21]

Für $\lambda \in \mathbb{R}^-$ sind die **Stabilitätsintervalle** $S \cap \mathbb{R}$ relevant:

Ordnung p	1	2	3	4	5
Stabilitätsintervall	$[-2, 0]$	$[-2, 0]$	$[-2.51, 0]$	$[-2.78, 0]$	$[-3.21, 0]$

Beispiel 3.49 (Stabilitätsgebiet der θ-Methode in Abhängigkeit von θ)**.** Wir betrachten die ein-parametrige Familie von RKV mit $\theta \in [0, 1]$ (**θ-Methode**)

$$y_{k+1} = y_k + h\big[(1-\theta)\, f(t_k, y_k) + \theta\, f(t_{k+1}, y_{k+1})\big]^{22},$$

Diese hat Konsistenzordnung 1 und enthält als Sonderfälle für

$\theta = 0$ das explizite Euler-Verfahren,

$\theta = 1$ das implizite Euler-Verfahren und

$\theta = \frac{1}{2}$ die implizite Trapezregel (Crank-Nicolson-Verfahren) (Ausnahme: Konsistenzordnung 2).

[21] Matlab-Demo: `rkm_stability_region.m`

Das Butcher-Diagramm ist also

$$\begin{array}{c|cc} 0 & & \\ 1 & (1-\theta) & \theta \\ \hline & (1-\theta) & \theta \end{array}.$$

Damit ist

$$I_{r\times r} - zB = \begin{bmatrix} 1 & \\ -z\,(1-\theta) & 1 - z\,\theta \end{bmatrix}, \qquad z\,\vec{1}\,c^\top = \begin{bmatrix} z\,(1-\theta) & z\,\theta \\ z\,(1-\theta) & z\,\theta \end{bmatrix}.$$

Also ergibt (3.41b)

$$\begin{aligned} R(z) &= \frac{\det(I_{r\times r} - zB + z\,\vec{1}\,c^\top)}{\det(I_{r\times r} - zB)} \\ &= \frac{(1 + z\,(1-\theta))\,1 - (z\,\theta)\,0}{(1 - z\,\theta) - 0} \\ &= \frac{1 + (1-\theta)\,z}{1 - \theta\,z}. \end{aligned}$$

Für das Stabilitätsgebiet S gilt:

$$S = \begin{cases} \{z \in \mathbb{C} : |z - \frac{1}{2\theta-1}| \le \frac{1}{1-2\theta}\} & \text{für } \theta < 1/2 \\ \{z \in \mathbb{C} : \mathrm{Re}(z) \le 0\} & \text{für } \theta = 1/2 \\ \{z \in \mathbb{C} : |z - \frac{1}{2\theta-1}| \ge \frac{1}{2\theta-1}\} & \text{für } \theta > 1/2. \end{cases}$$

Die Stabilitätsgebiete sind in Abbildung 3.8 für verschiedene Werte von θ dargestellt. Die θ-Methode ist also A-stabil für $\theta \ge 1/2$, siehe Abbildung 3.8.

§ 3.4.2 Steife Differentialgleichungen

Wir betrachten nun in Erweiterung der Testgleichung (3.37) das System

$$y'(t) = A\,y(t), \quad y(0) = y_a \tag{3.42}$$

mit $A \in \mathbb{C}^{n\times n}$ und $y_a \in \mathbb{C}^n$. Die exakte Lösung ist $y(t) = \mathrm{e}^{At}\,y_a$, sie fällt betragsmäßig für beliebige AW $y_a \neq 0$ genau dann streng monoton für hinreichend große $t > 0$, wenn $\mathrm{Re}(\lambda_j) < 0$ für alle Eigenwerte λ_j von A gilt. [23]

Ein ESV mit der (gebrochen-rationalen) SF R erzeugt die Näherungen

$$y_{k+1} = R(hA)\,y_k, \quad i = 0, 1, 2, \ldots$$

Diese fallen für gegebenes $h > 0$ betragsmäßig genau dann streng monoton für hinreichend große i, wenn $|R(h\,\lambda_j)| < 1$ für alle Eigenwerte λ_j von A gilt.

Definition 3.50 (Steife Differentialgleichung)**.**

(a) Das lineare Dgl-System (3.42) heißt **steif**, wenn gilt:

(*i*) $\mathrm{Re}(\lambda_j) < 0$ für alle Eigenwerte λ_j von A.

(*ii*) Der **Steifigkeitsquotient** (**SQ**) $\dfrac{\max_{j=1,\ldots,n}|\mathrm{Re}(\lambda_j)|}{\min_{j=1,\ldots,n}|\mathrm{Re}(\lambda_j)|}$ ist $\gg 1$.

[22] Implementierung der θ-Methode in Sheet 8, Exercise 22

[23] Dies kann man durch Diagonalisierung von A zeigen bzw. für nicht-diagonalisierbares A mit Hilfe der Jordanschen Normalform: Es gilt $y(t) = \sum_{j=1}^{r} \exp(\lambda_j\,t)\,p_j(t)$ mit Eigenwerten λ_j der Vielfachheiten n_j und Polynomen p_j vom Grad $\le n_j - 1$, vgl. Heuser, 1991, S. 467.

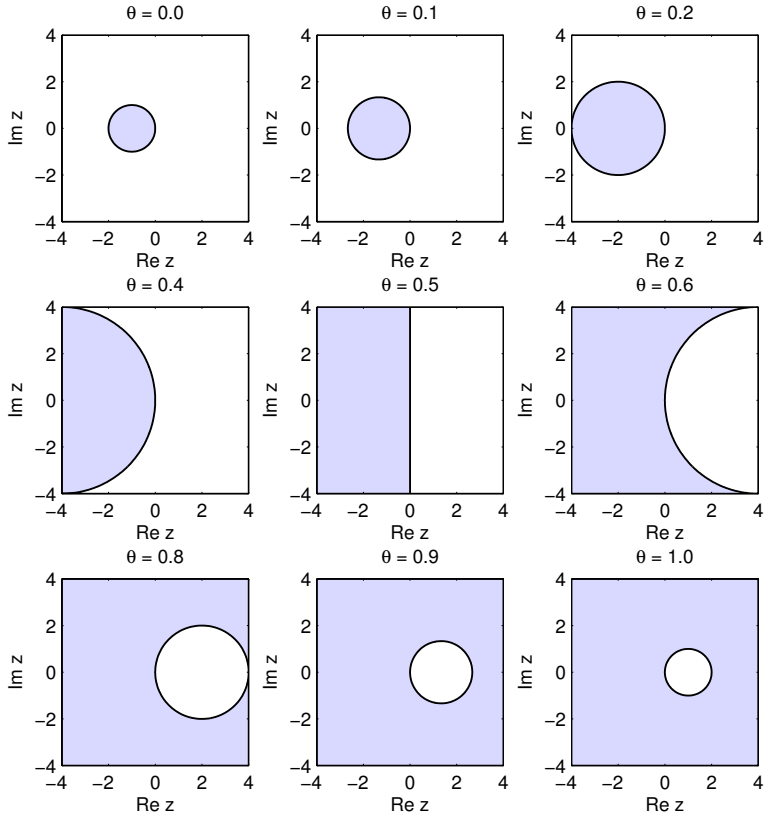


ABBILDUNG 3.8. Stabilitätsgebiete der θ-Methode für verschiedene $\theta \in [0, 1]$.

(b) Das Dgl-System $y' = f(t, y)$ heißt **steif** in einer Umgebung der Lösung $y(\cdot)$, wenn (i) und (ii) für die Jacobimatrix $A(t) = f_y(t, y(t))$ erfüllt sind (das wäre das linearisierte System). [24]

Das Problem steifer Dgl ist folgendes: Wegen $\mathrm{Re}(\lambda_j) < 0$ für alle Eigenwerte λ_j von A klingt die exakte Lösung betragsmäßig für große t ab. Die langsamsten Komponenten der Lösung, die gerade zum EW

$$\lambda_k, \qquad k = \arg\min_j |\mathrm{Re}(\lambda_j)|$$

gehören, bestimmen die globale Zeitskala[25] $[0, T]$ der Aufgabe.
Falls *nicht* mit einem A-stabilen ESV integriert wird, dann schränkt die Bedingung $|R(h\,\lambda_j)| < 1$ die Wahl der Schrittweite h ein. Hierbei sind die schnellsten Komponenten der Lösung, die zum EW

$$\lambda_k, \qquad k = \arg\max_j |\mathrm{Re}(\lambda_j)|$$

gehören, relevant.

[24] In Sheet 4, Exercise 13 wird Steifheit entlang der numerischen Lösungstrajektorie des Brusselator-Modells untersucht.

[25] Man kann eine Zeitskala durch $|y(t)/y'(t)|$ definieren, siehe Hermann, 2004, S. 154.

Bei SQ $\gg 1$ müssen deshalb sehr viele Zeitschritte ausgeführt werden, um die Skala $[0, T]$ abzudecken!

Bemerkung 3.51 (zu steifen Dgl)**.**

(a) Steife Differentialgleichungssysteme können nur mit Verfahren mit einem großen Stabilitätsbereich vernünftig gelöst werden. Insbesondere sollte $S \cap \mathbb{C}^-$ großsein. Explizite Verfahren sind daher wenig geeignet.

(b) Große SQ (Größenordnung 10^6) treten z. B. bei chemischen Reaktionen mit stark unterschiedlichen Zeitskalen auf. Beliebig große SQ entstehen auch bei der Semidiskretisierung parabolischer Differentialgleichungen (Wärmeleitung) mit der Linienmethode, siehe § 12.

(c) Bei nichtlinearen Dgl kann sich der SQ entlang einer Trajektorie ändern. [26]

Beispiel 3.52 (Steife Dgl)**.** Wir geben eine „zufällige“ Matrix $A \in \mathbb{R}^{4\times 4}$ mit Eigenwerten $\lambda_1, \overline{\lambda}_1, \lambda_2, \overline{\lambda}_2$ vor, wobei $\mathrm{Re}(\lambda_1) < \mathrm{Re}(\lambda_2) < 0$. Dazu sei $A := Q^\top C\, Q$ mit $Q \in R^{4\times 4}$ „zufällige“ orthogonale Matrix,

$$\begin{aligned} \lambda_1 &:= a_1 + b_1\mathrm{i}, & a_1, b_1 &\in \mathbb{R}, \\ \lambda_2 &:= a_2 + b_2\mathrm{i}, & a_2, b_2 &\in \mathbb{R}, \end{aligned}$$

$$C := \begin{bmatrix} a_1 & b_1 & & \\ -b_1 & a_1 & & \\ & & a_2 & b_2 \\ & & -b_2 & a_2 \end{bmatrix}.$$

Die Matrix $A = Q^\top C\, Q$ besitzt die gleichen Eigenwerte wie C.

Konkret verwenden wir $\lambda_1 = -1000 + 1000\mathrm{i}$, $\lambda_2 = -1 + 6\mathrm{i}$.

Wir betrachten wieder das System

$$y'(t) = A\, y(t), \quad y(0) = y_a \tag{3.42}$$

auf $[0, T]$, mit $T = 1$ und dem Anfangswert $y_a \in \mathbb{R}^4$ ebenfalls „zufällig“ gewählt.

Die Lösungen sowie Fehler in Abhängigkeit von h sind für explizites und implizites Euler-Verfahren auf äquidistanten Netzen in Abbildung 3.9 dargestellt. Das Verhalten entspricht wie erwartet dem der Testgleichung (3.37) im skalaren Fall.[27]

Wir wissen, dass die Lösungsanteile zu $\lambda_1, \overline{\lambda}_1$ schnell abklingen, die zu $\lambda_2, \overline{\lambda}_2$ langsam. Daher, teilen wir das Intervall $[0, T]$ in $[0, t_s]$ und $[t_s, T]$ auf, wobei t_s so gewählt wird, dass die Lösungsanteile zu $\lambda_1, \overline{\lambda}_1$ bis t_s auf die Größenordnung des erhofften Diskretisierungsfehlers abgeklungen sind,

$$\exp(a_1\, t_s) = h = \frac{1}{N}$$

$$t_s = \frac{\log h}{a_1} = \frac{-\log N}{a_1}.$$

Die beiden Intervalle $[0, t_s]$ und $[t_s, T]$ werden nun mit je N Zeitschritten diskretisiert. Die Lösungen sowie Fehler in Abhängigkeit sind in Abbildung 3.10 dargestellt, wobei

[26] Hierzu zitiert Hermann, 2004 ein schönes Beispiel.

jetzt $h = 1/(2\,N)$ in dem Fehlerverlauf dargestellt wird, um einen fairen Vergleich mit Abbildung 3.9 zu ermöglichen.

Wir stellen fest, dass λ_1 im expliziten Euler-Verfahren auch dann noch die Schrittweiten beschränkt, wenn der zugehörige Lösungsanteil längst keine Rolle mehr spielt. Eine dem Abklingverhalten angepasste Schrittweitenwahl führt nur bei dem Impliziten Euler-Verfahren zu einer Reduktion des Aufwands, auch mit relativ großen Schrittweiten werden brauchbare Näherungen erzielt.

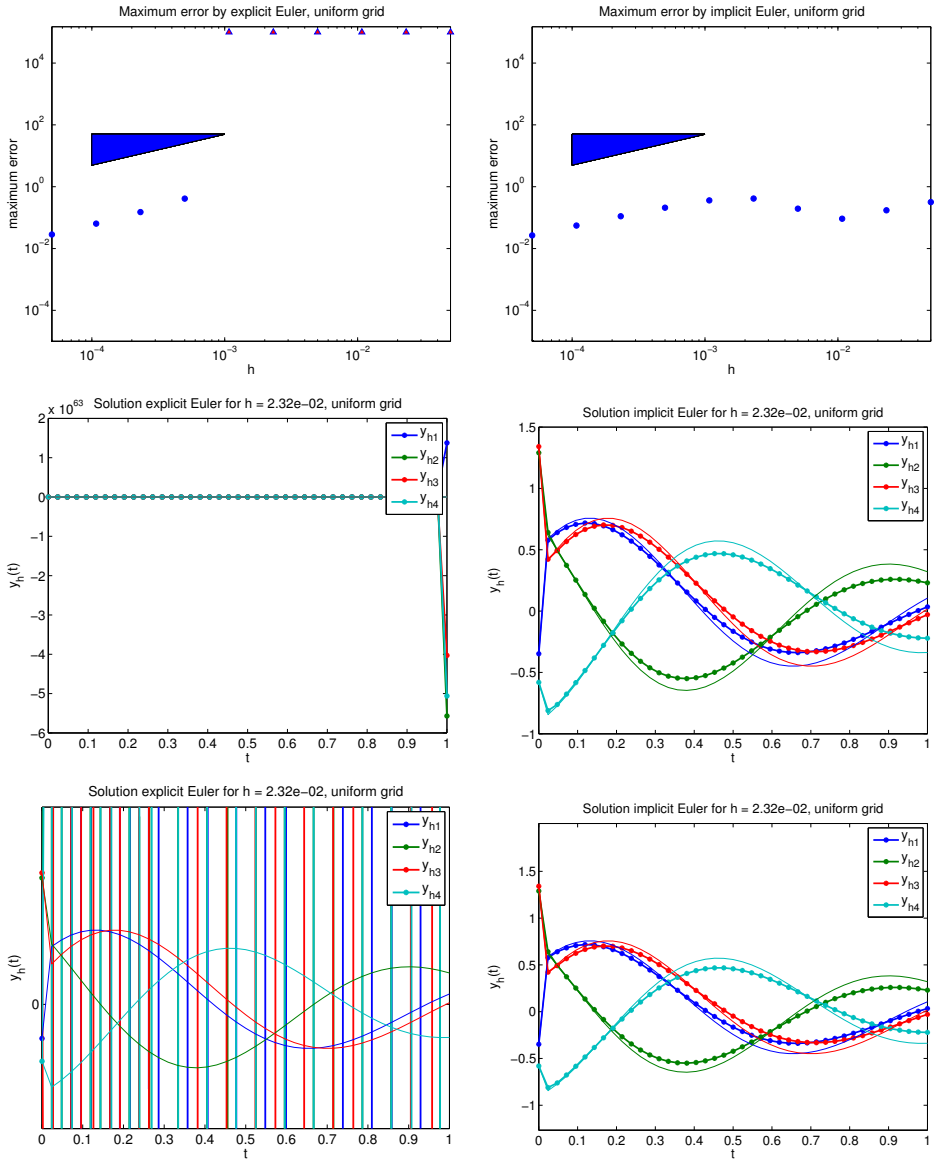


ABBILDUNG 3.9. Explizites (links) und implizites Euler-Verfahren (rechts) für eine steife Dgl, äquidistante Netze.

[27] Hierfür gibt es ein Matlab-Programm: `demo_stiff_n4.m`

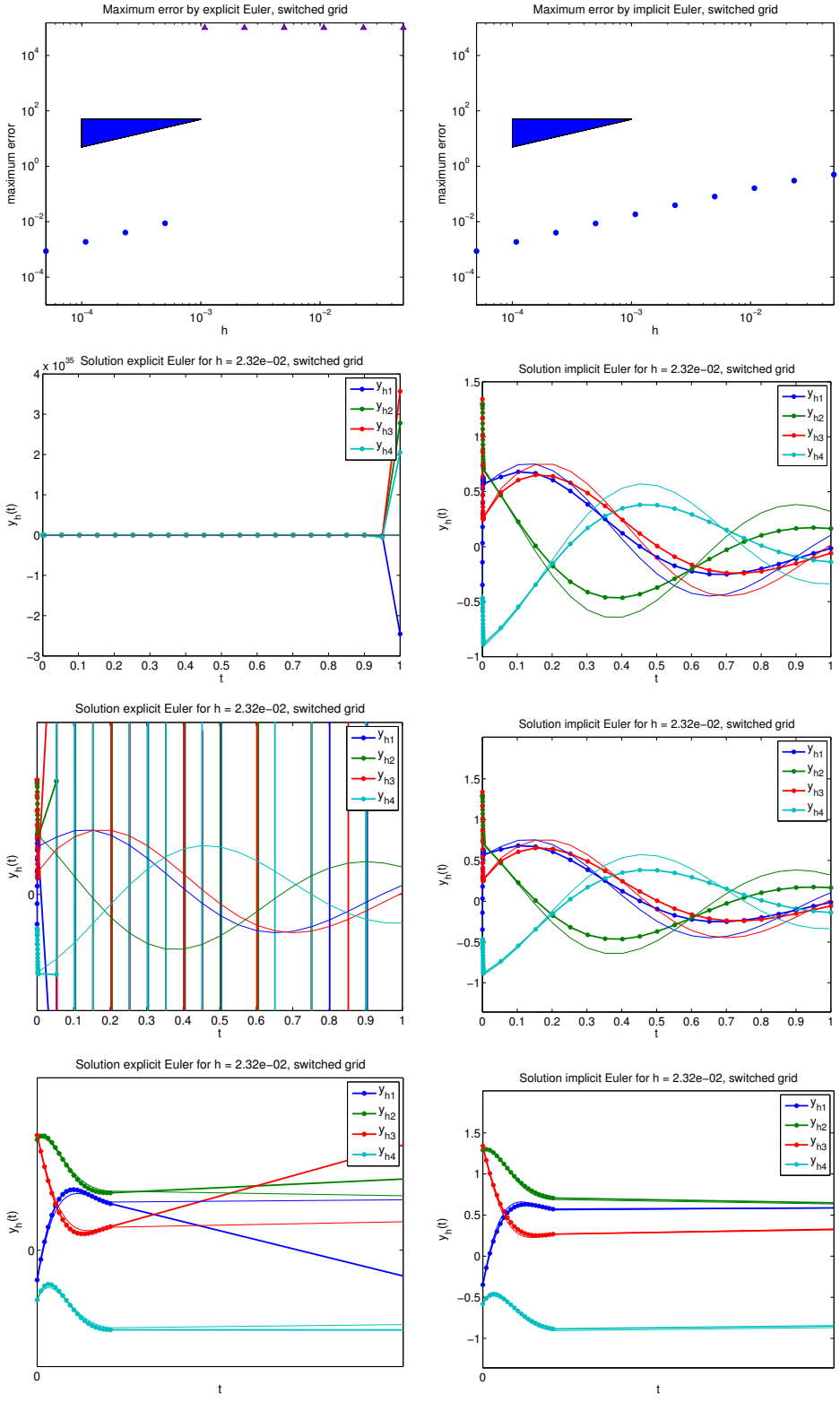


Abbildung 3.10. Explizites (links) und implizites Euler-Verfahren (rechts) für eine steife Dgl, problemangepasste Netze.

§ 3.4.3 L-Stabilität

Definition 3.53 (L-Stabilität)**.** Ein A-stabiles ESV heißt **L-stabil**, wenn die durch das Verfahren erzeugte Näherungslösung von (3.37) für jedes $\lambda \in \mathbb{C}^-$ die Eigenschaft

$$\lim_{k\to\infty} |y_k| \to 0$$

erfüllt.

Bemerkung 3.54 (Bedeutung der L-Stabilität)**.** Das ist das Verhalten, welches wir von der exakten Lösung von (3.37) erwarten. L-Stabilität bedeutet, dass auch die Näherungslösungen diese Eigenschaft haben, denn:

$$y_{k+1} = R(h\,\lambda)\,y_k = \ldots = [R(h\,\lambda)]^{k+1}$$

konvergiert genau dann gegen 0, wenn für $\lambda \in \mathbb{C}^-$

$$\lim_{h\to\infty} R(h\,\lambda) = 0$$

gilt.

Beispiel 3.55 (Eigenschaften der impliziten Trapezregel)**.** Die implizite Trapezregel besitzt die SF

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

Sie ist nach Beispiel 3.49 (mit $\theta = 1/2$) A-stabil, jedoch wegen

$$\lim_{h\to\infty} R(h\,\lambda) = -1 \quad \text{für alle } \lambda \neq 0,$$

nicht L-stabil. Dasselbe gilt für die implizite Mittelpunktsregel, da sie dieselbe SF besitzt.

Der folgende Satz erklärt, dass die Gauß-Verfahren (die IRKV maximaler Konsistenzordnung $p = 2\,r$) gewisse Nachteile haben:

Satz 3.56 (Eigenschaften von IRKV)**.**

(a) Gauß-Verfahren und Lobatto-IIIA-, Lobatto-IIIB-Verfahren sind A-stabil, aber nicht L-stabil.

(b) Radau-II- und Lobatto-IIIC-Verfahren sind sogar L-stabil.

(c) Das implizite Euler-Verfahren ist damit (als einfachstes Radau-II-Verfahren) L-stabil.

Beweis: In Hairer, Wanner, 1996, Section IV.5:

(a) direkt.

(b) über Eigenschaften der Padé Approximation. ⇒ Für RKV ist $R(z)$ stets eine Approximation der $\exp(z)$ Funktion. Explizite RKV liefern ein Polynom für $R(z)$, implizite eine Rationale Funktion

$$R(z) = \frac{P_k(z)}{Q_j(z)}, \tag{3.43}$$

wobei P_k und Q_j Polynome vom Grad k bzw. j sind.

Die genannten Verfahren resultieren in einer Padé Approximation (3.43) mit $j > k$. Damit ist

$$\lim_{h\to\infty} R(h\,\lambda) = 0 \quad \forall\,\lambda \neq 0$$

impliziert. Also sind die Verfahren L-stabil.

(c) Für implizit Euler wissen wir $R(z) = 1/(1-z)$. Damit folgt sofort

$$\lim_{h\to\infty} R(h\,\lambda) = 0 \quad \forall\,\lambda \neq 0.$$

□

Die obigen Stabilitätsbegriffe beziehen sich alle auf Systeme bei denen ohne äußere Erregung die Lösungen abklingen (Eigenwerte der Jacobimatrix f_y sind alle $\in \mathbb{C}^-$). In der Mechanik sind dies typischerweise Systeme denen Energie verloren geht (z. B. durch Reibung oder Dämpfung). Eine andere Anforderung ergibt sich wenn Systeme durch Energieerhaltung charakterisiert sind. Dies behandeln wir im folgenden Abschnitt.

§ 3.5 Gittersteuerung durch eingebettete Runge-Kutta-Verfahren

Beobachtung: Äquidistante Zeitgitter sind oft ineffizient:

Beispiel 3.57 (Notwendigkeit der Schrittweitensteuerung)**.** Der Van-der-Pol-Oszillator[28] wird durch die Differentialgleichung

$$\dot{y}_1 = y_2$$
$$\dot{y}_2 = \mu\left(1 - y_1^2\right) y_2 - y_1$$

beschrieben. Zum Anfangswert $y_0 = (2,0)^\top$ und $\mu = 1000$ liefert die MATLAB-Routine `ode23s` bei Standardeinstellungen auf dem Intervall $t \in [0, 3000]$ die in Abbildung 3.11 dargestellte Lösung.

Ziel:

Anpassen der Schrittweiten h_k während des Laufes, d, h.

- so klein wie nötig (geforderte Genauigkeit),
- so groß wie möglich (geringe Rechenzeit).

Eine solche automatische Anpassung nennt man **Schrittweitensteuerung**.

Mit der Schrittweitensteuerung könnte man folgende Ziele verfolgen:

- Kontrolle des globalen Diskretisierungsfehlers:

$$\|e_h\|_{\infty,h} = \max_{t\in\mathcal{T}} \|y_h(t) - y(t)\| \overset{!}{\leq} E_{\max}$$

- Kontrolle des lokalen Diskretisierungsfehlers (Fehler pro Schritt):

$$\|h\,d(y(\cdot), t+h, h)\| \overset{!}{\leq} D_{\max}$$

[28] Computer-Demonstration: Van_der_Pol

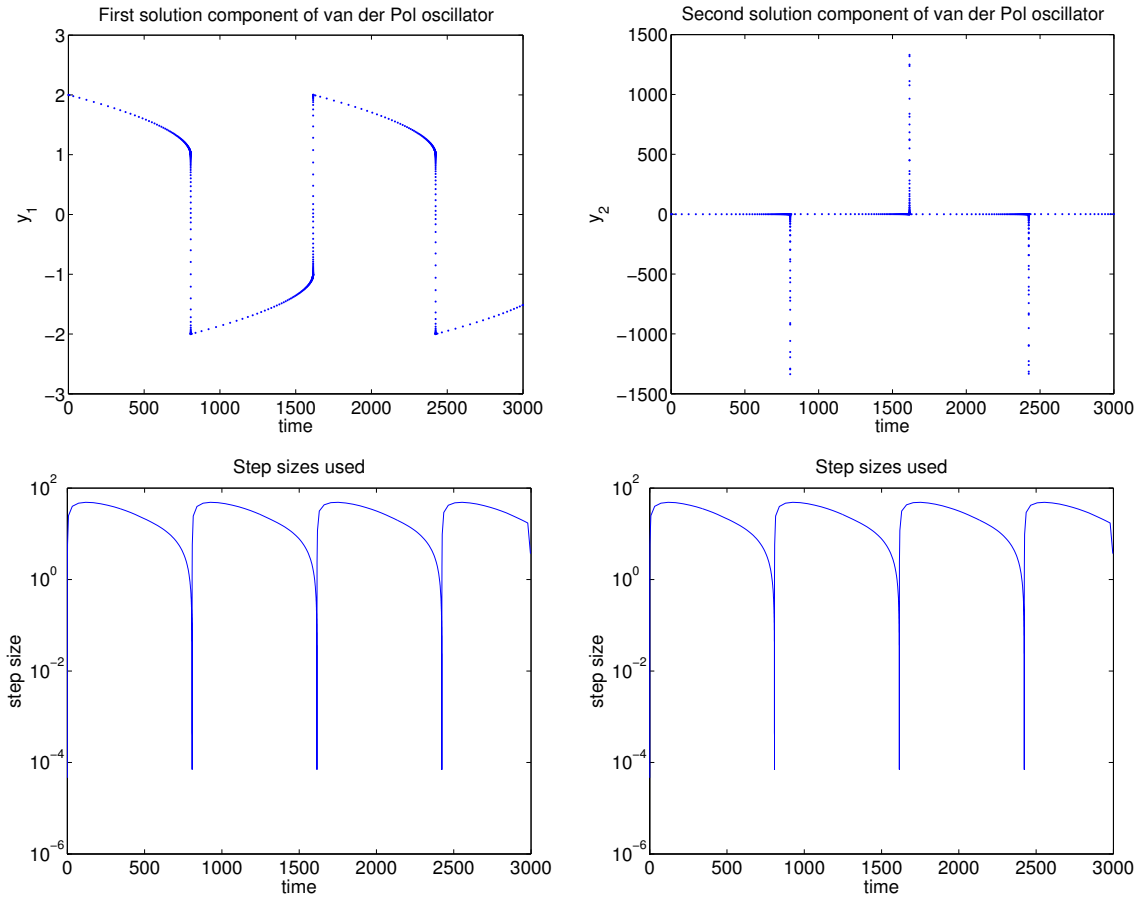


ABBILDUNG 3.11. Lösung des Van-der-Pol-Oszillators: Verlauf der Komponenten y_1 und y_2 (oben) und automatische Wahl der Schrittweiten durch `ode23s` (unten, zur besseren Illustration doppelt abgebildet, logarithmische Skala).

Wie bereits in Bemerkung 3.19 erwähnt, sind die lokalen Fehlerbeiträge nicht bekannt, sodass die *a priori* Fehlerschranke (3.19) nicht zur Kontrolle der globalen Fehlers verwendet werden kann. Wir konzentrieren uns zunächst darauf, den *lokalen Fehler a posteriori zu schätzen.*

Fehlerschätzung durch eingebettete Verfahren höherer Ordnung

Idee: Verwende zwei ESV der Ordnungen p und $p+1$ (Verfahrensfunktionen Φ und $\widehat{\Phi}$). Führe mit beiden Verfahren einen Schritt von (t_k, y_k) zur selben Schrittweite h aus. Betrachte die Näherung zum genaueren Verfahren als exakte Lösung und nutze diese, um den Fehler im Schritt der schlechteren Methode zu schätzen.

Es seien

$$y_{k+1} = y_k + h\,\Phi(t_k, h, y_k) \qquad \text{Ordnung } p \tag{3.44a}$$

$$\widehat{y}_{k+1} = y_k + h\,\widehat{\Phi}(t_k, h, y_k) \qquad \text{Ordnung } p+1 \tag{3.44b}$$

diese beiden Näherungen.

Ziel: Wähle die Schrittweite h, sodass gilt: $\|y(t_k + h) - y_{k+1}\| = \text{TOL}$.[29]

[29] Wir wollen also mit dem Schritt des *schlechteren* Verfahrens die Toleranz halten.

Wir ersetzen $y(t_k+h)$ durch die Näherungslösung $\widehat{y}_{k+1}$ des besseren Verfahrens und erhalten mit

$$\text{Est} := \|\widehat{y}_{k+1} - y_{k+1}\| \approx \|y(t_k+h) - y_{k+1}\| \qquad (**)$$

eine Schätzung des Fehlers für den aktuellen Schritt des schlechteren Verfahrens.[30]

Frage: Mit welcher Schrittweite h_* erreichen wir die Toleranz Tol? Wir machen den Ansatz

$$\|y(t_k+h) - y_{k+1}\| = \|h\, d(y_h(\cdot), t_k+h, h)\| = c\, h^{p+1}$$

und schätzen c mittels $(**)$ durch die vorhandenen Daten zur Testschrittweite h:

$$\overline{c} := \frac{\text{Est}}{h^{p+1}}.$$

Mit dem Ansatz

$$\text{Tol} = \overline{c}\, h_*^{p+1}$$

ergibt sich die Vorhersage, dass die Toleranz Tol mit der Schrittweite

$$h_* := \left(\frac{\text{Tol}}{\text{Est}}\right)^{\frac{1}{p+1}} h$$

erreicht wird. In der Praxis setzt man

$$h_* := \left(\frac{\rho\,\text{Tol}}{\text{Est}}\right)^{\frac{1}{p+1}} h$$

mit einem Sicherheitsfaktor $\rho \in (0,1]$ an.[31] Falls bereits $\text{Est} \le \text{Tol}$ war, also h hinreichend klein, wird der Schritt akzeptiert, ansonsten verworfen. In jedem Fall wird h_* neue Testschrittweite.[32]

In der Praxis kommen noch Begrenzungsfaktoren $\underline{\alpha}, \overline{\alpha}$ hinzu:

$$h_* := h\ \text{proj}_{[\underline{\alpha},\overline{\alpha}]}\left(\left[\frac{\rho\,\text{Tol}}{\text{Est}}\right]^{\frac{1}{p+1}}\right), \qquad (3.45)$$

wobei $\text{proj}_{[\underline{\alpha},\overline{\alpha}]}(x) = \min\{\overline{\alpha}, \max\{\underline{\alpha}, x\}\}$ die Projektion auf das Intervall $[\underline{\alpha}, \overline{\alpha}]$ ist.

Es ergibt sich also folgende Strategie zur Schrittweitensteuerung für gegebene $\underline{\alpha} < 1$, $\overline{\alpha} > 1$, $\rho \in (0,1]$, $\text{Tol} > 0$, eine erste Startschrittweite $h > 0$ und Endzeit $T > 0$:

[30] Dies wird damit gerechtfertigt, dass das bessere Verfahren eine höhere Fehlerordnung hat.

[31] Man zielt also mit h_* auf $\rho\,\text{Tol}$, ist aber dann bereits mit dem Erreichen von Tol zufrieden.

[32] Dies entspricht einer adaptiven Gittergenerierung „on the fly“, die natürlich nur funktioniert, weil wir bei AWP für ODEs eine sequentielle Berechnung haben.

Algorithmus 3.58 (Schrittweitensteuerung durch eingebettete Verfahren höherer Ordnung)**.**

Eingabe: Parameter $\underline{\alpha} < 1$, $\overline{\alpha} > 1$, $\rho \in (0, 1]$, TOL > 0, erste Testschrittweite $h > 0$, Endzeit $T > 0$
Ausgabe: Näherungslösung $y_h(\cdot)$ des (AWP) auf Gitter $\mathcal{T} = \{t_k\}$

Setze $k := 0$ und $t_k := 0$
while $t_k < T$ **do**
  akzeptiert := FALSE
  repeat
    Berechne y_{k+1} und $\widehat{y}_{k+1}$ aus (3.44)
    Setze EST := $\|y_{k+1} - \widehat{y}_{k+1}\|$
    Berechne h_* aus (3.45)
    if EST $\leq$ TOL **then** {Schritt wird akzeptiert}
      Verwende als Näherung y_{k+1} oder $\widehat{y}_{k+1}$
      Setze $t_{k+1} := t_k + h$ und $k := k + 1$ und akzeptiert := TRUE
      Setze $h := \min\{h_*, T - t_k\}$
    else {Schritt wird verworfen und die Schrittweite reduziert}
      Setze $h := h_*$
    end if
  until akzeptiert
end while

Bemerkung 3.59 (zu Algorithmus 3.58)**.** Die Schrittweitensteuerung versucht, den Fehler in jedem Schritt nahe TOL zu halten. In der Praxis wählt man häufig

$$\text{TOL} = \text{ATOL} + \text{RTOL} \cdot \|\widehat{y}_h(t_i + h)\| \tag{3.46}$$

mit absoluten und relativen Toleranzen.

Um den Aufwand für die Ausführung jedes Integrationsschrittes (3.44) mit *beiden* Verfahren gering zu halten, verwendet man sogenannte **eingebettete RKV** oder **Runge-Kutta-Fehlberg-Verfahren**. Typisch ist folgende Konstellation:

- Das schlechtere Verfahren Φ hat r Stufen und die Ordnung p.
- Das bessere Verfahren $\widehat{\Phi}$ hat $\widehat{r} = r + 1$ Stufen und die Ordnung $\widehat{p} = p + 1$.
- Für die ersten r Steigungen $\widehat{f}_{jk}$ von $\widehat{\Phi}$ gilt: $f_{jk} = \widehat{f}_{jk}$, $j = 1, \ldots, r$.

Somit „kostet“ $\widehat{\Phi}$ nur eine zusätzliche Stufen-Berechnung (eine f-Auswertung bei ERKV). Die Gewichte γ_j und $\widehat{\gamma}_j$ müssen wegen der Konsistenzbedingung (3.24b) natürlich unterschiedlich sein.

Beispiel 3.60 (Eingebettete RKV)**.** Einige Beispiele eingebetteter expliziter RKV sind:

(a) **Runge-Kutta-Fehlberg** RKF2(3):

0			
1	1		
1/2	1/4	1/4	
c	1/2	1/2	
$\widehat{c}$	1/6	1/6	4/6

Hier sind $(r,p)=(2,2)$ und $(\widehat{r},\widehat{p})=(3,3)$.

Beachte: Das Verfahren 2. Ordnung ist das Verfahren von Heun, vgl. Beispiel 3.24. Die α_i müssen nicht aufsteigend sein!

(b) **Runge-Kutta-Fehlberg** RKF4(5):

$$
\begin{array}{c|ccccc}
0 & & & & & \\
\frac{1}{4} & \frac{1}{4} & & & & \\
\frac{3}{8} & \frac{3}{32} & \frac{9}{32} & & & \\
\frac{12}{13} & \frac{1932}{2197} & -\frac{7200}{2197} & \frac{7296}{2197} & & \\
1 & \frac{439}{216} & -8 & \frac{3680}{513} & -\frac{845}{4104} & \\
\frac{1}{2} & -\frac{8}{27} & 2 & -\frac{3544}{2565} & \frac{1859}{4104} & -\frac{11}{40} \\
\hline
c & \frac{25}{216} & 0 & \frac{1408}{2565} & \frac{2197}{4104} & -\frac{1}{5} \\
\hline
\widehat{c} & \frac{16}{135} & 0 & \frac{6656}{12825} & \frac{28561}{56430} & -\frac{9}{50} \quad \frac{2}{55}
\end{array}
$$

Hier sind $(r,p)=(5,4)$ und $(\widehat{r},\widehat{p})=(6,5)$.

(c) **Dormand-Prince** DOPRI5(4) ist die Basis für `ode45` in Matlab:

$$
\begin{array}{c|ccccccc}
0 & & & & & & & \\
\frac{1}{5} & \frac{1}{5} & & & & & & \\
\frac{3}{10} & \frac{3}{40} & \frac{9}{40} & & & & & \\
\frac{4}{5} & \frac{44}{45} & -\frac{56}{15} & \frac{32}{9} & & & & \\
\frac{8}{9} & \frac{19372}{6561} & -\frac{25360}{2187} & \frac{64448}{6561} & -\frac{212}{729} & & & \\
1 & \frac{9017}{3168} & -\frac{355}{33} & \frac{46732}{5247} & \frac{49}{176} & -\frac{5103}{18656} & & \\
1 & \frac{35}{384} & 0 & \frac{500}{1113} & \frac{125}{192} & -\frac{2187}{6748} & \frac{11}{84} & \\
\hline
\widehat{c} & \frac{35}{384} & 0 & \frac{500}{1113} & \frac{125}{192} & -\frac{2187}{6784} & \frac{11}{84} & \\
\hline
c & \frac{5179}{57600} & 0 & \frac{7571}{16695} & \frac{393}{640} & -\frac{92097}{339200} & \frac{187}{2100} & \frac{1}{40}
\end{array}
$$

Dieses Verfahren ist etwas anders konstruiert als die beiden obigen:

- Das Verfahren niedrigerer Ordnung $p = 4$ benutzt $r = 7$ Stufen, das Verfahren höherer Ordnung $\widehat{p} = 5$ dagegen nur $\widehat{r} = 6$.
- Das DOPRI-Paar ist dafür optimiert eine gute Fehler-Konstante beim Verfahren höherer Ordnung zu erzielen. Das Verfahren der niedrigeren Ordnung wird meist nur zur Schrittweitensteuerung verwendet, also wird $y_{k+1} = \widehat{y}_{k+1}$ tatsächlich als Wert für den neuen Zeitschritt verwendet und der andere Wert wird verworfen, (sog. **lokale Extrapolation**).[33] Hairer, Nørsett, Wanner, 1993, § II.4.
- Es erscheint zunächst als Nachteil, für eine Näherung der Ordnung $p = 4$, $\widehat{p} = 5$ insgesamt $r = 7$ Stufen zu verwenden (da $\widehat{p} = 5$ auch mit $r = 6$ erreichbar ist). Jedoch ist das Verfahren mit dem „**First same as last (FSAL)-Trick**“ konstruiert:

$$\begin{aligned} f_{7k} &= f(t_k + h, y_k + h \sum_{i=1}^{6} \beta_{7i} f_{ik}) \\ &= f(t_k + h, y_k + h \sum_{i=1}^{6} \widehat{\gamma}_i f_{ik}) \qquad (\text{da } \widehat{\gamma}_i = \beta_{7i}) \\ &= f(t_{k+1}, y_{k+1}). \end{aligned}$$

Es kann also die Steigung f_{7k} eines (akzeptierten) Schrittes als Steigung f_{1k} des nächsten Schrittes wiederverwendet werden, sodass sich i. W. wieder nur 6 f-Auswertungen pro Schritt ergeben.

Bemerkung 3.61 (Einschrittverfahren in MATLAB)**.** In MATLAB sind folgende Einschrittverfahren (alle mit Schrittweitensteuerung) implementiert:

- `ode23` ist ein explizites RKV-Paar der Ordnung (2,3).
- `ode45` ist das explizite RKV-Paar DOPRI5(4) der Ordnung (5,4).
- `ode23s` ist eine modifizierte Rosenbrock-Formel (LIRK-Verfahren) der Ordnung 2 und daher für steife Dgl-Systeme geeignet.

Beispielaufruf:

```
% Zeitintervall, ggf. mit Zwischenpunkten
tspan = [0,3000];

% Anfangswert
y0 = [2;0];

% Optionen setzen
options = odeset('AbsTol', 1e-6, 'Stats', 'on');

% eigentlicher Aufruf des Integrators
```

[33] Der Fehler wird also eigentlich für einen Schritt des anderen Verfahrens geschätzt.

```
[t,y] = ode23s(@(t,y)RHS_Van_der_Pol(t,y,1000), tspan, ↪
   y0, options);
```

Beispiel einer rechten Seite:

```
function val = RHS_Van_der_Pol(t,y,mu)

% Initialize the output to a column vector
val = zeros(size(y));

% Evaluate the rhs for the van-der-Pol oscillator
val(1) = y(2);
val(2) = mu * (1-y(1)^2) * y(2) - y(1);
```

Beachte: `@(t,y)RHS_Van_der_Pol(t,y,1000)` erzeugt zur Laufzeit aus `RHS_Van_der_Pol` eine sogenannte anonyme Funktion mit den Eingabevariablen `(t,y)`, indem `mu = 1000` eingesetzt wird.

§ 3.6 Einschrittverfahren für Dgl zweiter Ordnung

Aufgrund des zweiten Newtonschen Gesetzes,

$$F = m\ddot{x},$$

kommen **(AWP)** mit Dgl zweiter Ordnung

$$\begin{aligned} y'' &= f(t,\ y,\ y'), \\ \text{mit AB} \qquad y(0) &= y_a, \\ y'(0) &= y_b, \end{aligned} \tag{3.47}$$

$f : [0,T] \times \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}^n$, besonders häufig in Anwendungen vor.

Umgewandelt in ein System erster Ordnung (vgl. Sheet 1, Exercise 1) erhält man

$$\begin{aligned} z' = \begin{pmatrix} y \\ y' \end{pmatrix}' &= \begin{pmatrix} y' \\ f(t,\ y,\ y') \end{pmatrix}, \\ z(0) = \begin{pmatrix} y(0) \\ y'(0) \end{pmatrix} &= \begin{pmatrix} y_a \\ y_b \end{pmatrix}. \end{aligned}$$

Wendet man hierauf nun ein (explizites oder implizites) RKV mit Butcher-Diagramm $\begin{array}{c|c} a & B \\ \hline & c^\top \end{array}$ an, so ergibt sich durch analoge Aufteilung der Steigungen in $(f_{jk}, f'_{jk})^\top$

$$f_{jk} = y'_k + h \sum_{i=1}^{r} \beta_{ji}\, f'_{ik}, \tag{3.48a}$$

$$f'_{jk} = f(t_k + \alpha_j\, h,\ y_k + h \sum_{i=1}^{r} \beta_{ji}\, f_{ik},\ y'_k + h \sum_{i=1}^{r} \beta_{ji}\, f'_{ik}), \tag{3.48b}$$

für $j = 1, \ldots, r$, und

$$y_{k+1} = y_k + h_i \sum_{j=1}^{r} \gamma_j\, f_{jk}, \qquad y'_{k+1} = y'_k + h_i \sum_{j=1}^{r} \gamma_j\, f'_{jk}. \tag{3.48c}$$

Setzt man (3.48a) in (3.48b) und (3.48c) ein, so erhält man unter Verwendung der Autonomie-Invarianz (3.24)

$$f'_{jk} = f(t_k + \alpha_j\, h,\ y_k + \alpha_j\, h y'_k + h^2 \sum_{i=1}^{r} \overline{\beta}_{ji}\, f'_{ik},\ y'_k + h \sum_{i=1}^{r} \beta_{ji}\, f'_{ik}), \tag{3.49a}$$

$$y_{k+1} = y_k + h_k\, y'_k + h_k^2 \sum_{j=1}^{r} \overline{\gamma}_j\, f'_{jk}, \qquad y'_{k+1} = y'_k + h_k \sum_{j=1}^{r} \gamma_j\, f'_{jk}, \tag{3.49b}$$

wobei

$$\overline{\beta}_{j\ell} := \sum_{k=1}^{s} \beta_{jk}\beta_{k\ell} \qquad\qquad \overline{\gamma}_j := \sum_{k=1}^{s} \gamma_k \beta_{kj},$$

bzw. in Matrixschreibweise

$$[\overline{\beta}_{j\ell}] =: \overline{B} = B^2 \qquad\qquad [\overline{\gamma}_j] =: \overline{c}^\top = c^\top B. \tag{3.50}$$

D. h., man muss die f_{jk} nicht speichern, was bei großem n bereits eine wesentliche Einsparung ist.

Die Beziehung (3.50) stellt dabei lediglich eine Möglichkeit zur Herleitung dar. Lässt man diese Forderung fallen, so erhält man folgende Definition.

Definition 3.62 (Runge-Kutta-Nyström-Verfahren)**.** Verfahren (3.49) mit Koeffizienten

$$\begin{array}{c|c|c} a & \overline{B} & B \\ \hline & \overline{c}^\top & c^\top \end{array}$$

heißen **Runge-Kutta-Nyström-Verfahren (RKNV)**.

Die Ordnungsbedingungen lassen sich auf RKNV verallgemeinern und auch eingebettete Verfahren dieses Typs sind bekannt, siehe z. B. Hairer, Nørsett, Wanner, 1993, Section II.14.

Eine nochmals wesentliche Einsparung erhält man, wenn f nicht von y' abhängt, also

$$y'' = f(t,\ y).$$

Dann wird B gar nicht mehr benötigt, und man kann besonders effiziente Verfahren herleiten, welche vergleichsweise sehr wenige f-Auswertungen benötigen.

Beispiel 3.63 (Einige Runge-Kutta-Nyström-Verfahren)**.**

(a) Das klassische Verfahren von Nyström (1925, Ordnung $p = 4$) .

$$\begin{array}{c|cccc|cccc} 0 & & & & & & & & \\ \frac{1}{2} & \frac{1}{8} & & & & \frac{1}{2} & & & \\ \frac{1}{2} & \frac{1}{8} & 0 & & & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & \frac{1}{2} & & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} \end{array} \qquad \begin{array}{c} \alpha_j \\ \\ \\ \\ \\ \overline{\beta}_{j\ell} \\ \\ \\ \\ \overline{\gamma}_j \end{array} \qquad \begin{array}{c} \\ \\ \\ \\ \\ \beta_{j\ell} \\ \\ \\ \\ \gamma_j \end{array}$$

Im Falle $f = f(t, y)$ vereinfacht sich dies zu:

$$\begin{array}{c|ccc|ccc} & & & & \overline{\beta}_{j\ell} & & \\ 0 & & & & & & \\ \alpha_j \quad \frac{1}{2} & \frac{1}{8} & & & & & \\ 1 & 0 & \frac{1}{2} & & & & \\ \hline & \frac{1}{6} & \frac{2}{6} & 0 & \frac{1}{6} & \frac{4}{6} & \frac{1}{6} \\ & & \overline{\gamma}_j & & & \gamma_j & \end{array}$$

Man erhält also Ordnung $p = 4$ mit nur drei f-Auswertungen je Schritt!

(b) Die Newmark-Verfahren, mit einem Parameter $\theta \in [0, 1]$:

$$\begin{array}{c|cc|cc} & \overline{\beta}_{j\ell} & & \beta_{j\ell} & \\ 0 & & & & \\ \alpha_j \quad 1 & \frac{1}{2} - \theta & \theta & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} - \theta & \theta & \frac{1}{2} & \frac{1}{2} \\ & \overline{\gamma}_j & & \gamma_j & \end{array}$$

Mit $\theta = 0$ erhält man als Spezialfall das **Störmer-Verlet**-Verfahren (engl. auch **leap-frog method**),

$$\begin{array}{c|cc|cc} & \overline{\beta}_{j\ell} & & \beta_{j\ell} & \\ 0 & & & & \\ \alpha_j \quad 1 & \frac{1}{2} & & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & 0 & \frac{1}{2} & \frac{1}{2} \\ & \overline{\gamma}_j & & \gamma_j & \end{array}$$

ein Verfahren der Ordnung $p = 2$. Für $y'' = f(t, y)$ ist dies sogar ein explizites Verfahren.

§ 4 Mehrschrittverfahren

§ 4.1 Einleitung und Grundbegriffe

Idee: Zum Erreichen höherer Konsistenzordnungen bei ESV sind viele Stufen r und deshalb viele f-Auswertungen erforderlich (teuer). Nutze stattdessen die vorhandene Information aus den zurückliegenden Näherungen y_k, y_{k-1}, y_{k-2}, ...

Wir machen den Ansatz [34]

$$\begin{aligned} &\alpha_r\, y_{k+r} + \alpha_{r-1}\, y_{k+r-1} + \cdots + \alpha_1\, y_{k+1} + \alpha_0\, y_k \\ &= \sum_{m=0}^{r} \alpha_m\, y_{k+m} = h_{k+r-1}\, \Phi(t_k, h_{k+r-1}, y_k, y_{k+1}, \ldots, y_{k+r-1}) \end{aligned} \tag{4.1}$$

mit $\alpha_r \neq 0$ zur Berechnung von y_{k+r} aus den $r \geq 1$ (Schrittzahl) zurückliegenden Näherungen y_k, y_{k+1}, ..., y_{k+r-1}, siehe Abbildung 4.1.

[34] y_{k+r} steht nicht rechts als Argument, da auch bei impliziten Verfahren eine solche Funktion Φ (ohne Argument y_{i+r}) (implizit) definiert ist.

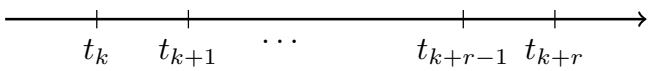


ABBILDUNG 4.1. Aktiver Ausschnitt des Zeitgitters bei Mehrschrittverfahren.

Definition 4.1 (Mehrschrittverfahren)**.**

(a) Ein Verfahren der Bauart (4.1) heißt ein **Mehrschrittverfahren** (MSV), genauer: ein r**-Schritt-Verfahren**. Φ heißt wieder die **Inkrementfunktion** oder **Verfahrensfunktion** des Verfahrens.

(b) Ein MSV heißt **explizit** (EMSV), wenn die Berechnung von y_{k+r} ohne die Auflösung linearer oder nichtlinearer Gleichungssysteme auskommt, ansonsten **implizit** (IMSV).

(c) Ein MSV heißt **linear**, wenn gilt:

$$\sum_{m=0}^{r} \alpha_m \, y_{k+m} = h_{k+r-1} \sum_{m=0}^{r} \beta_m \, f(t_{k+m}, y_{k+m}). \tag{4.2}$$

Beachte: Die Werte $f(t_k, y_k), \ldots, f(t_{k+r-1}, y_{k+r-1})$ sind schon aus dem vorhergehenden Schritt bekannt.

Bemerkung 4.2 (zu Mehrschrittverfahren)**.**

(a) ESV führen i. A. auf *nichtlineare*[35] MSV mit $r = 1$, $\alpha_1 = 1$ und $\alpha_0 = -1$, z. B. das verbesserte Euler-Verfahren [36]:

$$-y_k + y_{k+1} = h \, f\big(t_k + \tfrac{h}{2}, y_k + \tfrac{h}{2} f(t_k, y_k)\big).$$

(b) Ein lineares MSV ist genau dann explizit, wenn $\beta_r = 0$ gilt.

(c) MSV benötigen eine **Anlaufphase**, um die ersten Näherungen y_0, y_1, …, y_{r-1} zu bestimmen, z. B. durch ESV oder MSV mit niedrigerer Schrittanzahl r.

Im Folgenden betrachten wir *nur lineare MSV* auf *äquidistanten Gittern*.

Beispiel 4.3 (Explizite MSV)**.** Es sei $t_k = k\,h$ (äquidistantes Gitter). Wir betrachten die Darstellung

$$y(t_{k+r}) - y(t_{k+r-1}) = \int_{t_{k+r-1}}^{t_{k+r}} f(t, y(t)) \, \mathrm{d}t \tag{4.3}$$

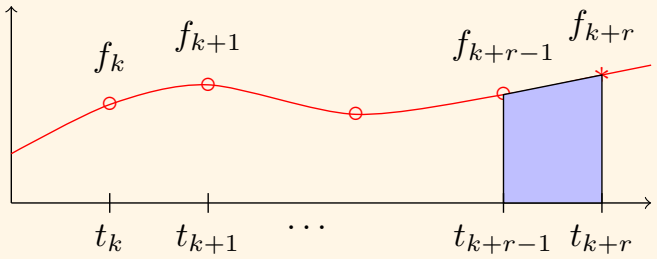
und ersetzen f durch das interpolierende Polynom P vom Grad $r-1$ (mit Werten in $\mathbb{R}^n$) zu den Daten

$$(t_j, f_j) \in \mathbb{R} \times \mathbb{R}^n, \quad j = k, k+1, \ldots, k+r-1$$

mit der Abkürzung $f_j := f(t_j, y_j)$. [37]

[35] Die Nichtlinearität kommt bei RKV von der Schachtelung der f-Auswertungen und kann nur für $s \geq 2$ auftreten.

[36] In Sheet 5, Exercise 14 werden verschiedene ESV als MSV gedeutet.



Wir konstruieren die Interpolierende mit der Lagrange-Basis

$$f(t,y(t)) \approx P(t) = \sum_{m=0}^{r-1} f_{k+m} L_{k+m}(t), \quad L_{k+m}(t) := \prod_{\substack{\ell=0 \\ \ell\neq m}}^{r-1} \frac{t - t_{k+\ell}}{t_{k+m} - t_{k+\ell}}. \tag{4.4}$$

Einsetzen von (4.4) in (4.3) ergibt die Vorschrift

$$\begin{aligned} y_{k+r} &= y_{k+r-1} + \int_{t_{k+r-1}}^{t_{k+r}} \sum_{m=0}^{r-1} f_{k+m}\, L_{k+m}(t)\mathrm{d}t \\ &= y_{k+r-1} + \sum_{m=0}^{r-1} f_{k+m} \int_{t_{k+r-1}}^{t_{k+r}} L_{k+m}(t)\mathrm{d}t \\ &= y_{k+r-1} + h \sum_{m=0}^{r-1} f_{k+m} \underbrace{\int_0^1 L_{k+m}(t_{k+r-1} + h\, s)\mathrm{d}s}_{\beta_m}. \end{aligned} \tag{4.5}$$

Nachrechnen liefert

$$\beta_m = \int_0^1 \prod_{\substack{\ell=0 \\ \ell\neq m}}^{r-1} \frac{t_{k+r-1} + h\, s - t_{k+\ell}}{t_{k+m} - t_{k+\ell}} ds = \int_0^1 \prod_{\substack{\ell=0 \\ \ell\neq m}}^{r-1} \frac{r - 1 - \ell + s}{m - \ell} ds.$$

MSV von diesem Typ sind explizit und heißen **explizite Adamsformeln** oder **Adams-Bashforth-Formeln** (kurz: AB-Formeln). [38]

$$\begin{aligned} r = 1 \quad &\Rightarrow \quad y_{k+1} - y_k = h f_k \qquad \text{(expliziter Euler)} \\ r = 2 \quad &\Rightarrow \quad y_{k+2} - y_{k+1} = \frac{h}{2}\left(3f_{k+1} - f_k\right) \\ r = 3 \quad &\Rightarrow \quad y_{k+3} - y_{k+2} = \frac{h}{12}\left(23f_{k+2} - 16f_{k+1} + 5f_k\right) \end{aligned}$$

Beispiel 4.4 (Implizite MSV)**.** Das Polynom P^* (nun vom Grad r) interpoliere zusätzlich bei t_{k+r}, also

$$(t_j, f_j) \in \mathbb{R} \times \mathbb{R}^n, \quad j = k,\, k+1, \ldots,\, k+r.$$

[37] In Sheet 5, Exercise 16 werden diese Methoden, sowie auch die impliziten Adamsformeln und BDF-Methoden, für $r = 2, 3$ nachgerechnet.

[38] Schöne Darstellung mit Lagrange-Polynom auch in http://www.mathematik.uni-stuttgart.de/studium/infomat/Numerische-Mathematik-II-SS11/Matlab/Simeon.pdf

Dann gilt statt (4.4)

$$f(t,y(t)) \approx P^*(s) = \sum_{m=0}^{r} f_{k+m} L_{k+m}(t), \quad L_{k+m}(t) := \prod_{\substack{\ell=0 \\ \ell\neq m}}^{r} \frac{t - t_{k+\ell}}{t_{k+m} - t_{k+\ell}}. \tag{4.6}$$

Analog zu (4.5) folgt dann

$$y_{k+r} = y_{k+r-1} + h \sum_{m=0}^{r} f_{k+m} \underbrace{\int_0^1 L_{k+m}(t_{k+r-1} + h\,s)\mathrm{d}s}_{=\beta_m} \tag{4.7}$$

Ausrechnen der β_m ergibt

$$\beta_m = \int_0^1 \prod_{\substack{\ell=0 \\ \ell\neq m}}^{r} \frac{r - 1 - \ell + s}{m - \ell}.$$

MSV von diesem Typ sind implizit, da $f_{k+r} := f(t_{k+r}, y_{k+r})$ die gesuchte Größe y_{k+r} verwendet.

Sie heißen **implizite Adams-Formeln** oder **Adams-Moulton-Formeln** (kurz: AM-Formeln).

$$\begin{aligned}
&\Big[r = 0 \quad \Rightarrow \quad y_k - y_{k-1} = h f_k \qquad \text{(impliziter Euler)}\Big] \\
&r = 1 \quad \Rightarrow \quad y_{k+1} - y_k = \frac{h}{2}\big(f_{k+1} + f_k\big) \quad \text{(implizite Trapezregel)} \\
&r = 2 \quad \Rightarrow \quad y_{k+2} - y_{k+1} = \frac{h}{12}\big(5f_{k+2} + 8f_{k+1} - f_k\big) \\
&r = 3 \quad \Rightarrow \quad y_{k+3} - y_{k+2} = \frac{h}{24}\big(9f_{k+3} + 19f_{k+2} - 5f_{k+1} + f_k\big)
\end{aligned}$$

39

Beispiel 4.5. Allgemeiner als AB- und AM-Formeln sind (d,ℓ)**-Verfahren**: Interpoliere die Daten

$$(t_j, f_j) \in \mathbb{R} \times \mathbb{R}^n, \quad j = k,\, k+1, \ldots,\, k+d$$

und integriere das Interpolationspolynom vom Polynomgrad d im Intervall $[t_{k+d-\ell}, t_{k+r}]$ der Länge $(r - d + \ell)\,h$:

$d = r-1,\ \ell = 0 \rightsquigarrow$ Adams-Bashforth-Verfahren (explizit), Länge h

$d = r-1,\ \ell = 1 \rightsquigarrow$ Nyström-Verfahren (explizit), Länge $2h$

$d = r,\ \ell = 1 \rightsquigarrow$ Adams-Moulton-Verfahren (implizit), Länge h

$d = r,\ \ell = 2 \rightsquigarrow$ Milne-Simpson-Verfahren (implizit), Länge $2h$.

Beispiel 4.6. Bei den obigen Ansätzen wurden *Funktionswerte* von f interpoliert. Bei **BDF-Verfahren** (*backward differentiation formulae*) dagegen interpolieren wir die Daten

$$(t_j, y_j) \in \mathbb{R} \times \mathbb{R}^n, \quad j = k,\, k+1, \ldots,\, k+r,$$

[39] In Sheet 6, Exercise 18 wird die Konvergenz einer Fixpunktiteration zur Realisierung dieses Verfahrens untersucht.

also den aktuellen (noch zu bestimmenden) und die zurückliegenden *Näherungswerte* y_j, durch ein Polynom Q vom Grad r. Es gilt die Darstellung

$$y(t) \approx Q(t) = \sum_{m=0}^{r} y_{k+m} L_{k+m}(t), \quad L_{k+m}(t) := \prod_{\substack{\ell=0 \\ \ell \neq m}}^{r} \frac{t - t_{k+\ell}}{t_{k+m} - t_{k+\ell}}.$$

Ansatz: Bestimme den aktuellen Näherungswert $y_{k+r} = Q(t_{k+r})$ so, dass Q die Dgl im Punkt (t_{k+r}, y_{k+r}) erfüllt, d. h.,

$$\begin{aligned} f(t_{k+r}, y_{k+r}) &\stackrel{!}{=} Q'(t_{k+r}) \\ &= \sum_{m=0}^{r} y_{k+m} \frac{\mathrm{d}}{\mathrm{d}t} \left[\prod_{\substack{\ell=0 \\ \ell \neq m}}^{r} \frac{t - t_{k+\ell}}{t_{k+m} - t_{k+\ell}} \right]_{t=t_{k+r}} \\ &= \sum_{m=0}^{r} y_{k+m} \sum_{\substack{z=0 \\ z \neq m}}^{r} \frac{1}{t_{k+m} - t_{k+z}} \prod_{\substack{\ell=0 \\ \ell \neq m, \ell \neq z}}^{r} \frac{t_{k+r} - t_{k+\ell}}{t_{k+m} - t_{k+\ell}} \\ &= \sum_{m=0}^{r} y_{k+m} \sum_{\substack{z=0 \\ z \neq m}}^{r} \frac{1}{h\,(m - z)} \prod_{\substack{\ell=0 \\ \ell \neq m, \ell \neq z}}^{r} \frac{r - \ell}{m - \ell} \end{aligned}$$

Allgemein ergibt sich die Gestalt

$$\sum_{m=0}^{r} \alpha_m y_{k+m} = h\, f_{k+r} \tag{4.8}$$

mit[40]

$$\alpha_m = \sum_{\substack{z=0 \\ z \neq m}}^{r} \frac{1}{m - z} \prod_{\substack{\ell=0 \\ \ell \neq m, \ell \neq z}}^{r} \frac{r - \ell}{m - \ell}.$$

MSV von diesem Typ sind implizit und heißen **BDF-Verfahren**. [41]

$$\begin{aligned} r = 1 \quad &\Rightarrow \quad \text{(impliziter Euler)} \qquad y_{k+1} - y_k = h\, f_{k+1} \\ r = 2 \quad &\Rightarrow \quad \frac{1}{2}\big(3y_{k+2} - 4y_{k+1} + y_k\big) = h\, f_{k+2} \\ r = 3 \quad &\Rightarrow \quad \frac{1}{6}\big(11y_{k+3} - 18y_{k+2} + 9y_{k+1} - 2y_k\big) = h\, f_{k+3} \end{aligned}$$

Definition 4.7 (Konsistenzfehler, Konsistenzordnung, vgl. Definition 3.9)**.** Gegeben sei ein lineares MSV (4.2). Es sei $y(\cdot)$ die Lösung der Dgl $y'(t) = f(t, y(t))$ auf dem Intervall $[0, T]$.

(a) Die Größe

$$d(y(\cdot), t + r\,h, h) := \frac{1}{h} \left[\sum_{m=0}^{r} \alpha_m\, y(t + m\,h) - h \sum_{m=0}^{r} \beta_m\, f(t + m\,h, y(t + m\,h)) \right] \tag{4.9}$$

[40] Für folgende Formel gibt es das Maple-Skript `BDF_Koeffizienten.mw`

[41] In Kunkel, Mehrmann, 1994 findet man noch weitere Verfahren mit $r \geq 4$ Schritten.

heißt der **Konsistenzfehler** des MSV (4.2) an der Stelle t zur Schrittweite h.

(b) Das MSV (4.2) heißt **konsistent** mit dem (**AWP**), falls gilt:

$$\lim_{h \searrow 0} \sup_{t \in [0, T - r\,h]} \| d(y(\cdot), t + r\,h, h) \| = 0. \tag{4.10}$$

(c) Das MSV besitzt die **Konsistenzordnung** p für das (**AWP**), falls

$$\sup_{t \in [0, T - r\,h]} \| d(y(\cdot), t + r\,h, h) \| \leq C\,h^p \tag{4.11}$$

gilt mit einer von h unabhängigen Konstanten C, also kurz:

$$\sup_{t \in [0, T - r\,h]} \| d(y(\cdot), t + r\,h, h) \| = \mathcal{O}(h^p).$$

Bemerkung 4.8 (Lokaler Diskretisierungsfehler)**.** Im Unterschied zu ESV kann der Konsistenzfehler i. A. nur bis auf Restterme und eine Konstante als **lokaler Diskretisierungsfehler** (Fehler in einem Schritt) interpretiert werden, vgl. Bemerkung 3.12, denn: Unter Annahme exakter Startwerte

$$y_0 = y(0), \quad \ldots, \quad y_{r-1} = y(t_{r-1})$$

gilt:

$$\begin{aligned} h\,d(y(\cdot), t_r, h) &= \sum_{m=0}^{r} \alpha_m\, y(t_m) - h \sum_{m=0}^{r} \beta_m\, f(t_m, y(t_m)) \\ &= \underbrace{\sum_{m=0}^{r} \alpha_m\, y_m - h \sum_{m=0}^{r} \beta_m\, f(t_m, y_m)}_{=0} \\ &\quad + \alpha_r \left(y(t_r) - y_r \right) - h\,\beta_r \left[f(t_r, y(t_r)) - f(t_r, y_r) \right] \end{aligned}$$

$$\Rightarrow \quad \| h\,d(y(\cdot), t_r, h) \| \leq (|\alpha_r| + h\,|\beta_r|\,L)\, \| y_r - y(t_r) \|,$$

mit Lipschitzkonstante L.

Für explizite MSV ($\beta_r = 0$) mit $\alpha_r = 1$ ist sogar wieder genau $h\,d(y(\cdot), t_r, h) = y(t_r) - y_r$.

Die Konsistenzordnung eines linearen MSV kann man gut mit Hilfe der assoziierten Polynome beschreiben.

Definition 4.9 (Assoziierte Polynome)**.** Einem linearen MSV (4.2) ordnet man zu:

(a) das **erste assoziierte Polynom** (oder **erstes charakteristische Polynom**)

$$\varrho(\mu) := \alpha_r\,\mu^r + \alpha_{r-1}\,\mu^{r-1} + \cdots + \alpha_1\,\mu^1 + \alpha_0 \tag{4.12a}$$

(b) das **zweite assoziierte Polynom** (oder **zweites charakteristische Polynom**)

$$\sigma(\mu) := \beta_r\,\mu^r + \beta_{r-1}\,\mu^{r-1} + \cdots + \beta_1\,\mu^1 + \beta_0. \tag{4.12b}$$

Beispiel 4.10 (Assoziierte Polynome)**.**

(a) Für alle expliziten und impliziten Adams-Formeln (4.5) und (4.7) gilt

$$\varrho(\mu) = \mu^r - \mu^{r-1}.$$

Für explizit Adams mit $r = 3$,

$$y_{k+3} - y_{k+2} = \frac{h}{12}\big(23f_{k+2} - 16f_{k+1} + 5f_k\big),$$

erhält man

$$\sigma(\mu) = \frac{1}{12}\big(23\mu^2 - 16\mu + 5\big)$$

und für implizit Adams mit $r = 3$,

$$y_{k+3} - y_{k+2} = \frac{h}{24}\big(9f_{k+3} + 19f_{k+2} - 5f_{k+1} + f_k\big),$$
$$\sigma(\mu) = \frac{1}{24}\big(9\mu^3 + 19\mu^2 - 5\mu + 1\big).$$

(b) Für alle BDF-Formeln (4.8) gilt

$$\sigma(\mu) = \mu^r.$$

Für BDF mit $r = 3$,

$$\frac{1}{6}\big(11y_{k+3} - 18y_{k+2} + 9y_{k+1} - 2y_k\big) = h\, f_{k+3},$$
$$\varrho(\mu) = \frac{1}{6}\big(11\mu^3 - 18\mu^2 + 9\mu - 2\big).$$

Satz 4.11 (Konsistenzordnung bei MSV[42])**.**

Folgende Aussagen sind äquivalent [43]:

(a) Das MSV (4.2) besitzt für $y \in C^{p+1}([0,T];\mathbb{R}^n)$ (mindestens) die Konsistenzordnung $p \geq 1$.

(b) Die Bedingungen

$$\sum_{m=0}^{r} \big(m^j \alpha_m - j\, m^{j-1} \beta_m\big) = 0 \tag{4.13}$$

sind für $j = 0, 1, \ldots, p$ erfüllt mit der Konvention $0^0 = 1$.

(c) Die Funktion

$$\varphi(\mu) := \varrho(\mu) - \sigma(\mu)\, \ln \mu \tag{4.14}$$

besitzt $\mu = 1$ als $(p+1)$-fache Nullstelle. Das heißt, die Ableitungen von φ bis zur Ordnung p einschließlich verschwinden bei $\mu = 1$.

[42] siehe Plato, 2004, Lemma 8.16, Hermann, 2004, Satz 3.3

[43] In Sheet 5, Exercise 15 wird die Konsistenzordnung verschiedener MSV mit Hilfe dieses Satzes bestimmt.

Beweis: Es sei $y \in C^{p+1}([0,T];\mathbb{R}^n)$. Eine Taylor-Entwicklung von $y(t+m\,h)$ und $y'(t+m\,h)$ im Punkt t liefert

$$y(t+m\,h) = \sum_{j=0}^{p} \frac{(mh)^j}{j!} y^{(j)}(t) + \underbrace{\frac{f^{(p+1)}(\xi)}{p!}(m\,h)^{p+1}}_{=\mathcal{O}(h^{p+1})}, \quad \xi \in (t, t+m\,h)$$

$$\begin{aligned} y'(t+m\,h) &= \sum_{j=0}^{p-1} \frac{(mh)^j}{j!} y^{(j+1)}(t) + \mathcal{O}(h^p) \\ &= \sum_{j=1}^{p} \frac{(mh)^{j-1}}{(j-1)!} y^{(j)}(t) + \mathcal{O}(h^p). \end{aligned}$$

Wir werten den Konsistenzfehler (Definition 4.7) aus:

$$\begin{aligned} & h\,d(y(\cdot), t+r\,h, h) \\ &= \sum_{m=0}^{r} \alpha_m\, y(t+m\,h) - h \sum_{m=0}^{r} \beta_m \underbrace{f(t+m\,h, y(t+m\,h))}_{=y'(t+m\,h)} \\ &= \sum_{m=0}^{r} \alpha_m \sum_{j=0}^{p} \frac{(mh)^j}{j!} y^{(j)}(t) - h \sum_{m=0}^{r} \beta_m \sum_{j=1}^{p} \frac{(mh)^{j-1}}{(j-1)!} y^{(j)}(t) + \mathcal{O}(h^{p+1}) \\ &= \underbrace{y(t) \sum_{m=0}^{r} \alpha_m}_{\text{Summand } j=0} + \sum_{j=1}^{p} \left(h^j y^{(j)}(t) \sum_{m=0}^{r} \left[\alpha_m \frac{m^j}{j!} - \beta_m \frac{m^{j-1}}{(j-1)!} \right] \right) + \mathcal{O}(h^{p+1}). \end{aligned}$$

Der von h unabhängige Summand verschwindet, wenn $\sum_{m=0}^{r} \alpha_m = 0$ gilt, also (4.13) für $j=0$. Gilt (4.13) auch für $j=1$, so ist $d(\cdot) = \mathcal{O}(h)$ und damit das MSV konsistent mit Ordnung 1. Analoges gilt für höhere Konsistenzordnungen. Damit ist (a) $\Leftrightarrow$ (b) bewiesen. Die Äquivalenz zu (c) wird in Sheet 6, Exercise 19 behandelt. □

Bemerkung 4.12. Bei RKV ergaben sich als Ordnungsbedingungen nichtlineare algebraische Gleichungen an die Koeffizienten. Das ist bei linearen MSV einfacher, (4.13) ist lineares Gleichungssystem. Grund hierfür ist, dass es bei MSV keine geschachtelten f-Auswertungen gibt.

Beispiel 4.13 (Konsistenzordnungen einiger MSV)**.**

(a) Die **explizite Mittelpunktsregel** ist ein Vertreter der Nyström-Verfahren (Beispiel 4.3 (c))

$$y_{k+2} = y_k + 2\,h\,f_{k+1}$$

mit der Abkürzung $f_{k+1} = f(t_{k+1}, y_{k+1})$. Sie besitzt Konsistenzordnung $p=2$.

(b) Die **Milne-Methode** ist ein Vertreter der Milne-Simpson-Verfahren (Beispiel 4.3 (c)):

$$y_{k+2} = y_k + \frac{h}{3}\left(f_{k+2} + 4f_{k+1} + f_k\right).$$

Sie besitzt (ausnahmsweise) die Konsistenzordnung $p=4$.[44]

Satz 4.14 (Konsistenzordnungen der klassischen MSV[45])**.** Ein r-Schritt-Verfahren

(a) der (expliziten) AB-Klasse besitzt Konsistenzordnung r.

(b) der (impliziten) AM-Klasse besitzt Konsistenzordnung $r+1$.

(c) der (impliziten) BDF-Klasse besitzt Konsistenzordnung r.

(d) besitzt maximal Konsistenzordnung $2r$. Dieses Verfahren ist eindeutig (bei Normierung $\alpha_r = 1$).

Beweis:

(a) Nach Konstruktion arbeiten AB-Verfahren exakt, falls $t \mapsto f(t, y(t))$ ein Polynom vom Grad $\leq r-1$ ist. Wähle nun $f(t) = j\, t^{j-1}$ (unabhängig von y) zunächst mit $1 \leq j \leq r$ als ein solches Polynom. Die exakte Lösung der Dgl mit AW $y(0) = 0$ ist dann $y(t) = t^j$. Das heißt,

$$\begin{aligned} 0 &= h\, d(y(\cdot), 0 + r\,h, h) && \text{kein Fehler} \\ &= \sum_{m=0}^{r} \alpha_m\, y(mh) - h \sum_{m=0}^{r} \beta_m\, f(mh) && \text{Definition von } d \\ &= \sum_{m=0}^{r} \alpha_m\, (mh)^j - h \sum_{m=0}^{r} \beta_m\, j (mh)^{j-1} && y \text{ und } f \text{ einsetzen} \\ &= h^j \sum_{m=0}^{r} \left[\alpha_m\, m^j - \beta_m\, j\, m^{j-1}\right]. \end{aligned}$$

Also gilt (4.13) für alle $j = 1, \ldots, r$. Der Sonderfall $j = 0$ folgt analog mit $y(t) \equiv 1$ und $f(t) \equiv 0$, oder man prüft $\sum_{m=0}^{r} \alpha_m = 0$ direkt nach. Nach Satz 4.11 hat das Verfahren Ordnung $p = r$.

(b) Beweis analog zu (a), benutze Polynome mit $0 \leq j \leq r+1$.

(c) Nach Konstruktion integrieren BDF-Verfahren exakt, wenn $t \mapsto y(t)$ ein Polynom vom Grad $\leq r$ ist. Beweis analog zu (a).

(d) Ein allgemeines lineares r-Schritt Verfahren (4.2) hat r Parameter α_m und $r+1$ Parameter β_m, denn o. B. d. A. gilt $\alpha_r = 1$.[46] Die Konsistenzbedingungen (4.13) liefern $p+1$ Bedingungen. Diese haben stets vollen Rang (ohne Beweis). System ist also für $p = 2r$ eindeutig lösbar.

Eine höhere Ordnung $p > 2r$ ist ausgeschlossen, denn bei Anwendung auf $y'(t) = f(t)$ erzeugt ein lineares MSV eine Quadraturformel mit r Stützstellen vom Genauigkeitsgrad $q \leq 2r-1$. Andererseits beträgt der Genauigkeitsgrad $q = p - 1$, vgl. Lemma 3.38.

□

[44] vgl. auch Satz 4.25

[45] Dies soll in Sheet 7, Exercise 20 numerisch überprüft werden.

[46] Ein Parameter in (4.2) ist redundant (Skalierung der Gleichung).

Es wird sich zeigen, dass es nicht sinnvoll ist, lineare MSV möglichst hoher Konsistenzordnung aus den Bedingungen (4.13) zu konstruieren. Die Aussage (d) ist damit praktisch bedeutungslos.

Beispiel 4.15 (Ein instabiles (nicht nullstabiles) MSV)**.** Wir betrachten ein lineares *explizites* Zweischrittverfahren ($r = 2$):

$$\alpha_0\, y_k + \alpha_1\, y_{k+1} + y_{k+2} = h\big[\beta_0\, f(t_k, y_k) + \beta_1\, f(t_{k+1}, y_{k+1})\big].$$

Die Ordnungsbedingungen (4.13) liefern

$$\begin{aligned} j = 0 \quad &\Rightarrow \quad \alpha_0 + \alpha_1 + \alpha_2 = 0 \\ j = 1 \quad &\Rightarrow \quad \alpha_1 + 2\alpha_2 = \beta_0 + \beta_1 + \beta_2 \\ j = 2 \quad &\Rightarrow \quad \alpha_1 + 4\alpha_2 = 2\,(\beta_1 + 2\beta_2) \\ j = 3 \quad &\Rightarrow \quad \alpha_1 + 8\alpha_2 = 3\,(\beta_1 + 4\beta_2). \end{aligned}$$

Dabei sind $\alpha_2 = 1$ (Konvention) und $\beta_2 = 0$ (explizites MSV). Die maximale Konsistenzordnung $p = 3$ erhalten wir genau für die Koeffizienten

$$\alpha_0 = -5, \quad \alpha_1 = 4, \quad \beta_0 = 2, \quad \beta_1 = 4,$$

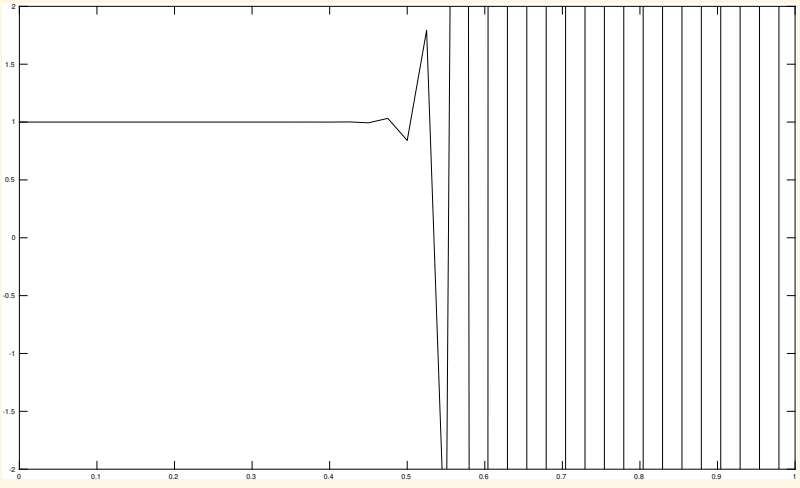
also für das Verfahren

$$y_{k+2} + 4y_{k+1} - 5y_k = h\big[4f(t_{k+1}, y_{k+1}) + 2f(t_k, y_k)\big]. \tag{4.15}$$

Wir wenden es auf die Dgl[47]

$$y'(t) = 0, \quad y(0) = 1$$

mit der Lösung $y(t) \equiv 1$ an. Mit den Startwerten $y_0 = 1$ und (gestörtem) $y_1 = 1 + \varepsilon$ (hier mit $\varepsilon = 10^{-8}$) ergibt sich die Näherungslösung [48]:



Die Näherungen erfüllen

$$\begin{aligned} &-5y_0 + 4y_1 + y_2 = 0 \\ \text{bzw. allgemein} \quad &-5\,y_k + 4\,y_{k+1} + y_{k+2} = 0. \end{aligned} \tag{4.16}$$

(4.16) ist eine **lineare Differenzengleichung** (im Folgenden: Δgl).

Ansatz für deren Lösung: $y_n = \mu^n$. Einsetzen liefert

$$\mu^n\big[-5 + 4\,\mu + \mu^2\big] = 0 \tag{4.17}$$

mit den nicht-trivialen Lösungen

$$\mu_1 = 1, \quad \mu_2 = -5.$$

Die Lösungsfolgen $\{y_n\}$ der Δgl (4.16) sind genau die Linearkombinationen der sogenannten Fundamentallösungen $\{\mu_1^n\}$ und $\{\mu_2^n\}$, siehe Satz 4.16, also

$$y_n = c_1\,\mu_1^n + c_2\,\mu_2^n.$$

Die Startwerte $y_0 = 1$ und $y_1 = 1 + \varepsilon$ legen die Konstanten fest:

$$c_1 = 1 + \frac{\varepsilon}{6}, \quad c_2 = -\frac{\varepsilon}{6}.$$

Die Folge der Näherungslösungen ist also

$$y_n = \big(1 + \frac{\varepsilon}{6}\big)\,1^n - \frac{\varepsilon}{6}\big(-5\big)^n,$$

wobei der zweite Term mit $n \to \infty$ zunehmend stärker oszilliert und die Lösung dominiert. Man spricht von einer **parasitären Komponente**.

Dies hängt offenbar damit zusammen, dass $\mu = -5$ eine Nullstelle des ersten assoziierten Polynoms

$$\varrho(\mu) = \mu^2 + 4\mu - 5$$

ist, die vom Betrag größer als eins ist.

Fazit: Das Verfahren (4.15) ist **instabil** (nicht diskret stabil) und offenbar nicht konvergent. Die fehlende diskrete Stabilität eines numerischen Verfahrens bedeutet, dass die numerische Lösung mit $h \searrow 0$ in der entsprechenden Norm beliebig groß werden kann. Das ist hier der Fall. [49]

Die diskrete Stabilität von ESV hatten wir in Lemma 3.15 nachgewiesen. Daraufhin übertrug sich die Konsistenz auf die Konvergenz mit derselben Ordnung. Bei MSV benötigen wir noch geeignete Bedingungen, um diskrete Stabilität zu erreichen.

Beachte: Die gerade beobachtete Instabilität eines MSV hat nichts mit fehlender A-Stabilität bei ESV (vgl. § 3.4) zu tun.[50]

§ 4.2 Konvergenzuntersuchung bei Mehrschrittverfahren

Wir betrachten die **lineare homogene Differenzengleichung** (Δgl)

$$\alpha_r\,y_{k+r} + \alpha_{r-1}\,y_{k+r-1} + \cdots + \alpha_1\,y_{k+1} + \alpha_0\,y_k = 0, \quad i = 0, 1, \ldots \tag{4.18}$$

mit $\alpha_r \neq 0$. Diese entsteht bei Anwendung des MSV (4.2) auf $y'(t) = 0$. Zu gegebenen Startwerten $y_0, \ldots, y_{r-1}$ liefert (4.18) eine eindeutige Lösungsfolge $\{y_n\}$.[51] Das zugehörige erste assoziierte (charakteristische) Polynom ist nach (4.12a):

$$\varrho(\mu) := \alpha_r\,\mu^r + \alpha_{r-1}\,\mu^{r-1} + \cdots + \alpha_1\,\mu^1 + \alpha_0.$$

[47] Das erklärt auch den Begriff „nullstabil“.

[48] Siehe Matlab-Skript `Unstable_MSV_demo.m`

[49] In Sheet 10, Exercise 27 wird ein stabiles lineares explizites Zweischrittverfahren mit Konsistenzordnung $p = 2$ konstruiert.

[50] Dort konvergierte die Lösung ja für $h \to 0$, hier jedoch nicht!

[51] Lösungen von Differenzengleichungen sind Folgen. Wir beginnen o. B. d. A. beim Index $n = 0$.

Satz 4.16 (Lösungen der linearen homogenen Δgl)**.** Es sei $\mu_0 \in \mathbb{C}$ eine Nullstelle von ϱ der Vielfachheit N, d. h.,

$$\varrho(\mu_0) = \varrho'(\mu_0) = \ldots = \varrho^{(N-1)}(\mu_0) = 0.$$

Dann gilt:

(a) Jede der Folgen

$$y_n = \frac{\mathrm{d}^s}{\mathrm{d}\mu^s}[\mu^n]_{\mu=\mu_0} = \Big(\prod_{\ell=0}^{s-1}(n-\ell)\Big)\,\mu_0^{n-s}, \quad n \in \mathbb{N}_0$$

für $s = 0, 1, \ldots, N-1$ ist eine Lösung von (4.18) und heißt eine **Fundamentallösung**.

(b) Die Lösungen von (4.18) (ohne Anfangsbedingungen) sind genau die Linearkombinationen aller Fundamentallösungen.

Beweis: (a): Es sei $s \in \{0, 1, \ldots, N-1\}$ beliebig. Wir setzen $y_n = \frac{\mathrm{d}^s}{\mathrm{d}\mu^s}\mu^n$ in (4.18) ein:

$$\begin{aligned}
&\alpha_r \frac{\mathrm{d}^s}{\mathrm{d}\mu^s}\mu^{k+r} + \alpha_{r-1}\frac{\mathrm{d}^s}{\mathrm{d}\mu^s}\mu^{k+r-1} + \cdots + \alpha_0 \frac{\mathrm{d}^s}{\mathrm{d}\mu^s}\mu^k \\
&\qquad = \frac{\mathrm{d}^s}{\mathrm{d}\mu^s}\left[\alpha_r\,\mu^{k+r} + \alpha_{r-1}\,\mu^{k+r-1} + \cdots + \alpha_0\,\mu^k\right] \\
&\qquad = \frac{\mathrm{d}^s}{\mathrm{d}\mu^s}\left[\mu^k \varrho(\mu)\right] \\
&\qquad = \sum_{\ell=0}^{s}\binom{s}{\ell}\left(\frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu^k\right)\underbrace{\varrho^{(s-\ell)}(\mu)}_{=0 \text{ für } \mu=\mu_0} \qquad \text{Leibniz-/Produktregel} \\
&\qquad = 0.
\end{aligned}$$

(b): Es seien $\mu_1, \ldots, \mu_p$ die paarweise verschiedenen Nullstellen von ϱ und

$$\varrho(\mu) = \alpha_r\,(\mu-\mu_1)^{N_1}\cdots(\mu-\mu_p)^{N_p}$$

mit $N_1 + \cdots + N_p = r$. Aus Teil (a) erhalten wir r Lösungsfolgen.

Wir zeigen: Diese Folgen bilden eine Basis des Lösungsraumes. Es sei dazu $\{y_n\}$ eine beliebige Lösung von (4.18). Wir verwenden zunächst den Ansatz:

$$\sum_{j=1}^{p}\sum_{\ell=0}^{N_j-1} c_{j\ell}\,\frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu^k\Big|_{\mu=\mu_j} = y_n \quad \text{für alle } k \in \mathbb{N}_0. \tag{$*$}$$

Zu zeigen: Die $c_{j\ell}$ sind eindeutig bestimmt. Wir fassen $c = c_{j\ell}$ als Vektor in $\mathbb{R}^r$ auf und werten ($*$) zunächst nur für $n = 0, 1, \ldots, r-1$ aus. Das generiert r Bedingungen für c, und wir erhalten das lineare Gleichungssystem

$$A\,c = \begin{pmatrix} y_0 \\ \vdots \\ y_{r-1} \end{pmatrix}.$$

Die Spalte von A, die zur Unbekannten $c_{j\ell}$ gehört, hat die Gestalt

$$\begin{pmatrix} \frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu^0 \\ \frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu^1 \\ \vdots \\ \frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu^{r-1} \end{pmatrix}\Bigg|_{\mu=\mu_j}.$$

Zu zeigen: A ist regulär.

Betrachte dazu $A^\top\gamma = 0$ mit $\gamma = (\gamma_0, \gamma_1, \dots, \gamma_{r-1})^\top$.

$$A^\top\gamma = 0 \quad \Leftrightarrow \quad \gamma_0 \frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu^0\Big|_{\mu=\mu_j} + \gamma_1 \frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu^1\Big|_{\mu=\mu_j} + \cdots + \gamma_{r-1} \frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu^{r-1}\Big|_{\mu=\mu_j} = 0$$
$$\text{für alle } j = 1, \dots, p, \quad \ell = 0, \dots, N_j - 1.$$

Mit anderen Worten, das Polynom $P(\mu) = \sum_{m=0}^{r-1} \gamma_m \mu^m$ löst die Hermitesche Interpolationsaufgabe[52]

$$\text{Finde } P \in \Pi_{r-1}, \text{ sodass } P^{(\ell)}(\mu_j) = 0 \quad \text{für alle } j = 1, \dots, p, \quad \ell = 0, \dots, N_j - 1.$$

Die eindeutige Lösung ist $P \equiv 0$ mit $\gamma = 0$. Dies zeigt die Injektivität von $A^\top$. Aus der Lösbarkeit der obigen Interpolationsaufgabe für beliebige Daten folgt die Surjektivität. Also sind $A^\top \in \mathbb{R}^{r\times r}$ und A regulär.

Die ersten r Glieder der Folge $\{y_n\}$ legen also die Koeffizienten $c_{j\ell}$ der LK der Fundamentallösungen eindeutig fest. Noch zu zeigen ist, dass sich dann die *gesamte* Folge $\{y_n\}$ als diese LK der Fundamentallösungen schreiben lässt. Betrachte dazu das nächste Folgenglied:

$$\begin{aligned} y_r \overset{(4.18)}{=} \frac{-1}{\alpha_r} \sum_{k=0}^{r-1} \alpha_k y_k &\overset{(*)}{=} \frac{-1}{\alpha_r} \sum_{k=0}^{r-1} \alpha_k \sum_{j=1}^{p} \sum_{\ell=0}^{N_j-1} c_{j\ell} \frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu_j^k \\ &= \frac{-1}{\alpha_r} \sum_{j=1}^{p} \sum_{\ell=0}^{N_j-1} c_{j\ell} \sum_{k=0}^{r-1} \alpha_k \frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu_j^k \\ &= \frac{-1}{\alpha_r} \sum_{j=1}^{p} \sum_{\ell=0}^{N_j-1} c_{j\ell} \, (-\alpha_r) \frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu_j^r, \end{aligned}$$

denn $\frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu^k$ ist Lösung von (4.18). D. h.,

$$y_r = \sum_{j=1}^{p} \sum_{\ell=0}^{N_j-1} c_{j\ell} \frac{\mathrm{d}^\ell}{\mathrm{d}\mu^\ell}\mu_j^r.$$

Per Induktion folgt die gleiche Aussage auch für alle späteren Folgenglieder. □

Beispiel 4.17 (Lösungen von Δgl)**.**

[52] siehe z. B. Stoer, 2004, § 2.1.5; falls nur einfache Nullstellen $\mu_1, \dots, \mu_r$ auftreten ist das eine Lagrangesche Interpolationsaufgabe

(a) Zur Δgl
$$y_{k+2} - 3y_{k+1} + 2y_k = 0$$
gehört $\varrho(\mu) = \mu^2 - 3\mu + 2 = (\mu - 1)(\mu - 2)$. Die allgemeine Lösung ist daher
$$y_n = c_1 + c_2\, 2^n.$$

(b) Zur Δgl
$$y_{k+2} - 4y_{k+1} + 4y_k = 0$$
gehört $\varrho(\mu) = \mu^2 - 4\mu + 4 = (\mu - 2)^2$. Die allgemeine Lösung lautet nun
$$y_n = c_1 2^n + c_2\, n\, 2^{n-1}.$$

Wir führen nun den entscheidenden Begriff ein, der bei MSV zur diskreten Stabilität (siehe Lemma 4.23) führt.

Definition 4.18 (Nullstabilität)**.** Ein lineares MSV mit erstem charakteristischen Polynom ϱ heißt **nullstabil** (**stabil**, **wurzelstabil**, **D(ahlquist)-stabil**), wenn die sogenannte **Wurzelbedingung** erfüllt ist:

(*i*) Die Nullstellen $\mu \in \mathbb{C}$ von ϱ erfüllen $|\mu| \le 1$.

(*ii*) $\varrho(\mu) = 0$ und $|\mu| = 1 \quad \Rightarrow \quad \varrho'(\mu) \neq 0$.

Die Nullstellen des ersten charakteristischen Polynoms liegen also im Einheitskreis, und die auf dem Rand liegenden Nullstellen sind einfach.

Beachte: Aufgrund der Bedingung (4.14) ist $\mu = 1$ für konsistente MSV immer eine Nullstelle von ϱ. Diese muss für ein nullstabiles Verfahren einfach sein. Sie gehört zur konstanten Fundamentallösung $y_n \equiv 1$ der homogenen Δgl (4.18).

Bemerkung 4.19 (Bedeutung der Nullstabilität)**.** Aus Satz 4.16 folgt: Nullstabilität ⇔ Jede Lösungsfolge der homogenen Δgl (4.18) ist beschränkt [53]. Dies sorgt später (Lemma 4.23) für eine kontrollierte Fortpflanzung der lokalen Fehler pro Zeitschritt. Das MSV aus Beispiel 4.15 war nicht nullstabil.

Analog zu ESV definieren wir (vgl. Definition 3.14):

Definition 4.20 (Fehler, Konvergenz, Konvergenzordnung bei MSV)**.** Es sei $y(\cdot)$ die exakte Lösung des (**AWP**) auf dem Intervall $[0, T]$. Weiter sei $y_h(\cdot)$ eine mit einem linearen MSV (4.2) erzeugte Näherungslösung auf dem äquidistanten Gitter[54]
$$\mathcal{T} = \{k\,h : k = 0, 1, \ldots, \lfloor T/h \rfloor\}$$
der Schrittweite h. Diese Näherung hängt von Startwerten $y_h(k\,h) = y_k$ für $k = 0, 1, \ldots, r-1$ ab.

(a) Die Größe
$$e_h(t) := y_h(t) - y(t), \quad t \in \mathcal{T}$$
heißt der **globale (Diskretisierungs-)Fehler** des Verfahrens auf dem Gitter $\mathcal{T}$ an der Stelle $t \in \mathcal{T}$.

[53] siehe auch www.am.uni-erlangen.de/am1/de/scripts/knabner/skript_num2_kap2.ps, S.114; dort wird die Nullstabilität mit der Lipschitz-Stabilität des Lösungsoperators in Verbindung gebracht! Dieser Zusammenhang gehört zu Lemma 4.23

(b) Das MSV (4.2) heißt **konvergent**, falls gilt:

$$\|y_h - y\|_{\infty,h} := \max_{t\in\mathcal{T}} \|y_h(t) - y(t)\| \to 0 \quad \text{für } h \searrow 0.$$

Beachte: Dies schließt insbesondere die Konvergenz der Startwerte ein:

$$\max_{k=0,1,\dots,r-1} \|y_h(k\,h) - y(k\,h)\| \to 0 \quad \text{für } h \searrow 0.$$

(c) Das MSV hat die **Konvergenzordnung** $p \in \mathbb{N}$, falls eine von h unabhängige Konstante $c > 0$ sowie $\overline{h} > 0$ existieren, sodass für die Näherungslösung auf dem Gitter $\mathcal{T}$ der äquidistanten Gitterweite $h \le \overline{h}$ gilt:

$$\|y_h - y\|_{\infty,h} := \max_{t\in\mathcal{T}} \|y_h(t) - y(t)\| \le c\,h^p.$$

Beachte: Dabei wird insbesondere dieselbe Konvergenzordnung für die Startwerte benötigt.

$$\max_{k=0,1,\dots,r-1} \|y_h(k\,h) - y(k\,h)\| = \mathcal{O}(h^p) \quad \text{für } h \searrow 0.$$

Beispiel 4.15 zeigt, dass ein konsistentes MSV noch nicht konvergent sein muss. Für lineare MSV gilt (unter geeigneten Voraussetzungen) der wichtige Zusammenhang:

Konvergenz $\Leftrightarrow$ Konsistenz + Nullstabilität
Außerdem überträgt sich die Konsistenz- auf die Konvergenzordnung.

Wir beweisen zunächst die Richtung „⇒“.

Satz 4.21. Konvergenz impliziert Nullstabilität und Konsistenz
Ist ein lineares MSV konvergent für alle mit $h \searrow 0$ konvergenten Startwerte $y_h(0), y_h(h), \dots, y_h((r-1)\,h)$, dann ist es nullstabil und konsistent.

Beweis: Zur Nullstabilität: Wir wenden das MSV (4.2) auf das AWP

$$y'(t) = 0, \quad y(0) = 0$$

mit der Lösung $y(t) \equiv 0$ an und erhalten die lineare homogene Δgl (4.18), also

$$\alpha_r\, y_h(t_{k+r}) + \alpha_{r-1}\, y_h(t_{k+r-1}) + \dots + \alpha_0\, y_h(t_k) = 0, \quad k = 0, 1, \dots \qquad (*)$$

Annahme: Das MSV ist nicht nullstabil, d. h., das Polynom ϱ besitzt eine Nullstelle $|\mu| > 1$ oder eine mehrfache Nullstelle $|\mu| = 1$. Wir untersuchen beide Fälle.

1. Fall, $|\mu| > 1$: Dann ist $c\,\mu^n$ nach Satz 4.16 (a) eine Lösungsfolge von $(*)$, die (falls $c \neq 0$) für $n \to \infty$ divergiert (betragsmäßig unbeschränkt wächst). Wir betrachten diese Folge insbesondere für $c = \sqrt{h}$, also

$$y_h(nh) := \sqrt{h}\,\mu^n.$$

Die ersten r Glieder (Startwerte) erfüllen die Konvergenzbedingung in Definition 4.20 (b) für die Startwerte, nämlich

$$\lim_{h\searrow 0} y_h(n\,h) = 0$$

[54] **Beachte:** Der Endpunkt T muss nicht als Vielfaches von h erreichbar sein. Deshalb gehen wir notfalls nur bis $\lfloor T/h \rfloor$ (abrunden).

für alle $n = 0, 1, \ldots, r-1$.

Es sei nun eine Stelle $t^* \in (0, T]$ fest gewählt.[55] Die feste Stelle t^* wird erreicht durch n Schritte der Schrittweite $h_n = t^*/n$. Daher gilt

$$y_h(t^*) - y(t^*) = y_h(n\,h_n) - 0 = \sqrt{h_n}\,\mu^n = \frac{\sqrt{t^*}}{\sqrt{n}}\,\mu^n.$$

Dies divergiert für $n \to \infty$, im Widerspruch zur vorausgesetzten Konvergenz des Verfahrens für *beliebige konvergente Startwerte*.

2. Fall, $|\mu| = 1$ ist mehrfache Nullstelle: analog zu obigem, $c\,n\,\mu^{n-1}$ ist Lösung von (∗). Die Folge $y_h(n\,h) := \sqrt{h}\,n\,\mu^{n-1}$ erfüllt $\lim_{h \searrow 0} y_h(n\,h) = 0\ \forall n = 0, 1, \ldots, r-1,$, aber

$$y_h(t^*) - y(t^*) = \sqrt{h_n}\,n\,\mu^{n-1} = \frac{n\sqrt{t^*}}{\sqrt{n}}\,\mu^{n-1}$$

divergiert.

Zur Konsistenz: Wir müssen mindestens Konsistenzordnung $p = 1$ zeigen, also (4.13) für $j = 0$ und $j = 1$ zeigen, d. h.

$$\sum_{m=0}^{r} \alpha_m = 0 \quad \text{und} \quad \sum_{m=0}^{r} m\,\alpha_m - \beta_m = 0. \tag{∗∗}$$

$j = 0$: Wir wenden das MSV auf das AWP

$$y'(t) = 0, \quad y(0) = 1$$

mit exakten Startwerten $y_h(t_0) = \ldots = y_h(t_{r-1}) = 1$ an. Die Näherungen erfüllen die Δgl (∗). Daraus folgt natürlich

$$\lim_{n\to\infty} \left(\alpha_r\,y_h(t_{n+r}) + \alpha_{r-1}\,y_h(t_{n+r-1}) + \cdots + a_0\,y_h(t_n)\right) = 0.$$

Wegen der Konvergenz des MSV konvergiert notwendig $y_h \to 1$, also auch die Teilfolgen $\{y_h(t_{n+1})\}, \ldots, \{y_h(t_{n+r})\}$. Somit folgt

$$\sum_{m=0}^{r} \alpha_m = 0,$$

also (∗∗) für $j = 0$.

$j = 1$: Nun wenden wir das MSV auf das AWP

$$y'(t) = 1, \quad y(0) = 0$$

mit exakter Lösung $y(t) = t$ an. Es ergibt sich die *inhomogene* Δgl

$$\sum_{m=0}^{r} \alpha_m\,y_h(t_{k+m}) = h \sum_{m=0}^{r} \beta_m \quad \text{für } k = 0, 1, \ldots \tag{4.19}$$

Wir setzen $\gamma := \sigma(1)/\varrho'(1)$. **Beachte:** $\varrho'(1) \neq 0$, denn oben wurde bereits die Nullstabilität gezeigt, sodass $\mu = 1$ nur eine einfache Nullstelle von ϱ sein kann.

[55] Für einen Widerspruch zur Konvergenz im Sinne der Norm $\|\cdot\|_{\infty,h}$ (Definition 4.20) reicht es, eine feste Stelle zu betrachten.

Wähle als Startwerte $y_h(k\,h) = \gamma\,k\,h$ für $k = 0, 1, \ldots, r-1$. Diese konvergieren gemäß Definition 4.20 (b), da

$$\max_{k=0,\ldots,r-1} |y_h(k\,h) - y(k\,h)| = |\gamma - 1|\,(r-1)\,h \to 0 \quad \text{für} \quad h \searrow 0.$$

Die eindeutige Lösung von (4.19) zu diesen Startwerten ist

$$y_h(t_k) = \gamma\,k\,h \quad \text{für alle } k \in \mathbb{N}_0,$$

denn es gilt:

$$\begin{aligned}
\sum_{m=0}^{r} \alpha_m\, y_h(t_{k+m}) &= h\,\gamma \sum_{m=0}^{r} \alpha_m\,(k+m) \\
&= h\,\gamma\,k \underbrace{\sum_{m=0}^{r} \alpha_m}_{=\varrho(1)=0,\ \text{s. o.}} + h\,\gamma \underbrace{\sum_{m=0}^{r} \alpha_m\,m}_{=\varrho'(1)} \\
&= h\,\gamma\,\varrho'(1) = h\,\sigma(1) \qquad \text{nach Definition von } \gamma \\
&= h \sum_{m=0}^{r} \beta_m.
\end{aligned}$$

Aus der Konvergenz des Verfahrens ergibt sich wieder mit $t^* \in (0, T]$ beliebig und $h_n = t^*/n$

$$t^* = y(t^*) = \lim_{n\to\infty} y_h(n h_n) = \lim_{n\to\infty} \gamma\,n\,\frac{t^*}{n} = \gamma\,t^*,$$

woraus $\gamma = 1$ und $\varrho'(1) - \sigma(1) = 0$ folgt, also (∗∗) für $j = 1$. □

Für den Beweis der Rückrichtung benötigt man wie bei ESV zunächst einen Beweis der diskreten Stabilität des MSV, der auf der Annahme der Nullstabilität beruht.

Bevor wir dazu kommen, noch ein Hilfssatz:

Lemma 4.22 (Spektralradius und passende Vektornorm)**.** Es sei $A \in \mathbb{R}^{n\times n}$ oder $\mathbb{C}^{n\times n}$ mit Eigenwerten $\lambda_1, \ldots, \lambda_n \in \mathbb{C}$ und dem Spektralradius

$$R(A) = \max_{i=1,\ldots,n} |\lambda_i|.$$

(a) Für jedes $\varepsilon > 0$ existiert eine Vektornorm $\|\cdot\|_*$ von $\mathbb{R}^n$, sodass gilt:

$$R(A) \le \|A\|_* \le R(A) + \varepsilon.$$

(b) Stimmen für jeden Eigenwert von A mit $|\lambda| = R(A)$ die algebraische und die geometrische Vielfachheit überein, dann kann $\varepsilon = 0$ gewählt werden. (Dies ist insbesondere dann der Fall, wenn die betragsgrößten Eigenwerte alle nur die algebraische Vielfachheit eins besitzen.)

Beweis: Die Matrix A ist ähnlich zu ihrer Jordanschen Normalform:

$$J = W^{-1} A\,W,$$

wobei W spaltenweise die Eigen- und ggf. Hauptvektoren von A enthält. Die blockdiagonale Matrix J besteht aus Jordanblöcken

$$\begin{bmatrix} \lambda_i & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_i \end{bmatrix}$$

mit den Eigenwerten von A auf der Hauptdiagonale. Es sei nun $\varepsilon > 0$. Wir setzen D_ε als Blockmatrix mit Blöcken

$$\begin{bmatrix} 1 & & & \\ & \varepsilon & & \\ & & \ddots & \\ & & & \varepsilon^{n-1} \end{bmatrix}$$

(passend zu J) und betrachten

$$J_\varepsilon := D_\varepsilon^{-1} J D_\varepsilon.$$

Die Matrix J_ε hat dieselbe Struktur wie J, besteht jedoch aus den Blöcken

$$\begin{bmatrix} \lambda_i & \varepsilon & & \\ & \ddots & \ddots & \\ & & \ddots & \varepsilon \\ & & & \lambda_i \end{bmatrix}.$$

Die gesuchte Vektornorm auf $\mathbb{R}^n$ ist gegeben durch

$$\|x\|_* := \|T^{-1}x\|_\infty \quad \text{mit} \quad T := W D_\varepsilon, \quad x \in \mathbb{R}^n,$$

denn: Für die durch sie induzierte Matrixnorm gilt

$$\begin{aligned} \|A\|_* &= \max_{x \in \mathbb{R}^n \setminus \{0\}} \frac{\|T^{-1} A\, x\|_\infty}{\|T^{-1}x\|_\infty} = \max_{y \in \mathbb{R}^n \setminus \{0\}} \frac{\|T^{-1} A\, T\, y\|_\infty}{\|y\|_\infty}, \\ &= \|T^{-1} A\, T\|_\infty = \|D_\varepsilon^{-1} W^{-1} A\, W\, D_\varepsilon\|_\infty = \|D_\varepsilon^{-1} J\, D_\varepsilon\|_\infty \\ &\le \max_{i=1,\dots,n} |\lambda_i| + \varepsilon = R(A) + \varepsilon. \end{aligned} \tag{*}$$

Wählen wir speziell für x einen Eigenvektor zu einem der betragsgrößten Eigenwerte, so ergibt sich

$$\frac{\|T^{-1} A\, x\|_\infty}{\|T^{-1}x\|_\infty} = \frac{\|\lambda\, T^{-1}x\|_\infty}{\|T^{-1}x\|_\infty} = |\lambda| = R(A),$$

also auch $\|A\|_* \ge R(A)$. Dies zeigt Behauptung (a).

Unter der Voraussetzung von Teil (b) besitzen alle Jordanblöcke, die zu einem der betragsgrößten Eigenwerte gehören, die Größe 1×1. Damit erhält man statt (*) für hinreichend kleine ε:

$$\|A\|_* = \|D_\varepsilon^{-1} J\, D_\varepsilon\|_\infty = \max_{i=1,\dots,n} |\lambda_i|.$$

□

Nun zum eigentlichen Ziel:

Lemma 4.23 (Diskrete Stabilität von MSV, vgl. Lemma 3.15)**.** Es sei $\mathcal{T} = \{k\,h : k = 0, 1, \ldots, N := \lfloor T/h \rfloor\}$ ein äquidistantes Gitter und w_h und z_h zwei *beliebige* Gitterfunktionen auf $\mathcal{T}$. Das lineare MSV (4.2) sei nullstabil. Dann gilt (bei impliziten MSV für hinreichend kleine $h \le \overline{h}$) die **diskrete Stabilitätsabschätzung**

$$\|w_n - z_n\| \le C \left(\max_{0 \le k \le r-1} \|w_k - z_k\| + \sum_{j=0}^{n-r} h \, \|d^w_{j+r} - d^z_{j+r}\| \right) \exp\left(K\, t_n\right) \tag{4.20}$$

für alle $n = 0, 1, \ldots, N$. Dabei steht d^w_{j+r} als Abkürzung für den Konsistenzfehler *zur Gitterfunktion*[56] w_h

$$d^w_{j+r} := d(w_h(\cdot), t_{j+r}, h) = \frac{1}{h}\Big[\sum_{m=0}^{r} \alpha_m \, w_{j+m} - h \sum_{m=0}^{r} \beta_m \, f(t_{j+m}, w_{j+m}) \Big]$$

für alle $j = 0, 1, \ldots, N-r$ und analog für d^z_{j+r}. Die Konstanten C und K sind durch die Lipschitz-Konstante der rechten Seite f bestimmt sowie durch die Koeffizienten α_j und β_j des MSV.

Beweis: Wir beweisen die Behauptung nur für *explizite* MSV ($\beta_r = 0$) und nur im Fall $y : [0, T] \to \mathbb{R}$ (skalare ODE). Weiterhin wird o. B. d. A. $\alpha_r = 1$ angenommen.

Schritt (i): Darstellung der Differenzen durch eine vektorwertige Rekursion. Nach Definition von d^w_{j+r} und d^z_{j+r} erfüllen die Gitterfunktionen die Rekursion

$$\begin{aligned} w_{n+r} &= -\sum_{m=0}^{r-1} \alpha_m \, w_{n+m} + h \sum_{m=0}^{r-1} \beta_m \, f(t_{n+m}, w_{n+m}) + h\, d^w_{n+r} \\ z_{n+r} &= -\sum_{m=0}^{r-1} \alpha_m \, z_{n+m} + h \sum_{m=0}^{r-1} \beta_m \, f(t_{n+m}, z_{n+m}) + h\, d^z_{n+r} \end{aligned}$$

für $n \ge 0$. Mit der Abkürzung $e_n := w_n - z_n$ ergibt sich

$$\begin{aligned} e_{n+r} = &-\sum_{m=0}^{r-1} \alpha_m \, e_{n+m} \\ &+ \underbrace{h \sum_{m=0}^{r-1} \beta_m \left[f(t_{n+m}, w_{n+m}) - f(t_{n+m}, z_{n+m}) \right] + h \left[d^w_{n+r} - d^z_{n+r} \right]}_{=: b_{n+r}} \end{aligned} \tag{$*$}$$

für $n \ge 0$. Zusammen mit den trivialen Gleichungen

$$e_{n+m} = e_{n+m}, \quad m = 1, \ldots, r-1$$

ergibt sich aus ($*$) das Gleichungssystem in $\mathbb{R}^r$

$$\underbrace{\begin{pmatrix} e_{n+1} \\ e_{n+2} \\ \vdots \\ e_{n+r} \end{pmatrix}}_{E_{n+1}} = \underbrace{\begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -\alpha_0 & -\alpha_1 & \cdots & -\alpha_{r-1} \end{bmatrix}}_{A} \underbrace{\begin{pmatrix} e_n \\ e_{n+1} \\ \vdots \\ e_{n+r-1} \end{pmatrix}}_{E_n} + \underbrace{\begin{pmatrix} 0 \\ \vdots \\ 0 \\ b_{n+r} \end{pmatrix}}_{B_n}$$

[56] Wir benutzen diesen wie bereits in Lemma 3.15 analog zu Definition 4.7 hier für die *beliebige* Gitterfunktion w_h.

oder kurz:

$$E_{n+1} = A\,E_n + B_n, \quad n = 0, 1, \ldots \tag{$**$}$$

Schritt (*ii*): Abschätzungen

Die Matrix A ist die Begleitmatrix des ersten assoziierten Polynoms ϱ, und ϱ ist gerade das charakteristische Polynom von A.[57] Aufgrund der Nullstabilität des MSV gilt $R(A) \leq 1$ und alle Eigenwerte λ von A mit $|\lambda| = 1$ sind einfach. Nach Lemma 4.22 (b) existiert deshalb eine Vektornorm $\|\cdot\|_*$, sodass bzgl. der induzierten Matrixnorm gilt:

$$\|A\|_* = R(A) \leq 1.$$

Aus $(**)$ erhalten wir also die Abschätzung

$$\|E_{n+1}\|_* \leq \|E_n\|_* + \|B_n\|_* \quad \text{für } n \geq 0$$

und damit (wiederholtes Einsetzen)

$$\|E_n\|_* \leq \|E_0\|_* + \sum_{j=0}^{n-1} \|B_j\|_*. \tag{4.21}$$

Wir müssen noch $\|B_n\|_*$ abschätzen. Zur Verkürzung der Notation verwenden wir

$$\begin{aligned}
&d_{n+r} := |d^w_{n+r} - d^z_{n+r}| \\
&\Rightarrow \\
|b_{n+r}| &\overset{(*)}{\leq} h \overbrace{\max_{m=0,\ldots,r-1} |\beta_m|}^{=:\overline{\beta}} \sum_{m=0}^{r-1} |f(t_{n+m}, w_{n+m}) - f(t_{n+m}, z_{n+m})| + h\,d_{n+r} \\
&\leq h\,\overline{\beta}\,L \sum_{m=0}^{r-1} |e_{n+m}| + h\,d_{n+r} \\
&= h\,\overline{\beta}\,L\,\|E_n\|_1 + h\,d_{n+r}.
\end{aligned}$$

Da alle Normen auf $\mathbb{R}^r$ äquivalent sind, gilt mit einer Konstanten $\gamma > 0$

$$\gamma^{-1}\,\|z\|_* \leq \|z\|_1 \leq \gamma\,\|z\|_* \quad \text{für alle } z \in \mathbb{R}^r$$

und deshalb:

$$\begin{aligned}
\|B_n\|_* &\leq \gamma\,\|B_n\|_1 = \gamma\,|b_{n+r}| \\
&\leq \gamma\left(h\,\overline{\beta}\,L\,\|E_n\|_1 + h\,d_{n+r}\right) \\
&\leq \gamma^2\,\overline{\beta}\,L\,h\,\|E_n\|_* + \gamma\,h\,d_{n+r}.
\end{aligned}$$

Aus (4.21) folgt also die Abschätzung

$$\|E_n\|_* \leq \|E_0\|_* + \sum_{k=0}^{n-1} \gamma^2\,\overline{\beta}\,L\,h\,\|E_k\|_* + \sum_{k=0}^{n-1} \gamma\,h\,\underbrace{|d^w_{k+r} - d^z_{k+r}|}_{=d_{j+r}}.$$

Schritt (*iii*): Anwendung des diskreten Gronwall-Lemmas

[57] Beweis durch Entwicklung von $\det(A - \lambda\,I)$ nach der letzten Spalte und Induktion. **Beachte:** $\alpha_r = 1$

Mit Hilfe von Lemma 3.4 mit den Setzungen

$$a_n := \|E_n\|_*, \quad b_n := \|E_0\|_* + \sum_{k=0}^{n-1} \gamma\, h\, |d^w_{j+r} - d^z_{j+r}|, \quad \delta_k := \gamma^2\, \overline{\beta}\, L\, h$$

erhalten wir

$$\|E_n\|_* \le \big(\|E_0\|_* + \sum_{k=0}^{n-1} \gamma\, h\, |d^w_{k+r} - d^z_{k+r}|\big) \exp\big(\underbrace{\gamma^2\, \overline{\beta}\, L}_{K}\, t_n\big) \quad \text{für } n \ge 0.$$

Die Abschätzungen

$$\|E_0\|_* \le \gamma\, \|E_0\|_1 \le \gamma\, r\, \|E_0\|_\infty = \gamma\, r \max_{0\le i\le r-1} |e_i|$$

und

$$\|E_n\|_* \ge \gamma^{-1} \|E_n\|_1 \ge \gamma^{-1} |e_{n+r-1}|$$

zeigen

$$|e_{n+r-1}| \le \gamma \Big(\gamma\, r \max_{0\le i\le r-1} |e_i| + \sum_{j=0}^{n-1} \gamma\, h\, |d^w_{j+r} - d^z_{j+r}|\Big) \exp\left(K\, t_n\right)$$

für $n \ge 0$ bzw. nach Indexverschiebung $n \rightsquigarrow n - r + 1$

$$|e_n| \le \gamma \Big(\gamma\, r \max_{0\le i\le r-1} |e_i| + \sum_{j=0}^{n-r} \gamma\, h\, |d^w_{j+r} - d^z_{j+r}|\Big) \underbrace{\exp\left(K\, (t_n + (1-r)\, h)\right)}_{\le \exp(K\, t_n)}$$

für $n \ge r-1$. Dies zeigt die Behauptung (4.20) im betrachteten Spezialfall expliziter MSV und skalarer Gleichungen. Allgemeiner siehe auch Hermann, 2004, Satz 3.9. □

Nun kann die Umkehrung von Satz 4.21 bewiesen werden.

Satz 4.24 (Konvergenz von MSV, vgl. Satz 3.17)**.** Das lineare MSV (4.2) sei nullstabil und besitze die Konsistenzordnung $p \in \mathbb{N}$. Weiterhin seien die Startwerte $y_h(t_0), \ldots, y_h(t_{r-1})$ mit Konvergenzordnung p gemäß Definition 4.20 (c) gegeben. Dann gilt für die Näherungslösungen auf dem äquidistanten Gitter $\mathcal{T} = \{k\, h : k = 0, 1, \ldots, N := \lfloor T/h \rfloor\}$ (bei impliziten MSV für hinreichend kleine $h \le \overline{h}$) die Abschätzung

$$\begin{aligned} \|y_h(t_n) - y(t_n)\| &\le C\, \mathrm{e}^{K\, t_n} \Big(\max_{0\le k\le r-1} \|y_h(t_k) - y(t_k)\| + c\, t_n\, h^p\Big) \\ &\le \widetilde{C}\, \mathrm{e}^{K\, t_n} (1 + t_n)\, h^p \end{aligned} \tag{4.22}$$

für alle $n = 0, 1, \ldots, N$ und damit

$$\begin{aligned} \|y_h - y\|_{\infty,h} &:= \max_{0\le n\le N} \|y_h(t_n) - y(t_n)\| \\ &\le C\, \mathrm{e}^{K\, T} \Big(\max_{0\le k\le r-1} \|y_h(t_k) - y(t_k)\| + c\, T\, h^p\Big), \\ &\le \widetilde{C}\, \mathrm{e}^{K\, T} (1 + T)\, h^p. \end{aligned} \tag{4.23}$$

Beweis: Wir setzen $z_h = y_h$ und $w_h = y|_{\mathcal{T}}$ in Lemma 4.23 ein. Dann gilt nach Voraussetzung für die jeweiligen lokalen Fehler

$$d_j^z = 0$$
$$\|d_{j+r}^w\| = \|d(y(\cdot), t_{j+r}, h)\| \leq c\, h^p.$$

Aus Lemma 4.23 ergibt sich nun die behauptete Abschätzung (4.22)

$$\|y_h(t_n) - y(t_n)\| \leq C \Bigg(\underbrace{\max_{0\leq k\leq r-1} \|y_h(t_k) - y(t_k)\|}_{\text{Startwerte in Definition 4.20}} + c\, h^p \underbrace{\sum_{j=0}^{n-r} h}_{=t_n+(1-r)\,h\leq t_n} \Bigg) \exp\left(K\, t_n\right),$$

von der (4.23) eine direkte Konsequenz ist.

Die Konvergenzordnung der Startwerte garantiert den Rest.

□

Satz 4.25 (Erste Dahlquist-Schranke[58] (Dahlquist, 1956))**.** Für die Konsistenzordnung p eines nullstabilen r-Schritt linearen MSV gilt:

(a) $p \leq r + 2$ für r gerade

(b) $p \leq r + 1$ für r ungerade

(c) $p \leq r$, falls $\dfrac{\beta_r}{\alpha_r} \leq 0$ (also insbesondere für explizite MSV, $\beta_r = 0$).

Schrittzahl r	1	2	3	4	5
max. Ordnung p	2	4	4	6	6

Bemerkung 4.26 (zur ersten Dahlquist-Schranke)**.** (a) Die Schranken sind scharf, es gibt also jeweils MSV, für die „=“ gilt.

(b) Nullstabile MSV der maximalen Ordnung $p = r + 2$ (r gerade) sind symmetrisch in den Koeffizienten:

$$\alpha_k = -\alpha_{r-k} \quad \text{und} \quad \beta_k = \beta_{r-k},$$

z. B. die Milne-Methode (Beispiel 4.13).

Satz 4.27 (Nullstabilität bekannter Verfahren)**.** (a) Die expliziten und impliziten Adams-Verfahren sind nullstabil.

(b) Die BDF-Verfahren sind nullstabil für $1 \leq r \leq 6$ und instabil für $r \geq 7$.

Beweis: (a): Das erste assoziierte Polynom $\varrho(\mu) = \mu^r - \mu^{r-1} = \mu^{r-1}(\mu - 1)$ hat die Nullstellen $\mu = 1$ (einfach) und $\mu = 0$ $(r-1)$-fach. (b): siehe Hairer, Nørsett, Wanner, 1993, Kapitel III.1. □

[58] siehe Dahlquist, 1956 oder Hairer, Nørsett, Wanner, 1993, Theorem III.3.5

§ 4.3 Stabilitätsbegriffe bei Mehrschrittverfahren

In der Praxis beschreibt die Nullstabilität alleine das Verhalten eines MSV natürlich nicht vollständig.[59] Wir fordern wie bei ESV, dass die Näherungslösung unabhängig von der Schrittweite das gleiche qualitative Verhalten aufzeigt wie die exakte Lösung. Wir betrachten wieder die *Dahlquist'sche Testgleichung*

$$y' = \lambda\, y$$

mit $\lambda \in \mathbb{C}$ und $\mathrm{Re}(\lambda) < 0$. Die durch das lineare MSV (4.2) erzeugten Näherungen erfüllen die lineare homogene Δgl

$$\sum_{m=0}^{r} \alpha_m\, y_{i+m} - h\,\lambda \sum_{m=0}^{r} \beta_m\, y_{i+m} = 0,$$

deren Lösungen nach Satz 4.16 über die Nullstellen $\mu \in \mathbb{C}$ von $\varrho(\mu) - h\,\lambda\,\sigma(\mu)$ bestimmt sind. Lösungsfolgen $\{y_n\}$ sind genau dann betragsmäßig nicht-wachsend (für hinreichend große $n \in \mathbb{N}$), wenn alle Nullstellen $|\mu| \le 1$ erfüllen und die Nullstellen mit $|\mu| = 1$ einfach sind.

Definition 4.28 (Stabilitätspolynom, A-Stabilität, Stabilitätsgebiet, vgl. Definition 3.42)**.** Wir betrachten das lineare MSV (4.2).

(a) Die Funktion
$$\chi(\mu; z) = \varrho(\mu) - z\,\sigma(\mu)$$
heißt das **Stabilitätspolynom** (in der Variablen μ) des MSV.

(b) Die Menge
$$S = \{z \in \mathbb{C} : \text{Alle Nullstellen } \mu \text{ von } \chi(\mu; z) \text{ erfüllen } |\mu| \le 1, \text{ und alle mehrfachen Nullstellen erfüllen } |\mu| < 1\}$$
heißt **Gebiet der absoluten Stabilität** (**Stabilitätsgebiet**) des MSV.

(c) Das MSV heißt **absolut stabil** oder **A-stabil**, wenn $\mathbb{C}^- \subset S$ gilt.

Bemerkung 4.29. Die Nullstabilität eines MSV bedeutet gerade: $0 \in S$.

Bemerkung 4.30 (Wurzelortskurve)**.** Besser beschreibbar (und darstellbar) werden die Stabilitätsgebiete durch folgende Überlegung: Am Rand ∂S besitzt mindestens eine Nullstelle μ von $\chi(\mu; z)$ den Betrag 1, d. h., es gilt dort

$$\varrho(e^{i\varphi}) - z\,\sigma(e^{i\varphi}) = 0$$

für ein $\varphi \in [0, 2\pi]$. Die Menge

$$\Gamma := \{z \in \mathbb{C} : z = \frac{\varrho(e^{i\varphi})}{\sigma(e^{i\varphi})}, \quad \varphi \in [0, 2\pi]\}$$

(die sogenannte **Wurzelortskurve**) (*root locus curve*) enthält also ∂S.

[59] Jetzt kommt die Schrittweite h ins Spiel. Bisher hatten wir immer nur die linke Seite eines linearen MSV betrachtet, wo h nicht eingeht.

Beispiel 4.31 (Stabilitätsgebiete der expliziten Adams-Verfahren)**.** Die (expliziten) Adams-Bashforth-Verfahren sind *nicht* A-stabil. Ihre Stabilitätsgebiete bis $r = 5$ sind in Abbildung 4.2 dargestellt.

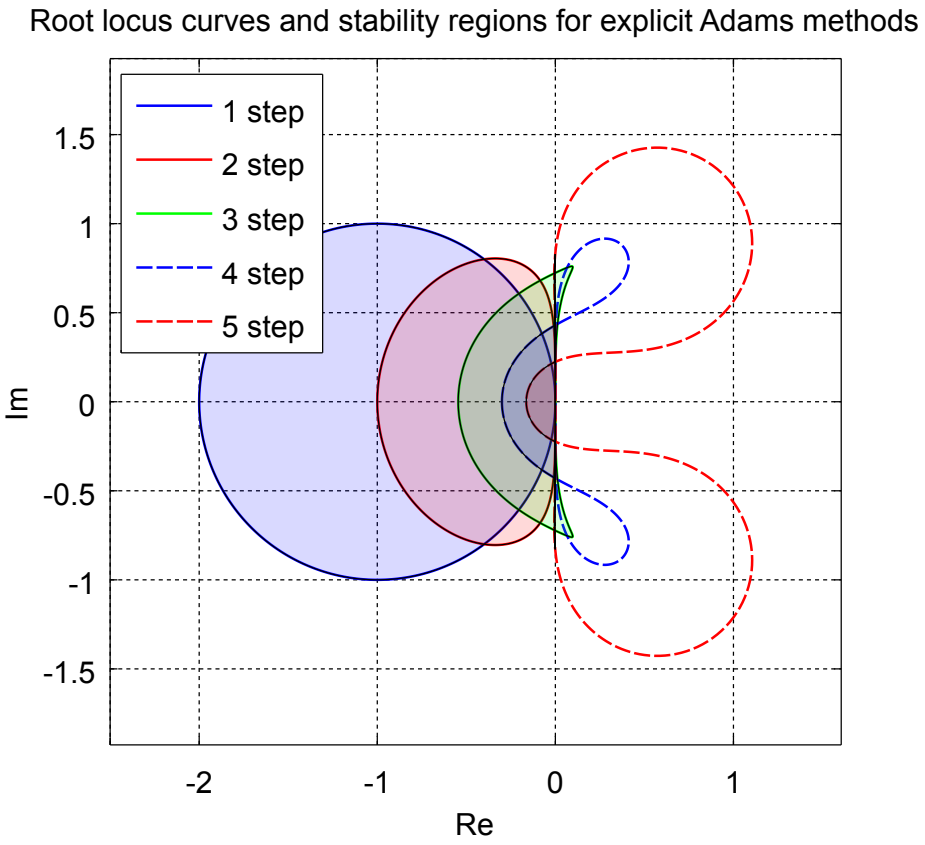


ABBILDUNG 4.2. Stabilitätsgebiete der (expliziten) Adams-Bashforth-Verfahren.

Der folgende Satz zeigt, dass nur wenige lineare MSV überhaupt A-stabil sein können.

Satz 4.32 (Zweite Dahlquist-Schranke[60] (Dahlquist, 1963))**.**

(a) Explizite konvergente MSV sind niemals A-stabil.

(b) Ein A-stabiles lineares MSV besitzt höchstens die Ordnung $p = 2$. Von allen A-stabilen linearen MSV der Ordnung 2 besitzt die implizite Trapezregel (Crank-Nicolson)[61]

$$y_{k+1} - y_k = \frac{h}{2}\big(f_{k+1} + f_k\big)$$

die kleinste Fehlerkonstante.

Beweis:

[60] siehe Dahlquist, 1963 oder Hairer, Nørsett, Wanner, 1993, Theorem V.1.4

[61] also das implizite Adams-Verfahren mit $r = 1$

(a) Das Stabilitätspolynom eines expliziten MSV lautet (beachte: $\alpha_r = 1$, $\beta_r = 0$)

$$\chi(\mu; z) = \mu^r + (\alpha_{r-1} - z\beta_{r-1})\,\mu^{r-1} + \ldots + (\alpha_1 - z\beta_1)\,\mu + (\alpha_0 - z\beta_0).$$

Aufgrund der Konsistenz gilt

$$\sum_{m=0}^{r} \alpha_m = 0 \quad \text{und} \quad \sum_{m=0}^{r} m\,\alpha_m - \beta_m = 0.$$

Dabei ist mindestens ein $\beta_j \neq 0$, da sonst $\mu = 1$ eine doppelte Nullstelle von $\varrho(.)$ wäre, und somit das Verfahren nicht nullstabil und nicht konvergent wäre (Satz 4.21). Für dieses j und $z \in \mathbb{R}$, $z \to -\infty$ strebt der Koeffizient von μ^j gegen ∞, also auch mindestens eine der Nullstellen, denn das Polynom hat die Darstellung

$$\chi_h(\mu) = (\mu - \mu_1(z))(\mu - \mu_2(z)) \cdots (\mu - \mu_r(z)).$$

Da die Nullstellen stetig von den Koeffizienten eines Polynoms abhängen, kann zu jedem $M > 0$ ein $z \in \mathbb{R}^-$ gefunden werden, so dass $\chi_h(\mu)$ eine Nullstelle μ mit $|\mu| > M$ hat.

(b) Siehe Hairer, Wanner, 1996, Theorem V.1.4.

□

Beispiel 4.33 (Stabilitätsgebiete der impliziten Adams-Verfahren)**.** Die (impliziten) Adams-Moulton-Verfahren sind nur für $r = 1$ (implizite Trapezregel) A-stabil. Ihre Stabilitätsgebiete sind jedoch wesentlich größer als die der expliziten Verfahren mit derselben Schrittzahl, siehe Abbildung 4.3.

Beispiel 4.34 (Stabilitätsgebiete der BDF-Verfahren)**.** Die BDF-Verfahren sind nur für $r = 1$ (implizites Euler-Verfahren) und $r = 2$ A-stabil. Ihre Stabilitätsgebiete bis $r = 6$ sind jeweils die Außengebiete der in Abbildung 4.4 dargestellten Wurzelortskurven. Für $r \geq 7$ sind diese Verfahren nicht einmal nullstabil, entsprechend enthalten die Stabilitätsgebiete nicht einmal die Null, siehe Abbildung 4.5 für $r = 7$. Das Verfahren wäre nur stabil für $h \geq \overline{h}$, was dem Gedanken $h \searrow 0$ widerspricht.

BDF-Verfahren werden (trotz fehlender A-Stabilität für $r \geq 3$) für steife Systeme erfolgreich eingesetzt.

Die A-Stabilität ist für MSV offenbar zu restriktiv (Zweite Dahlquist-Schranke, Satz 4.32). Einen Ausweg stellen die folgenden Stabilitätsbegriffe dar, die z. B. die Eigenschaften der BDF-Verfahren genauer beschreiben.

Definition 4.35 (A(α)-Stabilität)**.** Ein lineares MSV mit Stabilitätsgebiet S heißt **A(α)-stabil**, wenn gilt:

$$\{z \in \mathbb{C}^- : |\arg(z) - \pi| \leq \alpha\} \subset S.$$

Das heißt, der Sektor mit Öffnungswinkel $2\,\alpha$ gegen die negative reelle Achse ist Teil des Stabilitätsgebietes.

Beispiel 4.36. Der Öffnungswinkel 2α der A(α)-Stabilitätsgebiete für die nullstabilen BDF-Verfahren ist in folgender Tabelle angegeben:

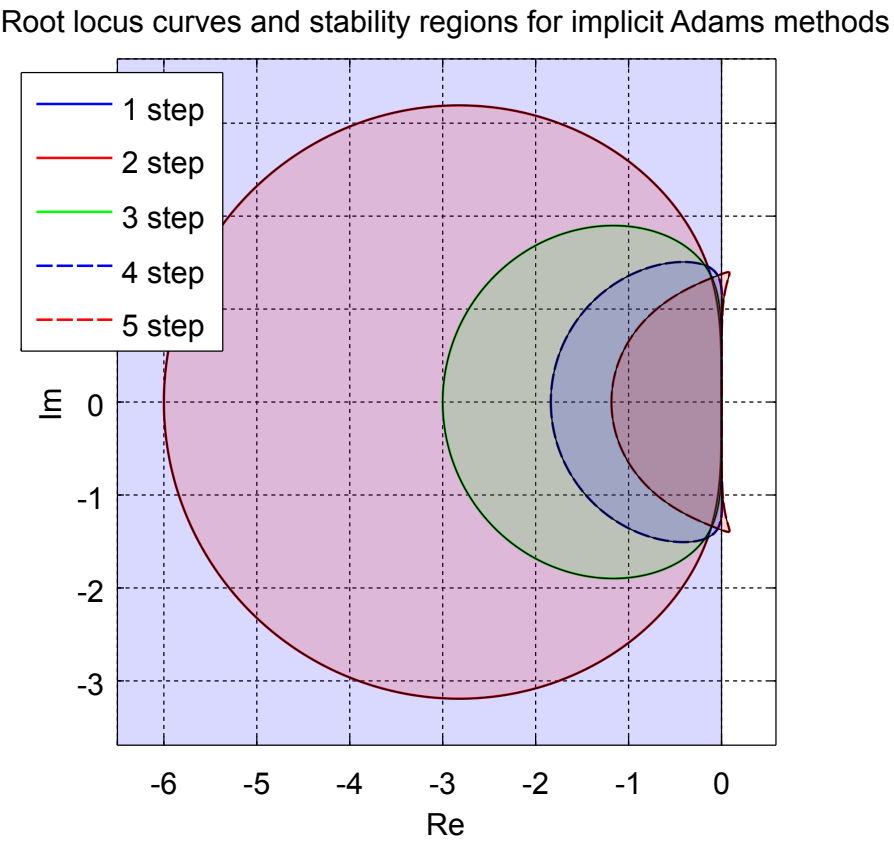


ABBILDUNG 4.3. Stabilitätsgebiete der (impliziten) Adams-Moulton-Verfahren.

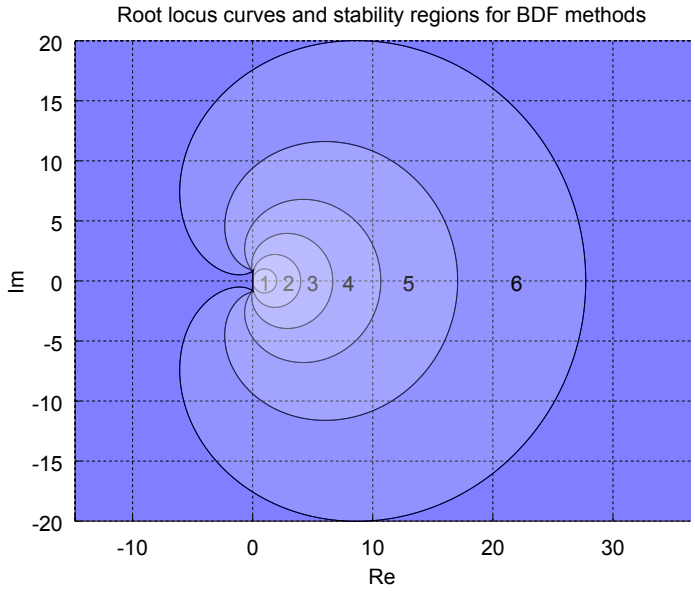


ABBILDUNG 4.4. Stabilitätsgebiete der (impliziten) BDF-Verfahren.

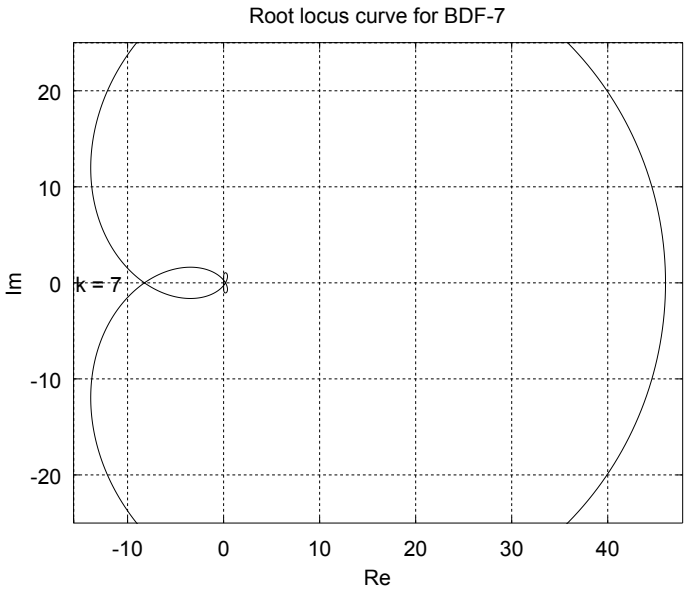


Abbildung 4.5. Die Wurzelortskurve des BDF-Verfahrens für $r = 7$ definiert ein leeres Stabilitätsgebiet.

Schrittzahl r	1	2	3	4	5	6
Winkel α	90°	90°	86.03°	73.35°	51.84°	17.84°

Die zugehörigen Stabilitätsgebiete sind in Abbildung 4.6 dargestellt.

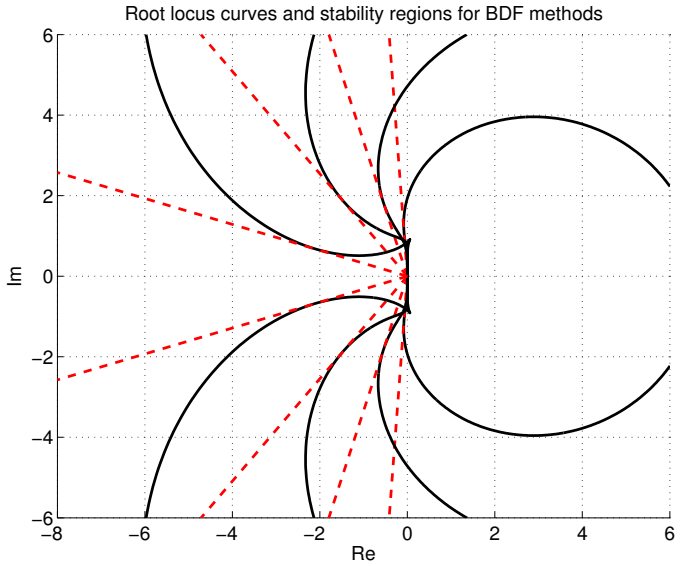


Abbildung 4.6. A(α)-Stabilität der BDF-Verfahren.

Definition 4.37 (Steif-Stabilität)**.** Ein lineares MSV mit Stabilitätsgebiet S heißt **steif-stabil**, wenn reelle Zahlen $a > 0$ und $c > 0$ existieren, sodass

$$S_1 := \{z \in \mathbb{C} : \mathrm{Re}(z) < -a\} \subset S,$$
$$S_2 := \{z \in \mathbb{C} : -a \leq \mathrm{Re}(z) < 0,\ |\mathrm{Im}(z)| \leq c\} \subset S$$

gelten, siehe auch Abbildung 4.7.

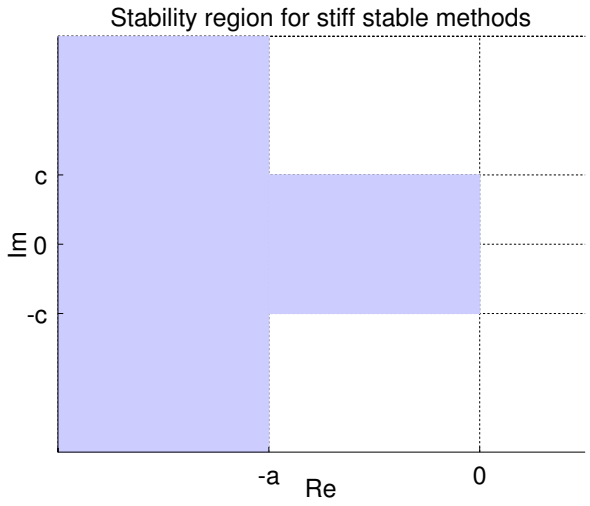


ABBILDUNG 4.7. Illustration der Gebiete S_1 und S_2 bei steif-stabilen MSV.

Bemerkung 4.38 (Relevanz der Steif-Stabilität)**.** Ein stark oszillierender Lösungsanteil (großer Imaginärteil des Eigenwertes) muss mit entsprechend kleiner Schrittweite aufgelöst werden, bis er hinreichend stark abgeklungen ist. Danach kann eine größere Schrittweite gewählt werden, falls kein neues Aufschaukeln stattfindet.

Es gibt noch einige weitere Stabilitätsbegriffe, siehe z. B. Hairer, Wanner, 1996.

§ 4.4 Praktische Aspekte bei Mehrschrittverfahren

Die Startwerte $y_0, y_1, \ldots, y_{r-1}$ eines r-schrittigen MSV der Ordnung p werden oft durch RKV der Ordnung p bestimmt, sodass die Bedingung in Definition 4.20 (c) sichergestellt ist. Die i. A. nichtlinearen Gleichungssysteme bei *impliziten* MSV können für hinreichend kleines h wie bei ESV (vgl. Satz 3.33) über eine Fixpunktiteration[62] (**Korrektor**) gelöst werden:

$$y_{k+r}^{(n+1)} = \underbrace{-\sum_{m=0}^{r-1} \alpha_m\, y_{k+m} + h \sum_{m=0}^{r-1} \beta_m f(t_{k+m}, y_{k+m})}_{\text{gegeben}} + h\,\beta_r f(t_{k+r}, y_{k+r}^{(n)}). \tag{4.24}$$

Dabei kann ein $\widehat{r}$-schrittiges *explizites* MSV (**Prädiktor**) zum Einsatz kommen, um eine Startschätzung $y_{i+r}^{(0)}$ vorzugeben:

$$y_{k+r}^{(0)} = -\sum_{m=0}^{\widehat{r}-1} \widehat{\alpha}_m\, y_{k+m+d} + h \sum_{m=0}^{\widehat{r}-1} \widehat{\beta}_m f(t_{k+m+d}, y_{k+m+d}) \tag{4.25}$$

[62] Alternativ könnte man auch das Newton-Verfahren verwenden.

mit $d := r - \widehat{r}$. Wir gehen wieder o. B. d. A. von $\alpha_r = \widehat{\alpha}_{\widehat{r}} = 1$ aus.

Für den ersten Fixpunktschritt mit $n = 0$ muss $f(t_{k+r}, y_{k+r}^{(0)})$ ausgewertet werden.

Damit kann $y_{k+r}^{(1)}$ aus (4.24) bestimmt werden. Ggf. werden weitere Iterationen ausgeführt, wobei jede eine zusätzliche f-Auswertung für den letzten Term in (4.24) benötigt.

Definition 4.39 (Prädiktor-Korrektor-Verfahren)**.** Es seien die Startwerte $y_0, y_1, \ldots, y_{r-1}$ und $f_0, f_1, \ldots, f_{r-1}$ mit $f_k = f(t_k, y_k)$ gegeben. Weiter seien (α, β) die Koeffizienten eines *impliziten* r-Schritt MSV (**Korrektor**) und $(\widehat{\alpha}, \widehat{\beta})$ die Koeffizienten eines *expliziten* $\widehat{r}$-Schritt MSV (**Prädiktor**). In der Regel ist $d := r - \widehat{r} \geq 0$. Ein MSV, in dem die neue Näherung y_{k+r}, $k \geq 0$, gemäß

[**P**redict] $y_{k+r} := -\sum_{m=0}^{\widehat{r}-1} \widehat{\alpha}_m \, y_{k+m+d} + h \sum_{m=0}^{\widehat{r}-1} \widehat{\beta}_m \, f_{k+m+d}$

for $n = 0, 1, \ldots, M-1$ **do**

[**E**valuate] $f_{k+r} := f(t_{k+r}, y_{k+r})$

[**C**orrect] $y_{k+r} := -\sum_{m=0}^{r-1} \alpha_m \, y_{k+m} + h \sum_{m=0}^{r} \beta_m \, f_{k+m}$

end for

bestimmt wird, heißt ein **P(EC)^M-Verfahren**, wobei $M \geq 1$ ist. Wenn nach dem letzten Korrektorschritt noch eine weitere f-Auswertung durchgeführt wird, heißt es **P(EC)^{M}E-Verfahren** (von **E**valuate).

Beachte: Prädiktor-Korrektor-Verfahren sind *explizite nichtlineare* MSV.

Satz 4.40 (Konsistenzordnung von Prädiktor-Korrektor-Verfahren)**.** Der Prädiktor habe Konsistenzordnung $\widehat{p}$ und der Korrektor die Konsistenzordnung p. Dann haben das P(EC)^M- und das P(EC)^{M}E-Verfahren für $M \geq 1$ die Konsistenzordnung

$$p_M = \min\{\widehat{p} + M, p\}.$$

Ist der Korrektor nullstabil, so konvergieren die Verfahren mit der Konvergenzordnung p_M.

Beweis: Siehe Hermann, 2004, Satz 3.13. □

Bemerkung 4.41 (zu Satz 4.40)**.**

(a) Jede Korrektoriteration erhöht, ausgehend von $\widehat{p}$, die Konsistenzordnung um eins, bis die Konsistenzordnung p des Korrektors erreicht ist.

(b) Gebräuchlich sind **ABM-Verfahren**:

- Prädiktor: $\widehat{r}$-Schritt-AB-Verfahren (Ordnung $\widehat{p} = \widehat{r}$)
- Korrektor: r-Schritt-AM-Verfahren (Ordnung $p = r + 1$)
- oft $\widehat{r} = r$ und $M = 1$ Korrektorschritte

(c) Jeder **E**valuate-Schritt benötigt eine f-Auswertung.

(d) Der Prädiktor darf instabil sein!

Beispiel 4.42 (ABM-Verfahren)**.**

(a) Das **ABM33**-Verfahren vom Typ P(EC) lautet:

$$\widehat{y}_{k+3} := y_{k+2} + \frac{h}{12}\left(23f_{k+2} - 16f_{k+1} + 5f_k\right) \qquad (\widehat{r} = 3)$$

$$f_{k+3} := f(t_{k+3}, \widehat{y}_{k+3})$$

$$y_{k+3} := y_{k+2} + \frac{h}{24}\left(9f_{k+3} + 19f_{k+2} - 5f_{k+1} + f_k\right) \qquad (r = 3)$$

(b) Das **ABM11**-Verfahren vom Typ P(EC)E (Prädiktor: explizites Euler-Verfahren, Korrektor: implizite Trapezregel) ist das Verfahren von Heun.

Beispiel 4.43 (Stabilitätsgebiete der ABM-PECE-Verfahren)**.** Die ABM-PECE-Verfahren bilden einen Kompromiss zwischen expliziten und impliziten Verfahren, entsprechend sind ihre Stabilitätsgebiete auch kleiner als bei den impliziten Verfahren, jedoch größer als bei den expliziten. Abbildung 4.8 zeigt die Stabilitätsgebiete für P(EC)^{1}E-Verfahren für verschiedene Schrittzahlen $r = \widehat{r}$.

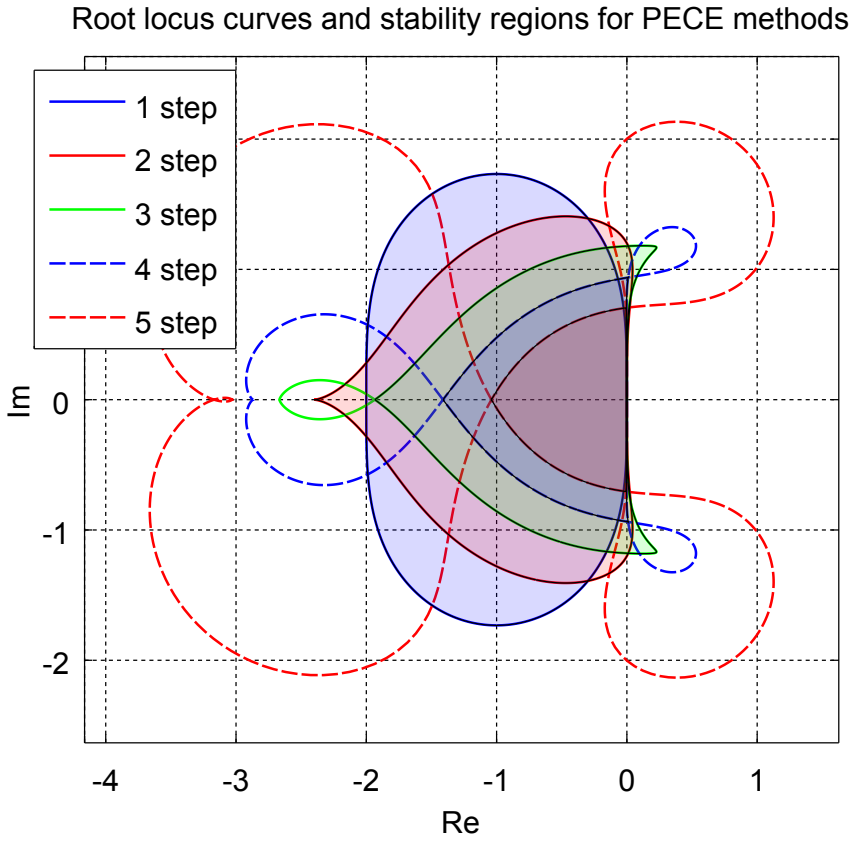


ABBILDUNG 4.8. Stabilitätsgebiete der ABM-P(EC)^{1}E-Verfahren.

Bemerkung 4.44 (MSV in der Praxis). (a) Man kann auch MSV mit variablen Schrittweiten ausführen und eine Fehlerschätzung vornehmen, um die Schrittweiten automatisch zu steuern. In der Praxis verwendet man oft Prädiktor-Korrektor-Verfahren mit zwei verschiedenen Korrektor-Verfahren benachbarter Ordnungen zur Fehlerschätzung.[63]

(b) Die Schrittweitensteuerung lässt sich sogar mit einer automatischen Steuerung der Ordnung (also der Schrittanzahl r) verbinden. Strategie: Schätze den lokalen Fehler mit *allen* zur Verfügung stehenden eingebetteten Paaren von MSV und wähle für den nächsten Zeitschritt diejenige Ordnung, die die größte Schrittweite (bei gleichem Fehler) erlaubt.

(c) MSV mit variabler Ordnung und Ordnungssteuerung sind selbststartend, benötigen also nur y_0. Der erste Schritt wird mit einem Einschrittverfahren ausgeführt und dann die mögliche Schrittzahl sukzessive erhöht.

Bemerkung 4.45 (Mehrschrittverfahren in MATLAB). In MATLAB sind folgende Mehrschrittverfahren (alle mit Schrittweitensteuerung) implementiert:

- `ode113` ist ein Prädiktor-Korrektor-Verfahren vom Typ PECE auf ABM-Basis (Adams-Bashforth-Moulton) mit Schrittweiten- und Ordnungssteuerung (bis Ordnung 12).
- `ode15s` ist ein Mehrschrittverfahren auf Basis von BDF- bzw. NDF-Formeln mit Schrittweiten- und Ordnungssteuerung.

Beide sind selbststartend: Als Startwert wird nur y_0 benötigt, da der erste Schritt mit einem Einschrittverfahren ausgeführt wird, der zweite höchstens mit einem Zweischrittverfahren usw.

§ 5 Unstetige Galerkin-Verfahren

Galerkin-Verfahren werden vorwiegend zur Lösung partieller (elliptischer) Dgl eingesetzt, siehe Vorlesung *Numerik partieller Differentialgleichungen*. Sie lassen sich jedoch auch auf AWP mit gewöhnlichen Dgl anwenden und bieten einen grundsätzlich anderen Zugang als ESV und MSV.

Problemstellung: Wir werden uns hier auf den affin linearen Fall beschränken. Das bedeutet, dass die betrachteten Differentialgleichungen die Form

$$y'(t) + A(t)y(t) = f(t), \quad t \in (0,T], \qquad y(0) = y_0. \tag{5.1}$$

besitzen. Die Lösung ist dabei eine Funktion $y\colon \mathbb{R} \supset I \to X$ mit einem Vektorraum X. Dabei sind folgende Anwendungen interessant:

- $X = \mathbb{R}^n$ (gewöhnliches Differentialgleichungssystem)

  Dann ist $A(t) \in \mathbb{R}^{n\times n}$ eine Matrix.

- X ist ein Funktionenraum mit Elementen $v\colon \Omega \subset \mathbb{R}^n \to \mathbb{R}$, wobei Ω ein beschränktes n-dimensionales Gebiet ist.

[63] Dann werden für das bessere Verfahren mehr Korrektorschritte ausgeführt, sonst profitiert das bessere Verfahren nicht, siehe Satz 4.40.

Wir können dann für A einen beliebigen linearen (Differential)operator wählen. Von Interesse ist häufig $A = -\Delta := -\frac{\partial^2}{\partial x_1^2} - \ldots - \frac{\partial^2}{\partial x_n^2}$, das bedeutet wir betrachten die *partielle* Differentialgleichung

$$\begin{cases} \partial_t y(x,t) - \Delta y(x,t) = f(x,t) & \text{für } x \in \Omega, t \in (0,T], \\ y(x,t) = 0 & \text{für } x \in \partial\Omega,\ t \in (0,T], \\ y(x,0) = y_0(x) & \text{für } x \in \Omega. \end{cases} \tag{5.2}$$

Diese Gleichung wird häufig als *Wärmeleitungsgleichung* bezeichnet und ist ein Beispiel für ein *parabolisches Problem*. Dabei ist $y(x,t)$ die Temperatur zum Zeitpunkt t im Ortspunkt x.

Zur Lösung derartiger Gleichungen verwendet man häufig unstetige Galerkin-Verfahren für die Zeitintegration, und ein geeignetes numerisches Verfahren für partielle Differentialgleichungen zur Diskretisierung bezüglich der Ortsvariablen (z.B. Finite Elemente, Finite Differenzen, ...).

Im Folgenden lassen wir ggf. die Ortskoordinate weg und interpretieren $y(t)$ als Element von X, d. h. als Vektor oder Funktion von x.

Idee: Multipliziere die Dgl mit einer sogenannten **Testfunktion** $v \in X$ und integriere die Gleichung in der Zeit. Damit erhalten wir die **Variationsformulierung**:

Finde $y \in C^1(I;X) \cap C(I;Y)$, so dass gilt:

$$\int_I (y'(t) + A(t)\,y(t) - f(t)) \cdot v(t)\mathrm{d}t + (y(0) - y_0) \cdot v(0) = 0 \quad \forall v \in C(I;X). \tag{5.3}$$

Beachte, dass die Anfangsbedingung in dieser Formulierung ebenfalls "erzwungen" wird. Hierbei ist Y ebenfalls ein Funktionenraum, welcher die Beziehung $Ay \in X$ für alle $y \in Y$ erfüllt. Im Beispiel (5.2) wäre $X = C(\overline{\Omega})$ sowie $Y = C^2(\Omega) \cap C(\overline{\Omega})$ eine sinnvolle Wahl. Das Skalarprodukt im Raum X ist definiert durch

$$u \cdot v := \int_\Omega u(x)\,v(x)\,\mathrm{d}x.$$

Man kann in (5.3) noch dazu übergehen, verallgemeinerte (distributionelle) Ableitungen zu verwenden und damit die Glattheitsanforderungen $y \in C^2$ zu senken, um auch unstetige rechte Seiten f zuzulassen. Man spricht dann von einer **schwachen Formulierung**. (Wird hier nicht betrachtet.)

Lemma 5.1. Die Funktion $y \in C^1(I;X) \cap C(I;Y)$ ist genau dann Lösung von (5.1), wenn sie auch Lösung von (5.3) ist.

Beweis: "$\Rightarrow$" ist klar. Um "$\Leftarrow$" zu zeigen verwendet man als Testfunktionen angenäherte Dirac-Distributionen $v(t) = \delta_\varepsilon(t - \widehat{t})$ verwendet, z. B.

$$\delta_\varepsilon(t) := \begin{cases} -\frac{1}{\varepsilon^2}|t| + \frac{1}{\varepsilon} & \text{für } |t| \le \varepsilon \\ 0 & \text{für } |t| > \varepsilon \end{cases}$$

(siehe Abbildung 5.1) und $\varepsilon \searrow 0$ gehen lässt (Hauptsatz der Variationsrechnung), um $y'(\widehat{t}) + Ay(\widehat{t}) = f(\widehat{t})$ für alle $\widehat{t} \in [0,T]$ zu erhalten. □

Die Idee eines **Galerkin-Verfahrens** besteht allgemein darin, in einer Variationsformulierung wie (5.3) die unendlichdimensionalen Räume für Ansatzfunktion y und Testfunktion v durch (dieselben) endlichdimensionale Räume zu ersetzen.

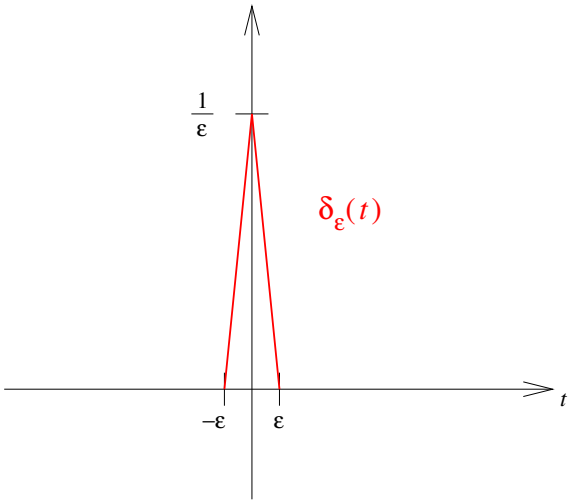


Abbildung 5.1. Angenäherte Dirac-Distributionen $\delta_\varepsilon(t)$.

Für den ODE-Fall haben sich **unstetige** oder **Discontinuous-Galerkin-Verfahren** (**DG-Verfahren**) als besonders günstig erwiesen. Dabei stimmen Ansatz- und Testraum überein und bestehen aus unstetigen Funktionen. Die Unstetigkeit der Testfunktionen ermöglicht dabei eine sukzessive Lösung der Variationsformulierung Zeitschritt für Zeitschritt, wie wir gleich sehen werden.

Wir definieren dazu eine Zerlegung von $I = [0, T]$

$$0 = t_0 < t_1 < t_2 < \ldots < t_N = T$$

und halboffene Teilintervalle

$$I_k := (t_k, t_{k+1}] \quad \text{mit Längen } h_k := t_{k+1} - t_k, \quad i = 0, 1, \ldots, N-1$$

und setzen wieder

$$h := \max_{k=0,\ldots,N-1} h_k,$$

siehe Abbildung 5.2 (a). Wir schränken den Ansatz- und Testraum in (5.3) weiter ein und wählen den Raum der stückweise glatten Funktionen

$$V := \{v : [0, T] \to X : v_{|I_k} \in C_c^1(I_k; X) \cap C_c(I_k; Y) \text{ für } k = 0, \ldots, N-1\},$$

siehe Abbildung 5.2 (b). Dabei bezeichnet $C_c^k(I_k)$, $k \in \{0, 1\}$ den Raum der auf I_k stetigen bzw. stetig diffbaren Funktionen (mit einseitigen Grenzwerten und Ableitungen im rechten Endpunkt t_{k+1}), die stetig diffbar zum linken Randpunkt t_k fortgesetzt werden können, d. h., dass dort die Funktion und ihre Ableitung endliche rechtsseitige Grenzwerte besitzen.

Für Funktionen in V führen wir die folgende Notation ein:

$$\begin{aligned} v_k^+ &:= \lim_{t \searrow t_k} v(t) \quad \text{rechtsseitiger Grenzwert bei } t_k \\ v_k^- &:= \lim_{t \nearrow t_k} v(t) \quad \text{linksseitiger Grenzwert bei } t_k \\ [v]_k &:= v_k^+ - v_k^- \quad \text{Sprung bei } t_k \end{aligned}$$

für $i = 0, \ldots, N-1$ mit der Konvention $v_0^- := v(0)$.

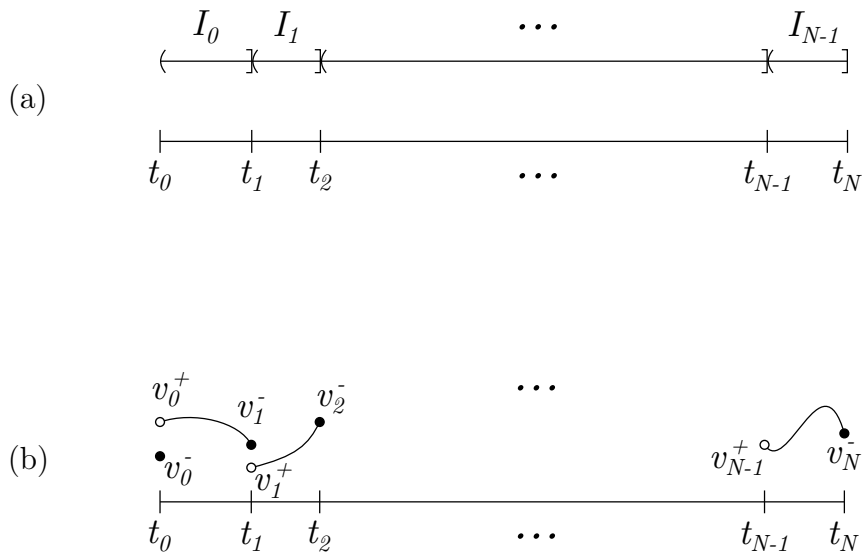


ABBILDUNG 5.2. Ansatz DG-Verfahren: (a) Unterteilung in Teilintervalle, (b) stückweise glatte Ansatzfunktionen.

Die Variationsformulierung (5.3) ist wegen der Unstetigkeit der Ansatz- und Testfunktionen $y, v \in V$ nicht mehr äquivalent zu (5.1) und muss erweitert werden. Eine geeignete Formulierung lautet

Definition 5.2. Das Problem
Finde $y \in V$, sodass für alle $v \in V$ gilt:

$$\sum_{k=0}^{N-1} \left[\int_{I_k} \big(y'(t) + A(t)\,y(t) - f(t)\big) \cdot v(t)\,\mathrm{d}t + [y]_k \cdot v_k^+ \right] + (y(0) - y_0) \cdot v_0^- = 0 \tag{5.4}$$

wird als **erweiterte Variationsformulierung** bezeichnet.

Lemma 5.3. Die Funktion $y \in V$ ist genau dann Lösung von (5.1), wenn sie auch Lösung von (5.4) ist.

Beweis: Der Beweis funktioniert analog zu dem von Lemma 5.1, jedoch muss zusätzlich mit Funktionen $v_{k,\varepsilon} \in V$ getestet werden, die die Eigenschaft

$$v_{k,\varepsilon}^+ = 1 \quad \text{und} \quad v_{k,\varepsilon}(t) \overset{\varepsilon \to 0}{\longrightarrow} 0 \quad \text{für alle } t \neq t_k$$

haben, siehe Abbildung 5.3. Für $\varepsilon \searrow 0$ ergibt dies

$$\int_{I_k} (y'(t) + A(t)y(t) - f(t)) \cdot v_{k,\varepsilon}\,\mathrm{d}t \to 0 \text{ for } \varepsilon \searrow 0,$$
$$(y(0) - y_0) \cdot v_{0,\varepsilon}^- = 0$$

und aufgrund von $v_{k,\varepsilon}^+ = 1$ folgt $[y]_k = 0$ für alle $k = 0, \ldots, N-1$. Damit ist die Lösung stetig auf $[0, T]$. Da f stetig ist und y auf allen Intervallen I_k die Dgl (5.1) punktweise erfüllt, muss y sogar stetig differenzierbar sein. [64] □

[64] Ein Beweis ist auch in Sheet 11, Exercise 28 zu finden.

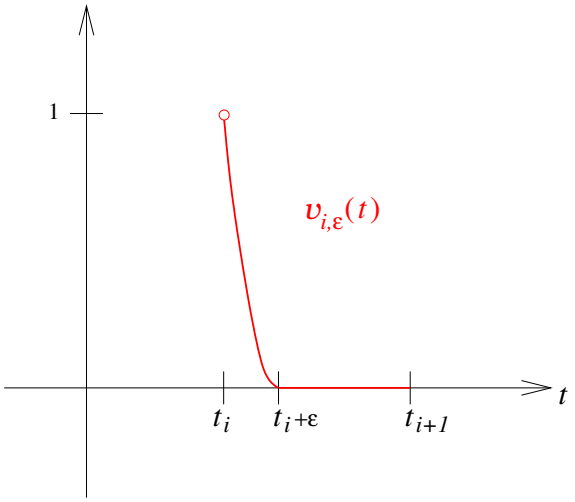


Abbildung 5.3. Testfunktionen $v_{k,\varepsilon}(t)$.

Zur Diskretisierung von (5.4) führen wir Teilräume $V_{h,p} \subset V$ von stückweisen Polynomen ein:

$$V_{h,p} := \{v : [0,T] \to X : v_{|I_i} \in \mathcal{P}_p(I_k;Y) \text{ für } k = 0,\dots,N-1\} \subset V,$$

wobei $\mathcal{P}_p(I_k;Y)$ den Raum der Polynome auf I_k vom Höchstgrad $p \in \mathbb{N}_0$ mit Werten in Y bezeichnet.[65]

Beispiel 5.4 (Siehe auch Abbildung 5.4)**.**

- Der Raum $V_{h,0}$ beinhaltet Funktionen der Form
$$y|_{I_k} = y_{k+1}^-, \qquad k = 0,\dots,N-1, \quad y(0) = y_0^-,$$
mit Koeffizienten $y_k^- \in Y$ für $k = 0,\dots,N$.
- Der Raum $V_{h,1}$ beinhaltet Funktionen der Form
$$y(t)|_{I_k} = y_k^+ \frac{t_{k+1}-t}{h_k} + y_{k+1}^- \frac{t-t_k}{h_k}, \qquad k = 0,\dots,N-1$$
mit Koeffizienten $y_k^- \in Y$ für $k = 1,\dots,N$ und $y_k^+ \in Y$ für $k = 0,\dots,N-1$.

Die Dimension von $V_{h,p}$ ist $\big(N\,(p+1)+1\big)\dim(Y)$. Falls Y unendlich-dimensional ist, dann sprechen wir lediglich von einer *Semidiskretisierung in der Zeit*. In diesem Fall muss anschließend eine weitere Diskretisierung des Ortes vorgenommen werden, um eine Lösung numerisch zu berechnen.

Die **DG-Diskretisierung** von (5.4) führt auf

$$\text{Finde } y_h \in V_{h,p}, \text{ sodass (5.4) für alle } v_h \in V_{h,p} \text{ gilt.} \tag{5.5}$$

Man spricht auch vom **DG(p)-Verfahren**.

Weitere Vorgehensweise:

[65] Beispiel: ein Element $p \in \mathcal{P}_1(I_k;Y)$ besitzt die Darstellung $p = a_0 + a_1\,t$, wobei die Koeffizienten $a_0, a_1 \in Y$ abhängig von der Anwendung reelle Zahlen, Vektoren oder Funktionen aus $C^2(\Omega) \cap C(\overline{\Omega})$ sind.

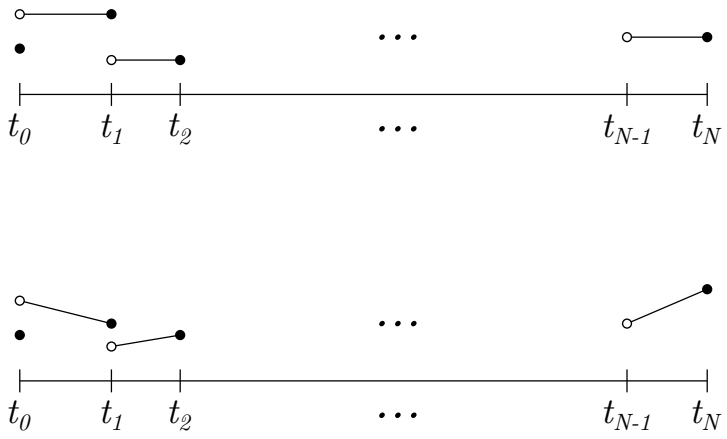


ABBILDUNG 5.4. Illustration von $V_{h,p}$ für $p = 0$ (oben) und $p = 1$ (unten).

(a) Teste mit möglichst simplen Basisfunktion v_h, welche nur auf einem I_k ungleich 0 sind.

(b) Rechne unbekannten Größen $y^{\pm}_{h,k}$ sukzessive aus. Beachte, dass sich aufgrund der Unstetigkeit der *Test*funktionen die Gleichungen auf den einzelnen Intervallen I_k beinahe vollständig entkoppeln.

Beispiel 5.5 (Das **DG(0)-Verfahren**[66])**.** Wir suchen die unbekannten Koeffizienten y^-_k für $k = 0, \dots, N$. Durch Testen von (5.5) mit $v_h|_{I_k} \equiv 0$, $k = 0, \dots, N-1$ und $v^-_0 = 1$ erhalten wir die Gültigkeit der Anfangsbedingung

$$y^-_0 = y_0.$$

Anschließend testen wir die Variationsformulierung (5.3) nacheinander für $k = 0, \dots, N-1$ mit

$$v_h \in V_{h,0}: \quad v^-_{\ell+1} = \begin{cases} 1, & \text{für } \ell = k, \\ 0, & \text{für } \ell \neq k. \end{cases}$$

Die Funktion v_h verschwindet überall, außer auf I_k. Dann erhalten wir wegen $y'_h|_{I_k} = 0$ und $[y]_k = y^-_{k+1} - y^-_k$, sowie $v^+_k = v^-_{k+1} = 1$

$$\int_{I_k} \Big(A(t) \underbrace{y^-_{k+1}}_{\text{ges.}} - f(t) \Big) \mathrm{d}t + (y^-_{k+1} - \underbrace{y^-_k}_{\text{geg.}}) = 0.$$

Dabei kann jeder Koeffizient y^-_{k+1} aus dem Letzten y^-_k (y^-_0 erhalten wir aus der Anfangsbedingung) berechnet werden. Umstellen nach der Unbekannten y^-_{k+1} ergibt

$$(I + \int_{I_k} A(t)\,\mathrm{d}t) y^-_{k+1} = y^-_k + \int_{I_k} f(t)\mathrm{d}t.$$

Die Integrale auf der rechten Seite werden außdem durch eine Quadraturformel approximiert. Verwendet man die **Rechtecksregel** $\int_{I_k} f(t)\mathrm{d}t \approx h_k f(t_{k+1})$, dann lautet unser Verfahren

$$(I + h_k\, A(t_{k+1})) y^-_{k+1} = y^-_k + h_k f(t_{k+1}). \tag{5.6}$$

Anschließend kann man die Lösung y_h aus Linearkombination der Basisvektoren "zusammenstückeln":

$$y_h(0) = y_0^-, \qquad y_h|_{I_k} = y_{k+1}^-,\ k = 0, \dots, N-1.$$

Beobachtung: Das durch (5.6) konstruierte Verfahren entspricht gerade dem implizite Euler-Verfahren aus Beispiel 3.11 b):

$$y_{k+1} = y_k + h_k \underbrace{f(t_{k+1}, y_{k+1})}_{:= f(t_{k+1}) - A(t_{k+1})\, y_{k+1}} .$$

Satz 5.6 (A-priori-Fehlerabschätzung)**.** Angenommen die Lösung von (5.1) gehöre zu $C^{p+1}(I;X) \cap C^p(I;Y)$. Dann gilt für den Fehler der Näherungslösungen des DG-Verfahrens (5.5) auf einem beliebigen Gitter $\{0 = t_0, t_1, \dots, t_N = T\}$ der Klasse $\mathcal{T}_h$ mit $h \leq \overline{h}$ die Abschätzung

$$\sup_{t\in[0,T]} \|y(t) - y_h(t)\| \leq K \max_{0\leq k\leq N-1} \left\{ h_k^{p+1} \sup_{t\in I_k} \|y^{(p+1)}(t)\| \right\} \tag{5.7}$$

mit einer i. A. exponentiell von L und T abhängigen Konstante $K = K(L,T)$.

Beweis: Siehe Rannacher, 2008, Satz 4.3 oder Schötzau, Schwab, 2000, Theorem 3.6. □

Vorteile der Variationsformulierung und des DG-Verfahrens:

- Im Gegensatz zu ESV und MSV wird die diskrete Lösung y_h hier als eine *globale* Gitterfunktion ganz $[0,T]$ betrachtet. Dies erlaubt eine direkte Betrachtung des globalen Fehlers, siehe Gleichung (5.7). Der globale Fehler von ESV und MSV können natürlich auf ähnliche Art und Weise geschätzt werden.
- Die variationelle Formulierung erlaubt die Schätzung und Kontrolle des *globalen* Fehlers, nachdem die gesamte Lösung auf einem zuvor bestimmten Gitter $\mathcal{T}$ berechnet wurde. Man spricht von einer Fehlerschätzung *a posteriori*. Der Wert des DG Verfahrens besteht in der Schätzung der Residuen. Diese erlauben eine genauere Schätzung des Fehlers in jedem Schritt, und somit eine effizientere Fehlerschätzung Rannacher, 2008.

Bemerkung 5.7 (zu Galerkin-Verfahren)**.** Als Alternative zur DG-Diskretisierung (5.5) sind auch CG(p)–DG($p-1$)-Ansätze gebräuchlich mit $p \in \mathbb{N}$. Dabei wird weiterhin $v_h \in V_{h,p}^0$ gewählt (DG), aber y_h wird zusätzlich als stetig (*continuous*) vorausgesetzt (**CG**). Da Ansatz- und Testraum nun verschieden sind, spricht man von einem **Petrov-Galerkin-Verfahren**. Auch in diesem Fall ergibt sich ein Zeitschrittverfahren. Das einfachste Beispiel ist das CG(1)–DG(0)-Verfahren mit einem stückweise linearen und global stetigen Ansatz für y_h und stückweise konstanten Testfunktionen v_h. Verwendet man zur Berechnung der Integrale

$$\int_{I_k} f(t) \cdot v_i \, \mathrm{d}t$$

[66] In der Übung sollten wir auch DG(1) herleiten

die Trapezregel, so entsteht die bereits aus Beispiel 3.24 und Beispiel 3.49 bekannte implizite Trapezregel (das Crank-Nicolson-Verfahren). [67]

§ 6 Kollokationsverfahren und IRKV

Viele (aber nicht alle) IRKV können als Kollokationsverfahren hergeleitet werden, daher besprechen wir kurz diese Klasse von Verfahren.

Die zugrundeliegende Idee von Kollokationsverfahren ist, dass man die gesuchte Funktion ansetzt als zu einem endlichdimensionalen Funktionenraum gehörend (z. B. Polynome vom Höchstgrad m) und fordert dass Gleichungen (i. A. Dgl, auch partielle Dgl, oder Integralgleichungen) in bestimmten Punkten (den **Kollokationspunkten**) erfüllt werden. Die Anzahl der Kollokationspunkte wird dabei so gewählt dass alle freien Parameter des Ansatzes durch die entstehenden Gleichungen bestimmt werden.

Für das AWP

$$\begin{aligned} y_h'(t) &= f(t, y_h(t)) && \forall t \in [t_k, t_k + h] \\ y_h(t_k) &= y_k \end{aligned}$$

wählt man so z. B. den Ansatz $y_h = p_m \in \Pi_m$ (Polynome vom Höchstgrad m),

$$p_m'(t_k + \alpha_j h) = f(t_k + \alpha_j h, p_m(t_k + \alpha_j h)), \qquad j = 1, \dots, m \tag{6.1}$$

$$p_m(t_k) = y_k. \tag{6.2}$$

Dies sind $m+1$ Bedingungen an die $m+1$ unbekannten Koeffizienten des Polynoms p_m. Mit der Wahl paarweise verschiedener $\alpha_j \in [0, 1]$, $j = 1, \dots, m$, wird also ein ESV festgelegt mittels

Definition 6.1 (Kollokationsverfahren)**.**

$$y_{k+1} = p_m(t_k + h).$$

[67] siehe auch Becker, Meidner, Vexler, 2007

Beispiel 6.2. Für $m = 1$ erhält man den Ansatz

$$p_1(t) = y_k + (t - t_k)k.$$

Aus

$$\begin{aligned} k = p_1'(t_k + \alpha_1 h) &= f(t_k + \alpha_1 h, p_1(t_k + \alpha_1 h)) \\ &= f(t_k + \alpha_1 h, y_k + \alpha_1 hk) \end{aligned}$$

erhält man

- für $\alpha_1 = 0$ das explizite Euler-Verfahren (Ordnung 1)
- für $\alpha_1 = 1$ das implizite Euler-Verfahren (Ordnung 1)
- für $\alpha_1 = 1/2$ die implizite Mittelpunktsregel (Ordnung 2), siehe Beispiel 3.24 (e).

Zusammenhang zu IRKV[68]:

Satz 6.3. Das Kollokationsverfahren aus Definition 6.1 ist äquivalent zu einem m-stufigen IRKV mit den Koeffizienten α_j wie gegeben und

$$\beta_{j\ell} := \int_0^{\alpha_j} L_\ell(s)\,\mathrm{d}s, \qquad \gamma_j := \int_0^1 L_j(s)\,\mathrm{d}s, \tag{6.3}$$

wobei

$$L_\ell(\tau) = \prod_{j=1,\, j\neq\ell}^{m} \frac{\tau - \alpha_j}{\alpha_\ell - \alpha_j}$$

das Lagrangesche Interpolationspolynom zur Stützstelle α_ℓ ist.

Beweis: Wir setzen zur Abkürzung $k_j := f(t_k + \alpha_j\, h, p_m(t_k + \alpha_j\, h))$. Nach der Lagrangeschen Interpolationsformel gilt

$$\underbrace{p_m'(t_k + \tau\, h)}_{\text{Grad} \le m-1} = \sum_{\ell=1}^{m} k_\ell\, L_\ell(\tau) \tag{6.4}$$

$$\Rightarrow \quad \int_0^{\alpha_j} p_m'(t_k + \tau\, h)\,\mathrm{d}\tau = \sum_{\ell=1}^{m} k_\ell \int_0^{\alpha_j} L_\ell(\tau)\,\mathrm{d}\tau$$

$$\Rightarrow \quad p_m(t_k + \alpha_j\, h) = \underbrace{p_m(t_k)}_{=y_k} + h \sum_{\ell=1}^{m} k_\ell\, \underbrace{\beta_{j\ell}}_{\text{wie oben def.}}.$$

Weiter erhält man aus (6.4)

$$\int_0^1 p_m'(t_k + \tau h)\,\mathrm{d}\tau = \sum_{\ell=1}^{m} k_\ell \int_0^1 L_\ell(\tau)\,\mathrm{d}\tau$$

$$\Rightarrow \quad p_m(t_k + h) = \underbrace{p_m(t_k)}_{=y_k} + h \sum_{\ell=1}^{m} k_\ell\, \underbrace{\gamma_\ell}_{\text{wie oben def.}}.$$

[68] siehe Hermann, 2004, Satz 2.4, Hairer, Nørsett, Wanner, 1993, Theorem 7.7

Zusammen ergibt sich ein Schritt eines IRKV:

$$y_{k+1} = p_m(t_k + h) = y_k + h \sum_{j=1}^{m} \gamma_j \, k_j$$

$$k_j = f(t_k + \alpha_j \, h, y_k + h \sum_{\ell=1}^{m} \beta_{j\ell} \, k_\ell).$$

□

Folgendes Resultat ist Grundlage für die Konstruktion von Kollokationsverfahren.

Satz 6.4 (siehe Hairer, Nørsett, Wanner, 1993, Theorem II.7.9)**.** Es sei $M(t) = \prod_{j}^{s}(t - \alpha_j)$ orthogonal zu allen Polynomen vom Höchstgrad $r - 1$, d. h.

$$\int_0^1 M(t) t^{q-1} \mathrm{d}t = 0, \qquad \forall q = 1, \ldots, r.$$

Dann hat das Kollokationsverfahren nach Definition 6.1 Ordnung $p = s + r$.

Die Nullstellen orthogonaler Polynome auf $[0, 1]$ sind also als α_j von besonderer Bedeutung. Dies führt auf die Definition der Gauß-, Radau- und Lobatto-Verfahren im folgenden Abschnitt. Darüber hinaus vereinfachen folgende Definitionen die Analyse der Ordnung von Kollokationsverfahren.

Definition 6.5. Für $p, q, r \in \mathbb{N}$ heißen

$$\frac{1}{k} = \sum_{\ell=1}^{s} \alpha_l^{k-1} \, \gamma_l, \qquad k = 1, \ldots, p. \tag{$B(p)$}$$

$$\frac{1}{k} \alpha_j^k = \sum_{\ell=1}^{s} \alpha_l^{k-1} \beta_{j\ell}, \qquad j = 1, \ldots, s, \quad k = 1, \ldots, q. \tag{$C(q)$}$$

$$\frac{1}{k} \gamma_l (1 - \alpha_l^k) = \sum_{j=1}^{s} \gamma_j \alpha_j^{k-1} \beta_{j\ell}, \qquad l = 1, \ldots, s, \quad k = 1, \ldots, r. \tag{$D(r)$}$$

die **vereinfachenden Annahmen von Butcher**.

Bemerkung 6.6. Betrachtet man das spezielle AWP

$$y'(t) = f(t), \qquad y(t_k) = 0$$

mit seiner exakten Lösung

$$y(t_k + h) = h \int_0^1 f(t_k + s\,h)\,\mathrm{d}s$$

für $f(t) = (t - t_k)^{r-1}$, so erhält man

$$y(t_k + h) = h \int_0^1 (s\,h)^{r-1}\,\mathrm{d}s = \frac{1}{r} h^r.$$

Wegen der Kollokationsbedingungen (6.1) werden solche Polynome bis Grad $r - 1 = m - 1 = s - 1$ exakt integriert. D. h. die zugehörige Quadraturformel hat Genauigkeitsordnung $s - 1$ und $(B(p))$ ist für $p = s$ erfüllt.

Analog heißt $(C(q))$ dass die Steigungen f_{jk} mit einem Quadraturverfahren der Genauigkeitsordnung $q - 1$ bestimmt werden.

Insbesondere heißt $(C(1))$ dass das RKV ist invariant gegenüber „Autonomisierung“ der Dgl (vgl. Satz 3.26).

Wegen

$$\tau^{r-1} = \sum_{j=1}^{m} \alpha_j^{r-1} L_j(\tau) \qquad \forall r = 1, \ldots, m$$

folgt aus (6.3) dass $(C(q))$ für $q = s$ erfüllt ist.

Zusammen folgt

(6.3) ist äquivalent zum linearen Gleichungssystem $(B(p))$, $(C(q))$ mit $p = c = s$.

Satz 6.7 (siehe Hairer, Nørsett, Wanner, 1993, Theorem II.7.4)**.** Erfüllt ein RKV die Bedingungen $(B(p))$, $(C(q))$ und $(D(r))$ für Zahlen $p, q, r \in \mathbb{N}$ mit $q + r \geq p - 1$ und $2q \geq p - 2$, dann besitzt das Verfahren die Konsistenzordnung p.

Bemerkung 6.8 (Bedeutung der Annahmen)**.**

(a) $(B(p)) \Leftrightarrow$ Die dem RKV zugrunde liegende Quadraturformel (vgl. Bemerkung 3.28 und (3.31)) ist exakt für Polynome $f(t) \in \Pi_{p-1}$ (Höchstgrad $p - 1$), besitzt also den Genauigkeitsgrad $p - 1$.

(b) $(C(q)) \Rightarrow$ Die Quadraturformel, die den Stufenwerten $y_{i+1}^{(j)}$ zugrunde liegt (vgl. Bemerkung 3.28), besitzt Ordnung $q - 1$.

(c) $(C(1)) \Leftrightarrow$ Das RKV ist invariant gegenüber „Autonomisierung“ der Dgl (vgl. Satz 3.26).

Es besteht ein enger Zusammenhang zwischen sogenannten Kollokationsverfahren und den Bedingungen $(B(p))$ und $(C(q))$.

In (3.22) hatten wir die Steigungen f_{jk} eines RKV eingeführt mit der Motivation, die Funktionswerte f an irgendwelchen Stützstellen zu approximieren:

$$f_{jk} \approx f(t_k + \alpha_j\, h, y(t_k + \alpha_j\, h)) = y'(t_k + \alpha_j\, h).$$

Lemma 6.9. Ein s-stufiges KV erfüllt $(C(s))$ und $(B(s))$.

Beweis: Die Funktionen $t \mapsto t^{r-1}$, $r = 1, \dots, s$ sind in $\Pi_{r-1} \subset \Pi_{s-1}$, lassen sich also exakt darstellen durch Interpolation der Funktionswerte:

$$t^{r-1} = \sum_{\ell=1}^{s} \alpha_l^{r-1} L_l(t), \quad r = 1, \dots, s.$$

Integration $\int_0^{\alpha_j} \cdots \,\mathrm{d}t$, $j = 1, \dots, s$, ergibt die Bedingungen in $C(s)$. Die Integration $\int_0^1 \cdots \,\mathrm{d}t$ liefert die Bedingungen in $B(s)$. □

Satz 6.10. Ein s-stufiges RKV mit paarweise verschiedenen α_j, $j = 1, \dots, s$ und Konsistenzordnung $p \geq s$, ist genau dann ein KV, wenn $(C(s))$ gilt.

Beweis: Hairer, Nørsett, Wanner, 1993 □

Bemerkung 6.11. Nach Satz 6.10 gilt: Die Gauß-Verfahren, die Radau-I- und die Radau-IIA-Verfahren sowie die Lobatto-IIIA-Verfahren sind Kollokationsverfahren.

Lemma 6.12 (Hairer, Nørsett, Wanner, 1993, p. 204)**.** Erfüllt ein s-stufiges RKV $(B(2s))$ und $(C(s))$, so besitzt es die maximale Konsistenzordnung $p = 2s$.

Beweis: Man kann zeigen, dass $(B(2s))$ und $(C(s))$ auch $D(s)$ implizieren. Nach Satz 6.7 und Lemma 3.38 besitzt das RKV genau die Ordnung $2s$. □

§ 7 Linear implizite Runge-Kutta-Verfahren

Wir haben in § 3.4 gesehen, dass z. B. bei steifen Dgl möglichst A-stabile, also nach Folgerung 3.47 notwendig *implizite* RKV verwendet werden sollten.

Idee zur Aufwandsreduktion bei IRKV:

Nutze ein DIRK-Verfahren: Dann zerfällt die Lösung des nichtlinearen Gleichungssystems $v = F(v)$ in sukzessive zu lösende nichtlineare Gleichungssysteme für die Steigungen $v = (f_{1k}^\top, f_{2k}^\top, \dots, f_{sk}^\top)^\top$ im Intervall $[t_k, t_{k+1}]$, siehe (3.35).

Nur ein Newton-Schritt: . Führe in der Newton-Iteration für jede Steigung k_j jeweils *nur einen Newton-Schritt* aus. Dadurch wird die Häufigkeit der Zerlegung (und der Aufbau) der Jacobimatrix reduziert.

Im Detail: Wir betrachten o. B. d. A. die autonome Dgl $y' = f(y)$ und ein DIRK-Verfahren mit Koeffizienten $B = (\beta_{j\ell})$ (untere Dreiecksmatrix) und γ_j. Das nichtlineare Gleichungssystem $v - F(v) = 0$ kann Stufe für Stufe gelöst werden:

$$f_{jk} - f\left(y_k + h \sum_{\ell=1}^{j} \beta_{j\ell}\, f_{\ell k}\right) = 0 \quad \text{für } j = 1, \dots, s.$$

Ein Newton-Schritt für Stufe $j \in \{1, \dots, s\}$ an irgendeiner Stelle $\tilde{v} = (\tilde{v}_1^\top, \dots, \tilde{v}_s^\top)^\top$ hat die Form (vgl. (3.35))

$$\left[I_{n\times n} - h\,\beta_{jj}\, f_y(y_i + h\sum_{\ell=1}^{j} \beta_{j\ell}\,\tilde{v}_\ell) \right] \Delta\tilde{v}_j = -\tilde{v}_j + f\left(y_i + h\sum_{\ell=1}^{j} \beta_{j\ell}\,\tilde{v}_\ell \right).$$

Führe den Newton-Schritt $v_j^{(1)} = v_j^{(0)} + \Delta v_j^{(0)}$ mit einer möglichst guten Startschätzung $v_j^{(0)}$ aus. Die bereits berechneten $v_\ell^{(1)}$ für $\ell < j$ gehen dabei in die Linearisierungsstelle für Stufe j ein:

$$\left[I_{n\times n} - h\,\beta_{jj}\; \overbrace{f_y\left(y_i + h\underbrace{\sum_{\ell=1}^{j-1} \beta_{j\ell} v_\ell^{(1)}}_{z_j^{(1)}} + h\,\beta_{jj} v_j^{(0)} \right)}^{=:A_j} \right] \left(v_j^{(1)} - v_j^{(0)} \right)$$

$$= -v_j^{(0)} + f(y_i + h\underbrace{\sum_{\ell=1}^{j-1} \beta_{j\ell} v_\ell^{(1)}}_{z_j^{(1)}} + h\,\beta_{jj} v_j^{(0)}). \quad (*)$$

Die Terme $z_j^{(1)}$ lassen sich sich aus den Anstiegen der vorherigen Stufen berechnen, sind also bekannt. Auch für die Definition des Startwertes $v_j^{(0)}$ benutzen wir soviel neue Information wie möglich, d. h. die bereits berechneten Werte $v_\ell^{(1)}$ für $\ell < j$. Wir machen den Ansatz

$$v_j^{(0)} := -\frac{1}{\beta_{jj}} \sum_{\ell=1}^{j-1} \kappa_{j\ell}\, v_\ell^{(1)}$$

mit noch zu wählenden $\kappa_{j\ell} \in \mathbb{R}$. Insbesondere gilt also $v_1^{(0)} = 0$. Einsetzen in (*) liefert:

$$(I_{n\times n} - h\,\underbrace{\beta_{jj}}_{=:\kappa_{jj}} A_j)\, v_j^{(1)} = \underbrace{v_j^{(0)} - v_j^{(0)}}_{=0} - h\,\beta_{jj}\, A_j\, v_j^{(0)} + f\left(y_i + h z_j^{(1)} + h\,\beta_{jj}\, v_j^{(0)} \right)$$

mit der Definition von $z_j^{(1)}$ wird $hz_j^{(1)} + h\beta_{jj} v_j^{(0)}$ zu

$$= h\,A_j \sum_{\ell=1}^{j-1} \kappa_{j\ell}\, v_\ell^{(1)} + f\left(y_i + h\sum_{\ell=1}^{j-1} \underbrace{(\beta_{j\ell} - \kappa_{j\ell})}_{=:\widehat{\beta}_{j\ell}}\, v_\ell^{(1)}) \right)$$

$$= h\,A_j \sum_{\ell=1}^{j-1} \kappa_{j\ell}\, v_\ell^{(1)} + f\left(y_i + h\sum_{\ell=1}^{j-1} \widehat{\beta}_{j\ell}\, v_\ell^{(1)} \right) \quad (7.1)$$

Die $v_j^{(1)}$ werden wie die Steigungen in einem RKV benutzt:

$$y_{i+1} = y_i + h\sum_{j=1}^{s} \gamma_j\, v_j^{(1)}. \quad (7.2)$$

Für ein konkretes Verfahren müssen die γ_j, die $\kappa_{j\ell}$ (untere Dreiecksmatrix, statt $\beta_{j\ell}$) sowie die *strikte* untere Dreiecksmatrix $\widehat{\beta}_{j\ell}$ angegeben werden. Der Aufwand pro Zeitschritt entspricht dem *eines Newton-Schrittes* eines DIRKV:

- s Auswertungen von f,
- Bestimmung der Jacobimatrix $f_y(\cdots)$ an s Stellen,
- Lösung von s linearen Gleichungssystemen der Dimension $n \times n$.

Die Wahl der Startwerte $v_\ell^{(0)}$ ermöglicht auch die Wiederverwendung der Steigungen $v_\ell^{(1)}$ in der Berechnung der Koeffizientenmatrix A_j

$$A_j = f_y\big(y_i + h\sum_{\ell=1}^{j-1} \beta_{j\ell}\, v_\ell^{(1)} + h\,\beta_{jj}\, v_j^{(0)}\big) = f_y\big(y_i + h\sum_{\ell=1}^{j-1} \widehat{\beta}_{j\ell}\, v_\ell^{(1)}\big)\,.$$

Falls etwa (für den aktuellen Zeitschritt) $A_j \equiv A$ für alle Stufen $j = 1, \ldots, s$ gilt und außerdem $\kappa_{jj} \equiv \kappa$ (SDIRKV), dann sind alle Koeffizientenmatrizen in (7.1) identisch, und eine einmal berechnete Zerlegung von A kann für alle s LGS wiederverwendet werden. Der Aufwand der LR-Zerlegungen sinkt damit von $\sim sn^3$ auf $\sim n^3$. Aus dieser Herleitung folgt die

Definition 7.1 (linear implizite RKV)**.** (a) Ein ESV (7.2), dessen Steigungen $v_j^{(1)}$ gemäß (7.1) für $j = 1, \ldots, s$ bestimmt werden (mit beliebigen Matrizen A_j), heißt ein s-stufiges **linear implizites RKV** (LIRKV).

Ein solches LIRKV heißt

(b) **Rosenbrock**-Verfahren für die Wahl

$$A_j \equiv A := f_y(y_i) \qquad \forall j = 1, \ldots, s.$$

$\hat{=}$ kein Update von A

(c) vom **Rosenbrock-Typ** für irgendwelche A_j,

(d) eine **W-Methode**, falls $A_j \equiv A$ ist für eine beliebige (vom Zeitschritt i unabhängige) Matrix A und $\kappa_{jj} = \kappa$ für $j = 1, \ldots, s$ gilt

(e) Es heißt ein **Rosenbrock-Wanner-Verfahren, (ROW)-Verfahren** für die Wahl $A_j \equiv f_y(y_i)$ und $\kappa_{jj} = \kappa$ für alle $j = 1, \ldots, s$.

verschiedener linear impliziter Runge-Kutta Verfahren.

Bemerkung 7.2 (Alternative Herleitung von Rosenbrock-Verfahren)**.** Rosenbrock-Verfahren lassen sich auch wie folgt herleiten: Schreibe

$$y'(t) = \underbrace{f_y(y_i)\, y(t)}_{\text{linear}} + \underbrace{f(y(t)) - f_y(y_i)\, y(t)}_{\text{nichtlinear}}\,.$$

Verwende ein RKV, das den linearen Teil wie ein DIRKV mit $B = (\beta_{j\ell})$ und den nichtlinearen Teil wie ein ERKV mit $\widehat{B} = (\widehat{\beta}_{j\ell})$ behandelt. Aus diesem Grund spricht man auch von **semi-impliziten Verfahren**.

Bemerkung 7.3. (a) Man hat nur lineare Gleichungssysteme zu lösen. Bei den Wanner- und ROW-Verfahren ist die Koeffizientenmatrix vor $v_j^{(1)}$ für alle $j = 1, \dots, s$ gleich (bei den Wanner-Verfahren sogar für alle Zeitschritte). Eine Zerlegung kann wiederverwendet werden.

(b) Ordnungsbedingungen sind auch Bedingungen an die Wahl von A_j.

(c) In der Praxis wertet man die Jacobi-Matrix $f_y(y_i)$ nur gelegentlich neu aus und verwendet sie über mehrere Integrationsschritte. Das entspricht $A_j \equiv f_y(y_i) + \mathcal{O}(h)$.

(d) Es existieren Erweiterungen für nichtautonome Dgl.

(e) Für rechte Seiten $f(y) = A\,y$ sind Rosenbrock-Verfahren wieder DIRK-Verfahren, und ROW-Verfahren sind SDIRK-Verfahren.

(f) Linear implizite RKV liegen vom Aufwand pro Schritt näher an expliziten RKV, erben aber die guten Stabilitätseigenschaften von impliziten RKV, siehe § 3.4.

Beispiel 7.4 (Linear implizite Runge-Kutta-Verfahren)**.**

(a) Wir betrachten das einstufige LIRKV (eine W-Methode)

$$\begin{aligned}(I - h\,\kappa A)\,v_1^{(1)} &= f(y_i)\\ y_{i+1} &= y_i + h\,v_1^{(1)},\end{aligned}$$

mit festem A. Für den lokalen Diskretisierungsfehler gilt

$$\begin{aligned}h\,d(y(\cdot), t+h, h) &= y(t+h) - y(t) - h\,k_1^{(1)}\\ &= y(t) + y'(t)\,h + \frac{1}{2}y''(t)\,h^2 + \mathcal{O}(h^3)\\ &\quad - y(t) - h\,(I - h\,\kappa A)^{-1} f(y(t))\end{aligned}$$

Dabei ist $(I - h\,\kappa A)^{-1} = \sum_{n=0}^{\infty}(h\,\kappa A)^n$ für $h \le \overline{h}$ (Neumannsche Reihe), und deshalb gilt weiter:

$$\begin{aligned}&= f(\cdot)\,h + \frac{1}{2}f_y(\cdot)f(\cdot)\,h^2 + \mathcal{O}(h^3)\\ &\quad - h\,\big[I + h\,\kappa A + \mathcal{O}(h^2)\big] f(\cdot)\\ &= 0\cdot h + \frac{1}{2}\big(f_y(\cdot) - A\big) f(\cdot)\,h^2 + \big(\frac{1}{2} - \kappa\big) A\,f(\cdot)\,h^2 + \mathcal{O}(h^3).\end{aligned}$$

Beachte: Unabhängig von der Wahl von A erhalten wir also die Ordnung $p = 1$!

(b) (Siehe Sheet 13, Exercise 34) Mit $\kappa = 1$ und $A = f_y(y_i)$ erhalten wir aus (a) das **linear implizite Euler-Verfahren** (eine ROW-Methode)

$$\begin{aligned}(I - h f_y(y_i))\,v_1^{(1)} &= f(y_i)\\ y_{i+1} &= y_i + h\,v_1^{(1)}.\end{aligned}$$

Die Ordnung ist wieder $p = 1$.

(c) Mit $\kappa = 1/2$ und $A = f_y(y_i)$ erhalten wir aus (a) die **linear implizite Mittelpunktsregel** (eine ROW-Methode)

$$\left(I - \frac{h}{2} f_y(y_i)\right) v_1^{(1)} = f(y_i)$$
$$y_{i+1} = y_i + h\, v_1^{(1)}.$$

Dies ist ein Verfahren der Ordnung $p = 2$.

§ 8 Differentiell-algebraische Systeme

Eine gute Einführung in das Thema sind Kunkel, Mehrmann, 1994 und Hairer, Wanner, 1996, S. VII. Die Notation orientiert sich fast vollständig an Kunkel, Mehrmann, 1994.

§ 8.1 Einführung

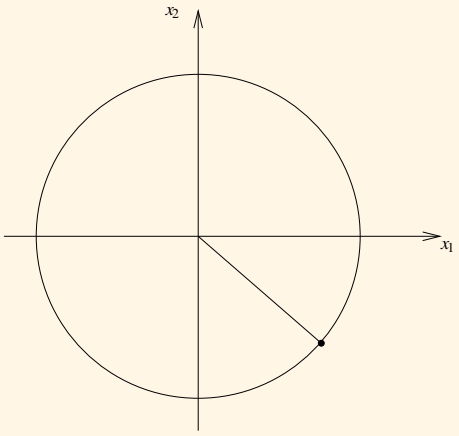
In vielen Anwendungen ergeben sich aus der Modellierung heraus keine Dgl der Form $y'(t) = f(t, y(t))$, sondern eine Gleichung des allgemeineren Typs

$$F(t, y(t), y'(t)) = 0 \quad \text{in } [0, T].$$

Beispiel 8.1 (Ebenes Pendel)**.** Wir betrachten ein Pendel mit einem masselosen Stab der festen Länge ℓ, an dessen Ende sich eine Punktmasse m befindet. Der Ortspunkt der Masse genügt der Zwangsbedingung

$$x_1^2 + x_2^2 = \ell^2,$$

siehe Beispiel 8.1.



Zwangsbedingung ebenes Pendel

Man kann die Bewegungsgleichungen mit dem Lagrange-Formalismus der klassischen Mechanik herleiten. Dabei wird das Prinzip der kleinsten Wirkung angewendet, was (vereinfacht) besagt, das die Bewegung entlang derjenigen Trajektorie erfolgt, entlang der die Bewegungsenergie minimal ist, unter allen möglichen Trajektorien. Obige Zwangsbedingung stellt eine Gleichungsbeschränkung in diesem Optimierungsproblem dar.

Minimiert wird $L := T - V$, die Differenz aus

$$\text{kinetischer Energie} \quad T := \frac{1}{2} m\big((x_1')^2 + (x_2')^2\big)$$
$$\text{und potentieller Energie} \quad V := m\,g\,x_2.$$

Um das restringierte Optimierungsproblem zu behandeln bilden wir die Lagrangefunktion[69]

$$\mathcal{L}(q, q', t) = \frac{1}{2} m\big((x_1')^2 + (x_2')^2\big) - m\,g\,x_2 - \lambda\,(x_1^2 + x_2^2 - \ell^2).$$

Dabei ist $q = (x_1, x_2, \lambda)$ der Vektor der **erweiterten Zustandsvariablen** und λ ein **Lagrange-Multiplikator**. Die notwendigen Optimalitätsbedingungen sind die Euler-Lagrange-Gleichungen,

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial q'}\right) - \frac{\partial \mathcal{L}}{\partial q} = 0. \tag{8.1}$$

Durch Einsetzen in (8.1) ergeben sich schließlich die Bewegungsgleichungen für q als Funktion der Zeit:

$$\begin{aligned} m\,x_1'' + 2\,x_1\,\lambda &= 0, \\ m\,x_2'' + 2\,x_2\,\lambda + m\,g &= 0, \\ x_1^2 + x_2^2 - \ell^2 &= 0. \end{aligned} \tag{8.2}$$

Es handelt sich um eine Dgl, gekoppelt mit einer algebraischen Gleichung (Zwangsbedingung). Daher bezeichnet man (8.2) als **differentiell-algebraische Gleichung (DAE)** oder **Algebro-Differentialgleichung** oder auch als **Deskriptor-System**.

Beachte: Der Lagrange-Multiplikator λ kann als Zwangskraft(dichte)[70] gedeutet werden, die die Pendelmasse auf der Kreisbahn hält. Der Lagrange-Formalismus ist invariant unter Koordinaten-Transformationen und kann selbst in beschleunigten Bezugssystemen verwendet werden. Er erlaubt eine einfache, formale Herleitung der Bewegungsgleichungen, und ist daher für die Erstellung von Software zur automatischen Modellierung und Simulation mechanischer Systeme besonders geeignet.

Als DAE 1. Ordnung geschrieben lautet (8.2):

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 \\ 0 & 0 & 0 & m & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{E} \begin{pmatrix} x_1' \\ x_2' \\ x_3' \\ x_4' \\ \lambda' \end{pmatrix} = \begin{pmatrix} x_3 \\ x_4 \\ -2\,x_1\,\lambda \\ -2\,x_2\,\lambda - m\,g \\ x_1^2 + x_2^2 - \ell^2 \end{pmatrix}$$

Charakteristisch für DAEs ist, dass E eine singuläre Matrix ist.

DAEs treten u. a. auf bei der Modellierung

- mechanischer Mehrkörpersysteme (Roboter, Fahrzeuge) sowie
- elektrischer Schaltkreise.

[69] Einheit: Nm
[70] Einheit: N/m

Die algebraischen Zwangsbedingungen resultieren dabei aus Bedingungen an Lagern oder Gelenken/Verbindungen zwischen Bauteilen bzw. aus den Kirchhoffschen Gesetzen.[71]

Beispiel 8.1 kann durch Einführung generalisierter Koordinaten wieder auf eine Dgl (2. Ordnung) ohne algebraische Bedingungen gebracht werden. In diesem einfachen Beispiel ist der Winkel die einzige (verallgemeinerte) Koordinate und die Polartransformation

$$\begin{pmatrix} x \\ y \end{pmatrix} = l \begin{pmatrix} \sin(\varphi) \\ \cos(\varphi) \end{pmatrix}$$

mit festem Radius $r = l$ die zugehörige Transformation zwischen den Koordinaten. Bei größeren Aufgaben ist das jedoch unpraktisch. Deshalb ist man daran interessiert, DAE-Systeme direkt numerisch zu lösen.

§ 8.2 Eigenschaften linearer DAE-Systeme

Wir wollen auf einige Besonderheiten bei DAE-Systemen eingehen. Wir betrachten dazu den Spezialfall eines *linearen* Systems mit konstanten Koeffizienten[72]

$$E y' = A\, y + f(t) \tag{DAE}$$

mit $E, A \in \mathbb{C}^{n\times n}$, $f \in C([0,T];\mathbb{C}^n)$ und einer Anfangsbedingung

$$y(0) = y_a,$$

mit $y_a \in \mathbb{C}^n$.

Beachte: Falls E invertierbar ist, so kann man (DAE) umformen in

$$y' = E^{-1}(A\, y + f(t))$$

und wie eine Dgl behandeln. Interessant sind daher besonders die Fälle, in denen E singulär ist.

Skaliert man die (DAE) mit einer regulären Matrix $P \in \mathbb{C}^{n\times n}$ und transformiert $y = Q\,\bar{y}$ mit $Q \in \mathbb{C}^{n\times n}$ regulär, so erhält man die lineare DAE

$$\bar{E}\,\bar{y}' = \bar{A}\bar{y} + \bar{f}(t), \qquad \bar{E} = PEQ,$$
$$\bar{f}(t) = P\, f(t), \qquad \bar{A} = PAQ.$$

Da $\bar{y} = Q^{-1}y$ die Lösungen umkehrbar eindeutig aufeinander abbildet, sind die Systeme äquivalent.

Definition 8.2 (DAE-Äquivalenz)**.** Zwei Matrixpaare (E_i, A_i), $i = 1, 2$, heißen (stark) **äquivalent**, wenn es reguläre $P, Q \in \mathbb{C}^{n\times n}$ gibt, sodass gilt:

$$E_2 = PE_1Q, \qquad A_2 = PA_1Q.$$

Man schreibt $(E_1, A_1) \sim (E_2, A_2)$.

Die durch Definition 8.2 beschriebene Relation ist eine Äquivalenzrelation (reflexiv, symmetrisch, transitiv: einfach zu prüfen). Später stellen wir die Frage nach einer Normalform.

[71] Die Summe aller Ströme in jedem Knoten ist null. Die Summe aller Teilspannungen in einer Masche ist null.

[72] Die algebraische Behandlung solcher Systeme geht wohl bereits auf Weierstraß (1815-1897) und Kronecker (1823-1891) zurück.

Definition 8.3 (Charakt. Polynom, reguläre und singuläre Matrixpaare)**.** (a) Das Polynom

$$p(\lambda) = \det(\lambda E - A)$$

heißt **charakteristisches Polynom** des Matrixpaares (E, A).

(b) Ist $p \equiv 0$ das Nullpolynom, so heißt das Matrixpaar (E, A) **singulär**, sonst **regulär**.

(c) Man spricht bei $\lambda E - A$ auch von einem (regulären oder singulären) **Matrixbüschel** (*matrix pencil*).

Bemerkung 8.4. Dies ist eine Verallgemeinerung des charakteristischen Polynoms einer Matrix: Falls $E = I$ ist, so ist $p(\lambda)$ bis auf den Faktor $(-1)^n$ das charakteristische Polynom von A.

Beispiel 8.5. Für das Matrixpaar

$$E_1 := \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \qquad A_1 := \begin{pmatrix} 1 & 0 \\ -1 & 0 \end{pmatrix}$$

ergibt sich

$$p(\lambda) = \det\left(\begin{bmatrix} \lambda - 1 & 0 \\ 1 & 0 \end{bmatrix}\right) = 0.$$

Hingegen für

$$E_2 := \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \qquad A_2 := \begin{pmatrix} 1 & 0 \\ -1 & -1 \end{pmatrix}$$

ergibt sich

$$p(\lambda) = \det\left(\begin{bmatrix} \lambda - 1 & 0 \\ 1 & 1 \end{bmatrix}\right) = \lambda - 1.$$

Lemma 8.6 (Reguläre Matrixpaare bilden eine Äquivalenzklasse)**.** Jedes zu einem regulären Matrixpaar äquivalente Matrixpaar ist selbst regulär.

Beweis: Es sei $E_2 = PE_1Q$, $A_2 = PA_1Q$ mit $P, Q \in \mathbb{C}^{n\times n}$ regulär. Dann gilt

$$\begin{aligned} p_2(\lambda) &= \det(\lambda E_2 - A_2) = \det(P\,(\lambda E_1 - A_1)\,Q) \\ &= \det(P)\det(\lambda E_1 - A_1)\det(Q) \\ &= c\,p_1(\lambda) \end{aligned}$$

mit $c \neq 0$. □

Lemma 8.7 (Singuläre Matrixpaare erzeugen DAE mit nichttrivialer homogener Lösung)**.** Ist das Matrixpaar (E, A) singulär, so besitzt das *homogene* DAE-AWP

$$Ey' = A\,y, \qquad y(0) = 0$$

eine nichttriviale Lösung.

Beweis: Es sei (E, A) singulär, also $p(\lambda) = \det(\lambda E - A) \equiv 0$, d. h., $(\lambda E - A)$ ist singulär für alle $\lambda \in \mathbb{C}$. Es seien $\lambda_i \in \mathbb{C}$, $i = 1, \ldots, n+1$ beliebige, aber paarweise

verschiedene Zahlen. Zu jedem λ_i existiert ein $v_i \in \mathbb{C}^n \setminus \{0\}$ mit

$$(\lambda_i E - A)\, v_i = 0\,.$$

Man nennt v_i einen **verallgemeinerten Eigenvektor** und λ_i einen **verallgemeinerten Eigenwert** zum **verallgemeinerten Eigenwertproblem** (8.2). Da die $n+1$ Vektoren v_i im $\mathbb{C}^n$ notwendig linear abhängig sind, existieren Koeffizienten $\alpha_i \in \mathbb{C}$, $i = 1, \ldots, n+1$ (nicht alle gleich null), sodass

$$\sum_{i=1}^{n+1} \alpha_i\, v_i = 0$$

gilt. Die Funktion

$$y(t) = \sum_{i=1}^{n+1} \alpha_i\, v_i\, \mathrm{e}^{\lambda_i t}$$

erfüllt $y(0) = 0$ und die homogene (DAE), da

$$E y'(t) = \sum_{i=1}^{n+1} \alpha_i\, \lambda_i\, E\, v_i\, \mathrm{e}^{\lambda_i t} \overset{(8.2)}{=} \sum_{i=1}^{n+1} \alpha_i A\, v_i\, \mathrm{e}^{\lambda_i t} = A\, y(t).$$

Somit ist $y(t)$ offenbar eine nichttriviale Lösung des homogenen AWP. □

Beispiel 8.8. Für das singuläre Matrixpaar aus Beispiel 8.5,

$$E := \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \qquad A := \begin{pmatrix} 1 & 0 \\ -1 & 0 \end{pmatrix}$$

wählt man z. B. $\lambda_1 = 2$, $\lambda_2 = 3$, und erhält verallgemeinerte Eigenvektoren aus

$$0 = (\lambda_1 E - A)v_1 = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix} v_1 \qquad \Rightarrow v_1 = t \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \qquad t \in \mathbb{C},$$

$$0 = (\lambda_2 E - A)v_2 = \begin{pmatrix} 2 & 0 \\ 1 & 0 \end{pmatrix} v_2 \qquad \Rightarrow v_2 = q \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \qquad q \in \mathbb{C}.$$

(Da $\{v_1, v_2\}$ bereits linear abhängig sind, brauchen wir kein λ_3.) Wir können also z. B. $t = 1$ und $q = -1$ wählen, wodurch sich ergibt

$$v_1 + v_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Daher ist

$$y(t) = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \mathrm{e}^{2t} + \begin{pmatrix} 0 \\ -1 \end{pmatrix} \mathrm{e}^{3t}$$

eine nichttriviale Lösung des homogenen DAE-AWP

$$E\, y' = A\, y, \qquad y(0) = 0.$$

Bemerkung 8.9 (DAE mit singulären Matrixpaaren)**.** Nach Lemma 8.7 ist ein DAE-AWP mit singulärem Matrixpaar nie eindeutig lösbar, da man stets ein beliebiges Vielfaches einer Lösung des homogenen AWP addieren kann. Daher schließen wir diesen Fall im Weiteren aus.

Frage: Ist das DAE-AWP im regulären Fall eindeutig lösbar?

Für reguläre Matrixpaare kann man die Eigenschaften der (DAE) mit Hilfe der Weierstraßschen Normalform analysieren.

Satz 8.10 (Weierstraßsche Normalform)**.** Es sei das Matrixpaar (E, A) regulär. Dann gilt:

$$(E, A) \sim \left(\begin{bmatrix} I & 0 \\ 0 & N \end{bmatrix}, \begin{bmatrix} J & 0 \\ 0 & I \end{bmatrix} \right), \tag{8.3}$$

wobei J in Jordanscher Normalform ist, also $J = \operatorname{diag}(J_1, \ldots, J_m)$ mit **Jordanblöcken**

$$J_i = \begin{bmatrix} \lambda_i & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_i \end{bmatrix}, \qquad i = 1, \ldots, m.$$

(Die λ_i sind also Eigenwerte von J.) N ist ebenfalls in Jordanscher Normalform, aber mit Nullen auf der Diagonale (nilpotent). Dabei ist es möglich, dass einer der Blöcke J oder N nicht vorkommt (die Größe null hat). Die rechte Seite von (8.3) heißt (eine) **Weierstraßsche Normalform** des regulären Matrixpaares (E, A).

Beweis: Es sei (E, A) regulär, d. h., es existiert $\lambda_0 \in \mathbb{C}$, sodass $\det(\lambda_0 E - A) \neq 0$.

Schritt (i): Da $\lambda_0 E - A$ regulär ist, gilt

$$\begin{aligned} (E, A) &= (E, A - \lambda_0 E + \lambda_0 E) \\ &\sim ((A - \lambda_0 E)^{-1} E, I + \lambda_0 (A - \lambda_0 E)^{-1} E) \end{aligned} \tag{8.4}$$

mit der Transformation $P_1 = (A - \lambda_0\, E)^{-1}$ und $Q_1 = I$.

Schritt (ii): Transformiere nun $(A - \lambda_0 E)^{-1} E$ auf Jordansche Normalform:

$$P_2\, (A - \lambda_0 E)^{-1} E\, P_2^{-1} = \operatorname{diag}(\bar{J}, \bar{N}),$$

wobei $\bar{J}$ die regulären Jordanblöcke enthält (also frei von Null-Eigenwerten ist) und $\bar{N}$ die restlichen Jordanblöcke (ausschließlich Null-Eigenwerte). Damit ist $\bar{N}$ eine strikte obere Dreiecksmatrix (insbesondere nilpotent). Durch Anwendung derselben Transformation $P_2 \cdot (\ldots) \cdot P_2^{-1}$ auf die rechte Matrix in (8.4) erhalten wir

$$\begin{aligned} &P_2 \left(I + \lambda_0 (A - \lambda_0 E)^{-1} E \right) P_2^{-1} \\ &= \underbrace{P_2\, P_2^{-1}}_{I} + \underbrace{\lambda_0 P_2 (A - \lambda_0 E)^{-1} E\, P_2^{-1}}_{\lambda_0 \operatorname{diag}(\bar{J}, \bar{N})}, \end{aligned}$$

also

$$(E, A) \sim \left(\begin{bmatrix} \bar{J} & 0 \\ 0 & \bar{N} \end{bmatrix}, \begin{bmatrix} I + \lambda_0 \bar{J} & 0 \\ 0 & I + \lambda_0 \bar{N} \end{bmatrix} \right).$$

Dabei ist $I + \lambda_0 \bar{N}$ eine reguläre obere Dreiecksmatrix.

Schritt (iii): Eine weitere Äquivalenztransformation mit $P_3 = \operatorname{diag}(\bar{J}^{-1}, (I + \lambda_0 \bar{N})^{-1})$ und $Q_3 = I$ ergibt

$$(E, A) \sim \left(\begin{bmatrix} I & 0 \\ 0 & (I + \lambda_0 \bar{N})^{-1} \bar{N} \end{bmatrix}, \begin{bmatrix} \bar{J}^{-1} + \lambda_0 I & 0 \\ 0 & I \end{bmatrix} \right).$$

Da $\bar{N}$ strikte obere Dreiecksmatrix und $(I + \lambda_0 \bar{N})^{-1}$ obere Dreiecksmatrix ist, ergibt $(I + \lambda_0 \bar{N})^{-1}\bar{N}$ wieder eine strikte obere Dreiecksmatrix, hat also nur Null-Eigenwerte.

Schritt (iv): Schließlich bringt man noch die entkoppelten Blöcke $(I+\lambda_0 \bar{N})^{-1}\bar{N}$ und $\bar{J}^{-1}+\lambda_0 I$ mit einer Transformation $P_4 = \operatorname{diag}(P_{41}, P_{42})$ und $Q = P_4^{-1}$ auf Jordansche Normalform. Diese liefert dann die gewünschte Blockstruktur (8.3). Insbesondere besteht die Jordansche Normalform N zu $(I + \lambda_0 \bar{N})^{-1}\bar{N}$ aus lauter Blöcken mit Null-Eigenwerten.

□

Bemerkung 8.11 (zur Weierstraßschen Normalform der DAE)**.** Bei regulärem Matrixpaar (E, A) zerfällt die (DAE) bei entsprechender Äquivalenztransformation damit in zwei entkoppelte Probleme

$$y_1' = Jy_1 + f_1(t), \tag{8.5a}$$

$$Ny_2' = \quad y_2 + f_2(t). \tag{8.5b}$$

mit $y_1(t) \in \mathbb{C}^{n-\nu}$ und $y_2(t) \in \mathbb{C}^{\nu}$. Man nennt dann (8.5) die **Weierstraßsche Normalform** von (DAE). Da (8.5a) eine lineare gewöhnliche Dgl mit konstanten Koeffizienten ist, existiert stets eine eindeutige Lösung zu diesem Teil des zugehörigen DAE-AWP. Man nennt dies den differentiellen Anteil der (DAE) Im Folgenden untersuchen wir Eigenschaften des **algebraischen Anteils** (8.5b).

Lemma 8.12 (Eigenschaften von (8.5b))**.** Die DAE (8.5b) besitzt für $f_2 \in C^{\nu}([0,T];\mathbb{C}^{\nu})$ die eindeutige Lösung

$$y_2(t) = -\sum_{\ell=0}^{\nu-1} N^{\ell} f_2^{(\ell)}(t), \tag{8.6}$$

wobei $\nu \geq 1$ der **Nilpotenzindex** von N ist (d. h., $N^{\nu} = 0$, aber $N^{\nu-1} \neq 0$). Die Lösung y_2 gehört zu $C^1([0,T];\mathbb{C}^{\nu})$ bzw. allgemein zu $C^{p+1}([0,T];\mathbb{C}^{\nu})$, falls sogar $f_2 \in C^{\nu+p}([0,T];\mathbb{C}^{\nu})$ ist für ein $p \in \mathbb{N}_0$.

Beweis: Falls N nur aus einem Jordanblock besteht, so hat (8.5b) die Form

$$\begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & 0 \end{bmatrix} y_2' = y_2 + f_2(t).$$

Der Nilpotenzindex ist gerade die Größe der Matrix. Daraus ergibt sich (von letzter Zeile bis erster)

$$\begin{aligned}
0 = y_{2,\nu}(t) + f_{2,\nu}(t) \quad &\Rightarrow \quad y_{2,\nu}(t) = -f_{2,\nu}(t) = -[N^0 f(t)]_{2,\nu},\\
y'_{2,\nu}(t) = y_{2,\nu-1}(t) + f_{2,\nu-1}(t) \quad &\Rightarrow \quad y_{2,\nu-1}(t) = -f_{2,\nu-1}(t) - f'_{2,\nu}(t)\\
&\qquad\qquad = -[N^0 f(t) + N^1 f'(t)]_{\nu-1},\\
\vdots \quad \vdots \qquad &\qquad\qquad \vdots \quad \vdots\\
y'_{2,j+1}(t) = y_{2,j}(t) + f_{2,j}(t) \quad &\Rightarrow \quad y_{2,j}(t) = -\sum_{\ell=0}^{\nu-j} f^{(\ell)}_{2,j+\ell}(t) = -\sum_{\ell=0}^{\nu-j} [N^\ell f_2^{(\ell)}(t)]_j\\
&\qquad\qquad = -\sum_{\ell=0}^{\nu-1} [N^\ell f_2^{(\ell)}(t)]_j
\end{aligned}$$

für alle $j = 1, \dots, \nu - 1$, denn die j-te Zeile von N^ℓ ist Null für $\ell \geq (\nu - j) + 1$. Also gilt (8.6) entsprechend für diesen Block.

Da der Nilpotenzindex der Gesamtmatrix das Maximum der Nilpotenzindizes der einzelnen Blöcke ist, und eine Erhöhung der Obergrenze der Summation nicht schädlich ist, liefert das Zusammensetzen der Blöcke die Behauptung. □

Bemerkung 8.13 (zur Lösbarkeit von (8.5))**.**

(a) Die Lösung (8.6) von (8.5b) ist bereits ohne Vorgabe eines Anfangswertes eindeutig. Eine Lösung des DAE-AWP existiert nur dann, wenn der AW mit (8.6) konsistent ist.

(b) Die Zahl ν ist ein wichtiges Merkmal der (DAE) und bestimmt u. a. die Glattheitsanforderung an die rechte Seite f_2. ($\nu - 1$ mal für (8.6) plus ein weiteres Mal, sodass y' stetig ist.)

(c) Die numerische Bestimmung und Nutzung von ν ist sehr schwierig, da die Jordanblockstruktur einer Matrix sehr empfindlich ist gegenüber Rundungsfehlern. Zum Beispiel hat

$$\begin{pmatrix} 1 & 1 \\ \varepsilon & 1 \end{pmatrix}$$

für $\varepsilon = 0$ offenbar die Jordansche Normalform

$$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix},$$

für $\varepsilon > 0$ jedoch

$$\begin{pmatrix} 1 + \sqrt{\varepsilon} & 0 \\ 0 & 1 - \sqrt{\varepsilon} \end{pmatrix}.$$

Die Berechnung und Verwendung kann daher praktisch nur analytisch erfolgen.

Definition 8.14 (Index eines regulären Matrixpaares)**.** Das Matrixpaar (E, A) sei regulär mit Normalform (8.3). Dann heißt der Nilpotenzindex $\nu \geq 1$ von N der

Index des Matrixpaares (E, A), kurz: $\nu = \operatorname{ind}(E, A)$. Man setzt $\operatorname{ind}(E, A) = 0$, falls kein nilpotenter Block N vorkommt (ODE-Fall).

Wir stellen nun sicher, dass ν eindeutig bestimmt ist.

Lemma 8.15 (Invarianz der Normalform einer regulären DAE)**.** 2 Es seien zum regulären Matrixpaar (E, A) zwei Normalformen gemäß Satz 8.10 gegeben:

$$(E, A) \sim \left(\begin{bmatrix} I & 0 \\ 0 & N_i \end{bmatrix}, \begin{bmatrix} J_i & 0 \\ 0 & I \end{bmatrix} \right), \quad i = 1, 2,$$

wobei d_i die Größe von J_i sei. Dann gilt $d_1 = d_2$, und J_1, J_2 sowie N_1, N_2 besitzen jeweils identische Jordanblöcke (möglicherweise in anderer Anordnung). Insbesondere haben N_1 und N_2 denselben Nilpotenzindex.

Beweis: Betrachte die charakteristischen Polynome der beiden Normalformen

$$\begin{aligned} p_i(\lambda) &= \det \begin{bmatrix} \lambda I - J_i & 0 \\ 0 & \lambda N_i - I \end{bmatrix} \\ &= \det(\lambda I - J_i) \det(\lambda N_i - I) \\ &= \det(\lambda I - J_i) (-1)^{n-d_i}, \quad \text{denn } \lambda N_i - I \text{ ist obere Dreiecksmatrix.} \end{aligned}$$

Damit ist p_i vom Grad d_i, und da sich die Polynome nach dem Beweis von Lemma 8.6 nur um einen konstanten Faktor unterscheiden können, folgt $d_1 = d_2$. Da die Normalformen auch untereinander äquivalent sind (Transitivität), gilt[73]

$$\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & N_1 \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & N_2 \end{bmatrix} \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix}$$

und

$$\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} J_1 & 0 \\ 0 & I \end{bmatrix} = \begin{bmatrix} J_2 & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix}$$

mit regulären Matrizen $P = [P_{ij}]_{i,j=1,2}$ und $Q = [Q_{ij}]_{i,j=1,2}$. Ausmultiplizieren ergibt

$$\begin{aligned} P_{11} &= Q_{11}, & P_{12}N_1 &= Q_{12}, \\ P_{21} &= N_2 Q_{21}, & P_{22}N_1 &= N_2 Q_{22}, \\ P_{11}J_1 &= J_2 Q_{11}, & P_{12} &= J_2 Q_{12}, \\ P_{21}J_1 &= Q_{21}, & P_{22} &= Q_{22}. \end{aligned}$$

Damit folgt

$$P_{12} = J_2 P_{12} N_1, \qquad P_{21} = N_2 P_{21} J_1.$$

Wiederholtes Einsetzen liefert

$$P_{12} = J_2 P_{12} N_1 = J_2^2 P_{12} N_1^2 = \ldots = J_2^{n-d_1} P_{12} N_1^{n-d_1} = 0$$

(und analog $P_{21} = 0$). Also hat P die Gestalt

$$P = \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix} = Q.$$

[73] Wir vereinfachen hier die Notation aus Definition 8.2 und schreiben rechts Q statt Q^{-1}.

Da P invertierbar ist, müssen $P_{11} = Q_{11}$ und $P_{22} = Q_{22}$ ebenfalls invertierbar sein. Also gilt

$$P_{11} J_1 P_{11}^{-1} = J_2, \qquad P_{22} N_1 P_{22}^{-1} = N_2,$$

d. h., die Matrizen J_1 und J_2 sowie N_1 und N_2 sind ähnlich. Damit stimmen sie jeweils in Eigenwerten und Dimension der Jordanblöcke überein (Theorie der Jordanschen Normalform). □

Satz 8.16 (Zusammenfassung der Lösungstheorie von DAE-AWP)**.**
Es sei (E, A) ein reguläres Matrixpaar und durch reguläre Matrizen P und Q eine Transformation von (DAE) und Anfangsbedingung $y(0) = y_a$ auf Normalform gemäß

$$PEQ = \begin{bmatrix} I & 0 \\ 0 & N \end{bmatrix}, \qquad PAQ = \begin{bmatrix} J & 0 \\ 0 & I \end{bmatrix}, \qquad Pf = \begin{bmatrix} \bar{f}_1 \\ \bar{f}_2 \end{bmatrix} \tag{8.7a}$$

und

$$Q^{-1}y = \begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \end{bmatrix}, \qquad Q^{-1}y_a = \begin{bmatrix} \bar{y}_{a1} \\ \bar{y}_{a2} \end{bmatrix} \tag{8.7b}$$

gegeben. Ferner sei $\nu = \mathrm{ind}(E, A)$ und $f \in C^\nu([0, T]; \mathbb{C}^n)$. Dann gilt:

(a) Die (DAE) besitzt mindestens eine Lösung (ohne Vorgabe einer Anfangsbedingung).

(b) Die Anfangsbedingung $y(0) = y_a$ ist genau dann **konsistent** mit der (DAE), wenn gilt

$$\bar{y}_{a2} = -\sum_{\ell=0}^{\nu-1} N^\ell \bar{f}_2^{(\ell)}(0).$$

(c) Jedes AWP für (DAE) mit konsistenter Anfangsbedingung ist eindeutig lösbar.

(d) Sind die Anfangsbedingungen nicht konsistent mit der DAE, so existiert **keine** Lösung für das DAE-AWP.

Bemerkung 8.17 (zu DAE allgemein)**.**

(a) Die behandelten linearen DAE mit konstanten Koeffizienten liefern nur einen ersten Einblick in die Effekte, welche bei allgemeinen DAE auftreten können. Bereits die Verallgemeinerung auf lineare DAE mit variablen Koeffizienten ergibt zusätzliche Schwierigkeiten und erfordert eine wesentlich aufwändigere Theorie, siehe Kunkel, Mehrmann, 1994.

(b) In der Literatur wird oft der **differentielle Index einer DAE** definiert als die minimale Anzahl an Differentiationen, bis man aus der allgemeinen impliziten Gleichung

$$0 = F(t, y(t), y'(t))$$

und ihren Ableitungen

$$\begin{aligned} 0 &= F_t + F_y y(t) + F_{y'} y'(t), \\ 0 &= F_{t,t} + 2F_{t,y} y(t) + 2F_{t,y'} y'(t) + F_{y,y}[y(t), y(t)] \\ &\quad + 2F_{y,y'}[y(t), y'(t)] + F_{y',y'}[y'(t), y'(t)], \\ &\vdots \end{aligned}$$

eine explizite Dgl $y' = f(t, y)$ extrahieren kann.

Ein Beispiel aus Hairer, Wanner, 1996, Def. VII.1.2 mit drei Komponenten dient zum Verständnis :

$$\begin{aligned} y_2' + y_1 &= f_1(t) \\ y_3' + y_2 &= f_2(t) \\ y_3 &= f_3(t) \end{aligned}$$

Die Komponente y_1 steht nur in der ersten Gleichung. Wird diese differenziert

$$y_2'' + y_1' = f_1'(t)$$

wird eine Gleichung für y_2'' benötigt. Die Komponente y_2 gibt es nur nochmals in der zweiten Gleichung. Somit muss ein zweites mal differenziert werden

$$y_3''' + y_2'' = f_2''(t)\,.$$

Mit dem gleichen Argument benötigt man die dritte Ableitung für die Gleichung

$$y_3''' = f_3'''(t)\,.$$

Diese DAE hat also Index drei. Nach dem Einsetzen der höheren Ableitungen erhält man die zugrundeliegende Differentialgleichung

$$\begin{aligned} y_1' &= f_1'(t) - f_2''(t) + f_3'''(t) \\ y_2' &= f_2'(t) - f_3''(t) \\ y_3' &= f_3'(t)\,. \end{aligned}$$

Für lineare DAE mit konstanten Koeffizienten ist dies gerade der Index aus Definition 8.14.

Es gibt noch weitere, teils ähnliche Index-Konzepte für DAE, welche alle zum Ziel haben, den „Schwierigkeitsgrad“ der DAE zu quantifizieren.

§ 8.3 Numerische Behandlung linearer DAE-Systeme

Aus Zeitgründen behandeln wir hier exemplarisch nur lineare MSV. Weiterführende Literatur: Hairer, Wanner, 1996, Ascher, Petzold, 1998.

Wir wollen also Verfahren vom Typ

$$y' = f(t,y) \qquad \rightsquigarrow \qquad \sum_{m=0}^{r} \alpha_m\, y_{k+m} = h \sum_{m=0}^{r} \beta_m\, f(t_{k+m}, y_{i+m})$$

verallgemeinern für Probleme

$$E y' = A\, y + f(t) \qquad \rightsquigarrow \qquad ?$$

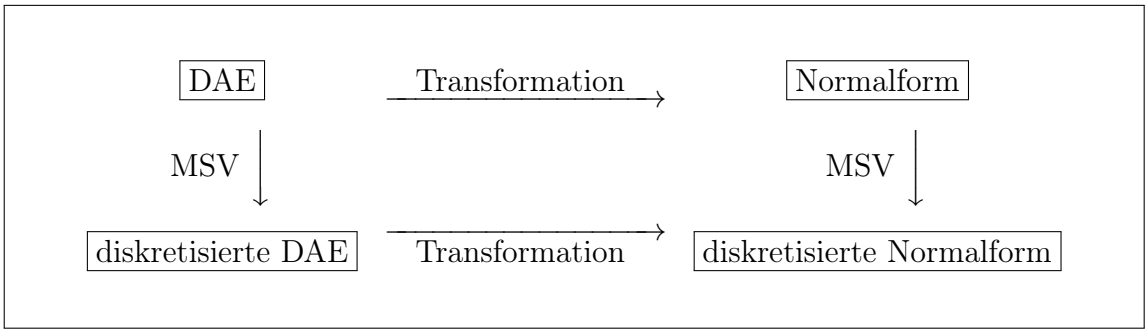
Falls E invertierbar ist, erhält man

$$\sum_{m=0}^{r} \alpha_m\, y_{k+m} = h E^{-1} \sum_{m=0}^{r} \beta_m\, (A\, y_{k+m} + f(t_{k+m})). \tag{8.8}$$

Die kanonische Verallgemeinerung von (8.8) ist

$$E \sum_{m=0}^{r} \alpha_m \, y_{k+m} = h \sum_{m=0}^{r} \beta_m \, (A \, y_{k+m} + f(t_{k+m})). \tag{8.9}$$

Wendet man die Transformationen auf Normalform analog zu (8.7) auf (8.9) an, so ist das Ergebnis gleich dem, wenn man die (DAE) zuerst auf Normalform (8.7) transformiert und dann das (lineare) MSV (8.9) anwendet:



Es reicht daher, zunächst die Eigenschaften des verallgemeinerten linearen MSV (8.9) für DAE in Normalform (8.5) zu betrachten.

Vorteil: Der algebraische und der ODE-Anteil sind entkoppelt. Das Verhalten linearer MSV für ODE wurde in § 4 behandelt. Wir betrachten nun zur Vereinfachung statt des allgemeinen algebraischen Anteils $Ny' = y + f(t)$ (siehe (8.5b)) zunächst die skalare Modell-Gleichung

$$0 \, y' = y + f(t), \tag{8.10}$$

die zu $E = 0$ und $A = 1$ gehört. Die Anwendung des MSV (8.9) auf (8.10) ergibt

$$0 = h \sum_{m=0}^{r} \beta_m (y_{k+m} + f(t_{k+m})). \tag{8.11}$$

Hier sieht man, dass das MSV notwendig implizit sein sollte, damit y_{k+r} vorkommt. Um die algebraische Gleichung (8.10) für den aktuellen Zeitschritt exakt zu erfüllen, muss gelten

$$0 = y_{k+r} + f(t_{k+r}). \tag{8.12}$$

Setzt man voraus, dass die algebraische Gleichung (8.12) für alle vorherigen Zeitschritte (bzw. die Startwerte) erfüllt war, $0 = y_{k+m} + f(t_{k+m}) \quad \forall m = 0, \ldots, r-1$, so ist sie für implizite Verfahren ($\beta_r \neq 0$) wegen (8.11) für $m = r$ auch erfüllt.

Wird die algebraische Gleichung andererseits nur approximativ erfüllt, $\delta_{k+m} = y_{k+m} + f(t_{k+m}) \quad \forall m = 0, \ldots, r-1$, dann folgt aus (8.11)

$$\begin{aligned} \delta_{k+r} = y_{k+r} + f(t_{k+r}) &= -\frac{1}{\beta_r} \sum_{m=0}^{r-1} \beta_m (y_{k+m} + f(t_{k+m})) \\ &= -\frac{1}{\beta_r} \sum_{m=0}^{r-1} \beta_m \delta_{k+m}, \end{aligned}$$

also pflanzt sich der Fehler im Allgemeinen fort.

Um Fehlerakkumulation zu vermeiden, liegt die Forderung

$$\frac{1}{|\beta_r|}\sum_{m=0}^{r-1}|\beta_m| < 1$$

nahe. Ideal wärend demnach also Verfahren mit

$$\beta_r = 1 \quad \text{und} \quad \beta_m = 0, \quad m = 0, \dots, r-1\,.$$

Verlangt man zusätzlich eine möglichst hohe Konsistenzordnung (für den differentiellen Anteil) des MSV bei gegebener Schrittzahl r, so sind das gerade die BDF-Verfahren aus § 4.

Satz 8.18 (Konvergenz der BDF-Verfahren für DAE)**.**
Es sei das Matrixpaar (E, A) regulär. Es sei $y : [0,T] \to \mathbb{C}^n$ die eindeutige Lösung von (DAE) mit konsistenter Anfangsbedingung. Es sei $p \le 6$, und die Lösung gehöre zu $C^{p+1}([0,T];\mathbb{C}^n)$.[74] Dann ist das BDF-Verfahren der Schrittzahl $r = p$ konvergent mit der Ordnung p im Sinne von Definition 4.20.

Skizze: Wie bereits oben bemerkt, ist es hinreichend, die Normalform (8.7) der (DAE) zu betrachten. Nach Satz 4.24 sind die BDF-Verfahren für $r \le 6$ konvergent von der angegebenen Ordnung für den ODE-Anteil (8.5a), da sie nach Satz 4.14 diese Konsistenzordnung besitzen und nach Satz 4.27 nullstabil sind.

Es bleibt der algebraische Anteil (8.5b) zu untersuchen. Wir bezeichnen diesen mit

$$Ny' = y + f(t)$$

und skizzieren den Beweis für nur einen Jordanblock,

$$N = \begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & 0 \end{bmatrix}.$$

Die Anwendung des BDF-Verfahrens führt auf

$$N\sum_{m=0}^{r}\alpha_m\, y_{k+m} = h\left(y_{k+r} + f(t_{k+r})\right). \tag{$*$}$$

Um die Lösungsfolge $\{y_k\}$ zu untersuchen, multiplizieren wir ($*$) mit $N^{\nu-1}$ und teilen durch h:

$$\begin{aligned} 0 &= N^{\nu-1}\, y_{k+r} + N^{\nu-1}\, f(t_{k+r}), \\ \Rightarrow \qquad N^{\nu-1}\, y_{k+r} &= -N^{\nu-1}\, f(t_{k+r}). \end{aligned} \tag{$**$}$$

Die Relation $**$ zwischen y_{k+r} und $f(t_{k+r})$ ist unabhängig von den Startwerten y_{k+m} für $0 \le m < r$, vgl. Lemma 8.12 und unabhängig von h. Im nächsten Schritt multiplizieren wir die Gleichung $*$ mit $N^{\nu-2}$ und nutzen $**$ rekursiv aus. Nach ν Schritten, und ausreichend fortgeschrittenem Verfahren, erhalten wir y_{k+m} in Abhängigkeit von $f(t_{k+m})$, unabhängig von der Schrittweite.

[74] Hinreichend dafür ist in der Weierstraßschen Normalform (8.5): $f_1 \in C^p([0,T];\mathbb{C}^{n-\nu})$ und $f_2 \in C^{\nu+p}([0,T];\mathbb{C}^{\nu})$, siehe Lemma 8.12.

Um die Eigenschaften der übrigen Lösungskomponenten zu untersuchen, definieren wir den diskreten Ableitungsoperator D_h durch

$$D_h y_{k+r} = \frac{1}{h} \sum_{m=0}^{r} \alpha_m y_{k+m}.$$

Damit wird (∗) zu

$$N D_h y_{k+r} = y_{k+r} + f(t_{k+r}).$$

Da D_h linear ist und mit N kommutiert, erhält man formal

$$\begin{aligned} y_{k+r} &= -(I - N D_h)^{-1} f(t_{k+r}) && \text{(Neumannsche Reihe)} \\ &= -\sum_{l=0}^{\infty} N^l D_h^l f(t_{k+r}) && \text{Nilpotenz} \\ &= -\sum_{l=0}^{\nu-1} N^l D_h^k f(t_{k+r}). \end{aligned}$$

Da das BDF-Verfahren konsistent ist von der Ordnung $p = r$ (vgl. Definition 4.7) gilt mit (8.3)

$$D_h f(t_{k+r}) = f'(t_{k+r}) + \sum_{q \geq r} c_q \frac{h^q}{q!} f^{(q)}(t_{k+r}) = f'(t_{k+r}) + \mathcal{O}(h^r).$$

Wendet man D_h auf diese Beziehung an, erhält man

$$\begin{aligned} D_h^2 f(t_{k+r}) &= D_h f'(t_{k+r}) + \sum_{q \geq r} c_q \frac{h^q}{q!} D_h f^{(q)}(t_{k+r}) \\ &= f''(t_{k+r}) + \sum_{q \geq r} \bar{c}_q \frac{h^q}{q!} f^{(q+1)}(t_{k+r}) \\ &= f''(t_{k+r}) + \mathcal{O}(h^r). \end{aligned}$$

Wiederholt man dies ergibt sich

$$D_h^l f(t_{k+r}) - f^{(l)}(t_{k+r}) = \mathcal{O}(h^r), \qquad l = 0, \ldots, \nu - 1$$

und damit

$$y_{k+r} - y(t_{k+r}) = -\sum_{l=0}^{\nu-1} N^l (D_h^l f(t_{k+r}) - f^{(l)}(t_{k+r})) = \mathcal{O}(h^r).$$

Also konvergiert auch der DAE-Teil der Lösung wie behauptet. □

Bemerkung 8.19 (zur Konvergenz von BDF-Verfahren)**.**

(a) Damit ist für konstante Schrittweiten die optimale Konvergenzordnung der BDF-Verfahren mit $r \leq 6$ für (DAE) gezeigt. Für lineare DAE mit variablen Koeffizienten verhält sich dies jedoch bereits wesentlich komplizierter, siehe Kunkel, Mehrmann, 1994.

(b) Bei variabler Schrittweite kann es für DAE-Systeme vom Index $\nu \geq 3$ zu einer Reduktion der Konvergenzordnung kommen, siehe Kunkel, Mehrmann, 1994, Bemerkung 38.

Bemerkung 8.20 (Lösung der LGS in BDF-Verfahren)**.**
In jedem Schritt des BDF-Verfahrens für (DAE) ist ein LGS der Form

$$\Big(\frac{1}{h}\alpha_r\, E - A\Big) y_{k+r} = f(t_{k+r}) - \frac{1}{h}\sum_{m=0}^{r-1} \alpha_m\, y_{k+m} \tag{8.13}$$

zu lösen. Da das charakteristische Polynom $p(\lambda) = \det(\lambda E - A)$ für reguläre Matrixpaare nicht identisch null ist, ist es entweder konstant, oder es gilt $|p(\lambda)| \to \infty$ für $|\lambda| \to \infty$. Daraus folgt, dass das LGS (8.13) für $h \le \overline{h}$ eindeutig lösbar ist.

Für eine ausführliche Behandlung von DAEs (auch im allgemeinen Fall) verweisen wir auf die Bücher Hairer, Wanner, 1996 oder Ascher, Petzold, 1998.

§ 9 Symplektische Verfahren

In der Mechanik werden die Bewegungsgleichungen für Mehrkörpersysteme oft aus Überlegungen zu Energiebilanzen hergeleitet ((Euler-)Lagrange-Theorie, Hamilton-Theorie). Ein wichtiges und allgemeines Modell für Systeme in denen Energieerhaltung gilt, sind die Hamiltonschen Bewegungsgleichungen

$$\dot{p}_i = -\frac{\partial H}{\partial q_i}, \qquad \dot{q}_i = +\frac{\partial H}{\partial p_i}, \tag{9.1}$$

$i = 1, \dots, n$, wobei die Hamilton-Funktion $H(p, q)$, $H : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}$ die Erhaltungsgröße des Systems darstellt, $q \in \mathbb{R}^n$ heißen verallgemeinerte Koordinaten, $p \in \mathbb{R}^n$ verallgemeinerte Impulse (verwandt mit Geschwindigkeiten). Aus (9.1) folgt, dass

$$\begin{aligned}\frac{\mathrm{d}H(p(t), q(t))}{\mathrm{d}t} &= \frac{\partial H}{\partial p}\cdot\dot{p} + \frac{\partial H}{\partial q}\cdot\dot{q} \\ &= -\frac{\partial H}{\partial p}\cdot\frac{\partial H}{\partial q} + \frac{\partial H}{\partial q}\cdot\frac{\partial H}{\partial p} = 0,\end{aligned}$$

also ist $H(p(t), q(t))$ konstant.

Beispiel 9.1 (Federschwinger/Harmonischer Oszillator)**.** Die Energie in einem Federschwinger (siehe Abbildung 9.1) setzt sich zusammen aus

$$E = T + U$$

T**:** kinetische Energie, $T(v) = \frac{1}{2} m\, v^2$,

U**:** potentielle Energie, $U(x) = \frac{1}{2} k\, x^2$,

wobei m die Masse des Schwingkörpers, $v = \dot{x}$ die Geschwindigkeit und k die Federkonstante bezeichnet.

Ohne Reibung und äußere Einflüsse gilt $E = T + U$ ist konstant. Mit

$$\begin{aligned} H(p, q) &:= \frac{1}{2}\frac{1}{m} p^2 + \frac{1}{2} k\, q^2, \\ p &:= m\, \dot{x} && \text{(kinetischer Impuls)}, \\ q &:= x \end{aligned}$$

liefert (9.1) die Bewegungsgleichung

$$\dot{p} = -\frac{\partial H}{\partial q} = -k\,q = -k\,x, \qquad \dot{q} = +\frac{\partial H}{\partial p} = \frac{1}{m}\,p = \dot{x}.$$

$$\left(\Rightarrow \qquad \ddot{q} = \frac{1}{m}\,\dot{p} = \frac{-k}{m}\,q. \right)$$

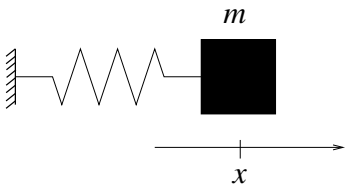


ABBILDUNG 9.1. Federschwinger.

Ein RKV welches für jede Hamilton-Dgl (9.1) garantiert dass $H(p_h(t), q_h(t))$ konstant ist, ist bis heute nicht bekannt. Aber mit einer anderen Erhaltungseigenschaft hat man mehr Erfolg:

Definition 9.2 (Symplektische Abbildungen)**.** Es sei

$$J := \begin{pmatrix} 0 & I_{n\times n} \\ -I_{n\times n} & 0 \end{pmatrix}. \tag{9.2}$$

Eine differenzierbare Abbildung $g : \mathbb{R}^{2n} \mapsto \mathbb{R}^{2n}$ heißt **symplektisch**, wenn für die Jacobimatrix g'

$$g'(p,q)^\top J g'(p,q) = J$$

überall in $\mathbb{R}^{2n}$ gilt.

Satz 9.3. Es sei $H(p,q)$ zweimal stetig differenzierbar auf $\mathbb{R}^{2n}$. Wir betrachten die Lösung $(p(t), q(t))^\top$ des AWP bestehend aus Hamilton-Dgl (9.1) und AB

$$p(0) = p_0, \qquad q(0) = q_0$$

und die damit definierte Abbildung $\varphi_t(p_0, q_0) := (p(t), q(t))^\top$, den **Fluss** der Dgl.

Für jedes feste $t \in \mathbb{R}$ ist φ_t eine symplektische Abbildung.

Beweis: Siehe Hairer, Lubich, Wanner, 2006, Theorem VI.2.4. □

Definition 9.4 (Symplektische ESV)**.** Ein ESV

$$y_{i+1} = y_i + h_i\,\Phi(t_i, y_i, h_i)$$

heißt **symplektisch** genau dann, wenn die durch Anwendung auf die Hamilton-Dgl (9.1) definierte Abbildung

$$g : \ (p_i, q_i) \mapsto (p_{i+1}, q_{i+1})$$

für jede hinreichend glatte Hamilton-Funktion H symplektisch ist.

Die Bedeutung ergibt sich u.a. aus:

Satz 9.5. Es sei $G \in \mathbb{R}^{2n\times 2n}$ eine beliebige symmetrische Matrix. Für jedes symplektische ESV ist die quadratische Hamilton-Funktion

$$H(p(t), q(t)) := \begin{pmatrix} p \\ q \end{pmatrix}^\top G \begin{pmatrix} p \\ q \end{pmatrix}$$

konstant für die Näherungslösung $(p_h(t), q_h(t))^\top$.

Beweis: Siehe Hairer, Nørsett, Wanner, 1993, Theorem II.16.7. □

Darüber hinaus beobachtet man auch für allgemeineres H, dass symplektische Verfahren die Energie im System auch über sehr große Integrationsintervalle $[0, T]$ nahezu erhalten und brauchbare Näherungslösungen ergeben, während nicht-symplektische Verfahren das Langzeitverhalten solcher Systeme nur mit sehr hohem Aufwand zufriedenstellend wiedergeben können, Hairer, Nørsett, Wanner, 1993, Theorem II.16.7.

Beispiel 9.6. Die **symplektischen Eulerverfahren**

$$p_{k+1} := p_k - h\,\frac{\partial H}{\partial q}(p_{k+1}, q_k), \qquad q_{k+1} := q_k + h\,\frac{\partial H}{\partial p}(p_{k+1}, q_k) \tag{9.3}$$

und

$$p_{k+1} := p_k - h\,\frac{\partial H}{\partial q}(p_k, q_{k+1}), \qquad q_{k+1} := q_k + h\,\frac{\partial H}{\partial p}(p_k, q_{k+1}) \tag{9.4}$$

sind beide symplektisch. Sie zählen zu den **geteilten RKV**, bei denen die Steigungen und der Schritt unterschiedliche Koeffizienten für die p- und q-Komponenten verwenden dürfen.

Beweis: Als HA in der Übung. □

Beispiel 9.7.

(a) Explizites, implizites Euler-Verfahren und Crank-Nicolson sind nicht symplektisch.

(b) Alle (impliziten) Gauß-Verfahren (Beispiel 3.39) sind symplektisch, also z. B. auch die implizite Mittelpunktsregel.

(c) Die impliziten Verfahren Radau IA, Radau IIA, Lobatto IIIA und Lobatto IIIB sind nicht symplektisch (Beispiele 3.40 und 3.41).

Ein allgemeines Kriterium für symplektische RKV liefert der folgende Satz.

Satz 9.8. Zu einem RKV mit Butcher-Diagramm $\begin{array}{c|c} a & B \\ \hline & c^\top \end{array}$ sei $M \in \mathbb{R}^{s\times s}$ definiert durch

$$M_{ij} := \gamma_i\beta_{ij} + \gamma_j\beta_{ji} - \gamma_i\gamma_j.$$

Das Verfahren ist symplektisch genau dann, wenn $M = 0$.

Beweis: Siehe Hairer, Nørsett, Wanner, 1993, Theorem II.16.6 + Literaturangabe dort. □

Folgerung 9.9. Es gibt keine konsistenten symplektischen ERKV.

Beweis: Betrachte die Spur von M,

$$\operatorname{trace}(M) = \sum_{i=1}^{s} M_{ii}.$$

Da für ERKV alle $\beta_{ii} = 0$, ist

$$\operatorname{trace}(M) = -\sum_{i=1}^{s} \gamma_i^2.$$

Falls $M = 0$ gelten würde, dann wäre also $c^\top = 0$, und somit das Verfahren nicht konsistent. □

In § 3.6 wurde eine Verallgemeinerung der RKV eingeführt, die speziell für Differentialgleichungen zweiter Ordnung entwickelt wird und sich nicht als ein RKV für das auf erste Ordnung reduzierte System darstellen lässt. Unter diesen Verfahren gibt es dann auch explizite symplektische Verfahren.

Literatur

Ascher, U. M.; Petzold, L. R. (1998). *Computer methods for ordinary differential equations and differential-algebraic equations*. Philadelphia, PA: Society for Industrial und Applied Mathematics (SIAM).

Becker, R.; Meidner, D.; Vexler, B. (2007). „Efficient Numerical Solution of Parabolic Optimization Problems by Finite Element Methods“. *Optimization Methods and Software* 22.5, S. 813–833. DOI: 10.1080/10556780701228532.

Dahlquist, G. (1956). „Convergence and stability in the numerical integration of ordinary differential equations“. *Mathematica Scandinavica* 4, S. 33–53.

Dahlquist, G. G. (1963). „A special stability problem for linear multistep methods“. *BIT* 3, S. 27–43.

Hairer, E.; Nørsett, S.; Wanner, G. (1993). *Solving Ordinary Differential Equations I. Nonstiff Problems*. Bd. 8. Springer Series in Computational Mathematics. Berlin: Springer.

Hairer, E.; Wanner, G. (1996). *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*. Bd. 14. Springer Series in Computational Mathematics. Berlin: Springer.

Hairer, E.; Lubich, C.; Wanner, G. (2006). *Geometric numerical integration*. 2. Aufl. Bd. 31. Springer Series in Computational Mathematics. Structure-preserving algorithms for ordinary differential equations. Berlin: Springer-Verlag.

Hanke-Bourgeois, M. (2006). *Grundlagen der Numerischen Mathematik und des Wissenschaftlichen Rechnens*. Stuttgart: Teubner.

Hermann, M. (2004). *Numerik gewöhnlicher Differentialgleichungen*. München: Oldenbourg.

Heuser, H. (1991). *Gewöhnliche Differentialgleichungen*. Stuttgart: Teubner.

Kunkel, P.; Mehrmann, V. (1994). *Analysis und Numerik linearer differentiell-algebraischer Gleichungen*. Techn. Ber. SFB393/94–27. TU Chemnitz.

Langer, U. (1996). *Skript zur Vorlesung Numerik III: Numerische Verfahren für Anfangs- und Anfangsrandwertaufgaben*. Lecture Notes.

Plato, R. (2004). *Numerische Mathematik kompakt*. 2. Aufl. Wiesbaden: Vieweg.

Rannacher, R. (2008). *Numerische Mathematik I — Numerik gewönlicher Differentialgleichungen*. Vorlesung an der Universität Heidelberg.

Schötzau, D.; Schwab, C. (2000). „An *hp* a priori error analysis of the DG time-stepping method for initial value problems“. *Calcolo* 37.4, S. 207–232.

Stoer, J. (2004). *Numerische Mathematik Band 1*. 9. Aufl. Berlin: Springer.

Stoer, J.; Bulirsch, R. (2005). *Numerische Mathematik Band 2*. 4. Aufl. Berlin: Springer.

Index

1. assoziiertes Polynom, 66
1. charakteristisches Polynom, 66
2. assoziiertes Polynom, 66
2. charakteristisches Polynom, 66

A-Stabilität, 43
ABM-Verfahren, 89
ABM33-Verfahren, 90
Abschneidefehler, *siehe* Konsistenzfehler
A-Stabilität, 43
Adams-Bashforth-Verfahren, 63
Adams-Moulton-Verfahren, 64
Adams-Verfahren
  explizite, 63
  implizite, 64
Algebro-Differentialgleichung, 108
Anfangswert, 5
Anfangswertproblem, 5
  autonom, 5
äquidistantes Gitter, 12

BDF-Verfahren, 64, 65
  für DAE, 120
Butcher-Diagramm, 26
  Crank-Nicolson-Verfahren, 98
  3/8-Regel, 33
  explizites Euler-Verfahren, 27
  implizites Euler-Verfahren, 27
  Methode von Heun dritter Ordnung, 32
  Methode von Kutta dritter Ordnung, 32
  Mittelpunktsregel
    explizite, 27
    implizite, 27
  Theta-Methode, 47
  Trapezregel
    implizite, 27
  verbessertes Euler-Verfahren, 27
  Verfahren von Heun, 27
  Verfahren von Runge-Kutta vierter Ordnung, 32
Butcher-Tableau, 26

charakteristisches Polynom
  Matrixpaar (DAE), 110
charakteristisches Polynom
  erstes (MSV), 66
  zweites (MSV), 66
Crank-Nicolson-Verfahren, 20, 98, 148

DAE, 108
  algebraischer Anteil, 113
  Weierstraßsche Normalform, 113
Dahlquist'sche Testgleichung, 33
Dahlquist-Schranke
  erste, 82
  zweite, 84
Deskriptor-System, 108
DG-Verfahren, 93
Differentialgleichung
  autonome, 28
  steif, 47
differentiell-algebraische Gleichung, 108
  Index, 115
Differenzengleichung
  lineare homogene, 71
Differenzenquotienten, 136
DIRK, *siehe* Runge-Kutta-Verfahren, diagonal-implizites
Diskretisierungsfehler
  globaler
    Einschrittverfahren, 22
    Mehrschrittverfahren, 74
  lokaler
    Einschrittverfahren, 20
    Mehrschrittverfahren, 66
Diskretisierungsparameter, 12
DOPRI5(4), 57
3/8-Regel, 33

Eigenvektor
  verallgemeinert, 111
Eigenwert
  verallgemeinert, 111
Eigenwertproblem
  verallgemeinertes, 111
eingebettete Runge-Kutta-Verfahren, 56
Einschrittverfahren, 17
  explizites, 17
  implizites, 17
  konsistentes, 18
  konvergentes, 22

L-stabil, 52
Endzeit, 5
ESV, *siehe* Einschrittverfahren
Euler-Verfahren
explizites, 13, 19
implizites, 19
linear implizites, 106
verbessertes, 19, 27
explizite Mittelpunktsregel, 68
explizites Euler-Verfahren, 13
Extrapolation
lokale, 58, 153

Fehlerordnung
Einschrittverfahren, 18
Mehrschrittverfahren, 66
Finite-Differenzen-Verfahren, 135
Fundamentallösung, 72

Galerkin-Verfahren, 93
Gauß-Verfahren, 39
Gauß-Verfahren, 39
Geisterpunkte, 137
Gitter, 12
äquidistant, 12
Schrittweiten, 12
Stützstellen, 12
Gitterweite, 12
globaler Diskretisierungsfehler
Einschrittverfahren, 22
globaler Diskretisierungsfehler
Mehrschrittverfahren, 74

horizontale Linienmethode, 143

implizite Mittelpunktsregel, 39, 148
implizite Trapezregel, 27, 98, 148
implizites Euler-Verfahren, 19
Index (DAE), 115
Inkrementfunktion
Einschrittverfahren, 17
Mehrschrittverfahren, 62
IRK, *siehe* Runge-Kutta-Verfahren, implizites

Konsistenz
Einschrittverfahren, 18
Mehrschrittverfahren, 66
Konsistenzfehler
Einschrittverfahren, 17
Mehrschrittverfahren, 66
Konsistenzordnung
Einschrittverfahren, 18
Mehrschrittverfahren, 66
Konvergenz
Einschrittverfahren, 22
Mehrschrittverfahren, 75
Konvergenzordnung
Einschrittverfahren, 22
Mehrschrittverfahren, 75
Korrektor-Verfahren, 89

Legendre-Polynom, 39
linear implizite Mittelpunktsregel, 106
linear implizites Euler-Verfahren, 106
linear implizites Runge-Kutta-Verfahren, 104
Lipschitz-Bedingung, 22
LIRK, *siehe* Runge-Kutta-Verfahren, linear implizites
lokale Extrapolation, 58
lokale Extrapolation, 153
lokaler Diskretisierungsfehler
Einschrittverfahren, 20
Mehrschrittverfahren, 66

Matrixbüschel, 110
Matrixpaar, 109
charakteristisches Polynom, 110
Index, 115
regulär, 110
singulär, 110
Weierstraßsche Normalform, 112
Matrixpaare, äquivalente, 109
Mehrschrittverfahren, 62
A(α)-stabiles, 85
A-stabiles, 83
absolut stabiles, 83
Anlaufphase, 62
Dahlquist-stabiles, *siehe* Mehrschrittverfahren, nullstabiles
explizites, 62
implizites, 62
konsistentes, 66
konvergentes, 75
lineares, 62
nullstabiles, 74
schwach stabiles, 145
stabiles, *siehe* Mehrschrittverfahren, nullstabiles
Stabilitätspolynom, 83
stark stabiles, 145
steif-stabiles, 88
Wurzelbedingung, 74
wurzelstabiles, *siehe* Mehrschrittverfahren, nullstabiles
Methode von Euler, 13
Methode von Heun dritter Ordnung, 32
Methode von Kutta dritter Ordnung, 32
Milne-Methode, 68
Mittelpunktsregel
explizite, 27, 68
implizite, 27, 39, 148
linear implizite, 106
MSV, *siehe* Mehrschrittverfahren

Nilpotenzindex, 113
Normalform
  Weierstraßsche, 112, 113

parasitäre Komponente, 71
P(EC)^M-Verfahren, 89
P(EC)$^M E$-Verfahren, 89
Petrov-Galerkin-Verfahren, 98
Polygonzugmethode, 13
Prädiktor-Verfahren, 88

Quadratur-Verfahren
  Genauigkeitsgrad, 30, 38

Radau-I-Verfahren, 40
Radau-II-Verfahren, 40
Randbedingung
  Dirichlet, 133
  Neumann, 133
Randwertproblem (RWP), 133
rechte Seite, 5
reguläres Matrixpaar, 110
RKF2(3), 56
RKF4(5), 57
Rosenbrock-Typ, 104
Rosenbrock-Verfahren, 104
Rosenbrock-Wanner-Verfahren,ROW-Verfahren, 104
Rothe-Methode, 143
Rückwärts-Euler-Verfahren, 27
Runge-Kutta-Fehlberg-Verfahren, 56
Runge-Kutta-Nyström-Verfahren, 60
Runge-Kutta-Verfahren, 26
  diagonal-implizites, 37
  einfach diagonal-implizites, 37
  eingebettete, 56
  implizites, 37
  linear implizites, 104
Runge-Kutta-Verfahren, semi-implizites, 105
RWP, 133

Schießmethoden, 134
Schrittweitensteuerung, 53
schwache Formulierung, 92
SDIRK, *siehe* Runge-Kutta-Verfahren, einfach diagonal-implizites
semi-implizites Runge-Kutta-Verfahren, 105
singuläres Matrixpaar, 110
stabil
  (MSV), 74
  A(α)-stabil, 85
  A-stabil (MSV), 83
  absolut-stabil (MSV), 83
  D(ahlquist)-stabil (MSV), 74
  L-stabil (ASV), 52
  nullstabil (MSV), 74
  stark stabil (MSV), 145
  steif-stabil, 88
Stabilitätsbedingung, 22
Stabilitätsbereich, 43
Stabilitätsgebiet
  Mehrschrittverfahren, 83
Stabilitätsfunktion
  Einschrittverfahren, 43
Stabilitätspolynom, 83
steif-stabil, 88
steife Differentialgleichung, 47
Steifigkeitsquotient, 47
stetige Runge-Kutta-Verfahren,Runge-Kutta-Verfahren, 154

θ-Methode, 46
Trapezlregel
  implizite, 148
Trapezregel
  implizite, 27, 98

unstetiges Galerkin-Verfahren, 93

Variationsformulierung, 92
verallgemeinerten Eigenvektor, 111
verallgemeinerten Eigenwert, 111
verallgemeinertes Eigenwertproblem, 111
vereinfachende Annahmen von Butcher, 100
Verfahren von Heun, 20, 27, 57
Verfahren von Runge-Kutta vierter Ordnung, 32
Verfahrensfunktion
  Einschrittverfahren, 17
  Mehrschrittverfahren, 62
vertikale Linienmethode, 143
Vorwärts-Euler-Verfahren, 13
Vorwärts-Euler-Verfahren, 27

W-Methode, 104
Wärmeleitungsgleichung, 140
Weierstraßsche Normalform, 112, 113
Wurzelortskurve, 83

KAPITEL 2

Numerische Lösung von Randwertaufgaben

Definition 9.10 (Randwertproblem)**.** Sei $\Omega := (a, b) \subset \mathbb{R}$ ein Intervall. Das Problem lautet

Finde $y \in X$, so dass

$$\begin{cases} [L\,y](x) &= f(x) & \text{für } x \in \Omega, \\ y(a) &= y_a \qquad y(b) = y_b \end{cases} \tag{RWP}$$

mit gegebenen Randwerten $y_a, y_b \in \mathbb{R}^n$ und einem Differentialoperator $L\colon X \to C(\overline{\Omega})$ heißt **Randwertaufgabe**.

Der Einfachheit halber betrachten wir lediglich lineare Randwertprobleme zweiter Ordnung, das bedeutet, der Operator $L\colon C^2(\Omega) \cap C(\overline{\Omega}) \to C(\overline{\Omega})$ besitzt die Gestalt

$$L\,y = y''(x) + P(x)y'(x) + Q(x)y(x) \tag{9.5}$$

mit Funktionen $P, Q\colon \Omega \to \mathbb{R}^{n\times n}$. Wir nennen die Variable nun x und nicht t, da derartige Probleme meist für ortsabhängige Größen y auftauchen.

Beispiel 9.11 (Erwärmung eines Stabes)**.** Beispielsweise beschreibt das Gebiet Ω einen Stab der Länge $b - a$. Die Lösung y der Gleichung

$$-(k(x)y'(x))' + q(x)(y(x) - y_0(x)) = f(x) \qquad \text{in } \Omega$$

beschreibt die Temperaturverteilung im Stab. Hierbei ist

- y_0 die Umgebungstemperatur
- $k \geq k_0 > 0$ die Wärmeleitfähigkeit des Stabes
- $q > 0$ der Wärmeaustauschkoeffizient
- f die Dichte der Energiequelle im Stab

Damit das Problem eindeutig lösbar wird müssen wir uns an den 2 Endpunkten des Gebiets Randbedingungen vorgeben, beispielsweise

$$y(a) = y_a, \qquad y(b) = y_b.$$

Diese Art von Randbedingungen werden auch als **Dirichlet-Randbedingungen** bezeichnet und beschreiben das Erwärmen/Abkühlen des Stabes am Rand auf eine festgelegte Temperatur. Oft werden auch sogenannte **Neumann-Randbedingungen** gefordert. Diese besitzen die Form

$$y'(a) = -v_a, \qquad y'(b) = v_b.$$

Beispielsweise bedeutet der Fall $v_a = v_b = 0$, dass das System isoliert ist, das heißt am Rand wird keine Wärme hinzugefügt oder entzogen.

§ 10 Schießmethoden

Idee: Forme die Differentialgleichung (RWP) in ein System 1. Ordnung um und steuere die Anfangsbedingung der neuen unbekannten Größe, so dass der Randwert an der Stelle $x = b$ "getroffen" wird.

Wir führen also einen Vektor $u = (y,\ z)$ ein mit $y' = z$. Angewendet auf die DGL (RWP) mit dem Operator (9.5) ergibt sich

$$\begin{pmatrix} y' \\ z' \end{pmatrix} = \begin{pmatrix} 0 & I_{n\times n} \\ -Q(x) & -P(x) \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} + \begin{pmatrix} 0 \\ f(x). \end{pmatrix} \tag{10.1a}$$

mit Anfangsbedingung

$$\begin{pmatrix} y \\ z \end{pmatrix}(a) = \begin{pmatrix} y_a \\ s \end{pmatrix} \tag{10.1b}$$

Beachte: Dieses System ist äquivalent zum Anfangswertproblem 2. Ordnung

$$\begin{cases} y''(s;\cdot) + Py'(s;\cdot) + Qy(s;\cdot) = f \quad \text{in } \Omega \\ y(s;a) = y_a, \quad y'(s;a) = s \qquad\qquad . \end{cases} \tag{10.2}$$

Ziel: Finde einen geeigneten Anstieg s an der Stelle a (Beachte: $z(a) = y'(a)$), so dass $y(b) = y_b$ erfüllt ist.

Nun wird der Begriff "Schießmethode" offensichtlich. Wir versuchen quasi einen Ball so zu werfen, dass er im Punkt (b, y_b) landet. Dabei können wir den Abwurfwinkel variieren. Zur Vereinfachung der Darstellung betrachten wir den skalaren Fall $n = 1$.

Wir haben bereits in Kapitel 1 gelernt, wie wir Anfangswertprobleme lösen können. Die Lösung von (10.1) zum Anstieg s wird nun mit $y(s;x)$ bezeichnet. Eine Lösung löst dann das Randwertproblem (RWP), falls

$$F(s) := y(s;b) - y_b = 0. \tag{10.3}$$

Wir müssen also die Nullstelle(n) einer nicht-linearen Funktion ermitteln.

Numerische Umsetzung: Wir wollen ein ein Newton-Verfahren zur Lösung der Gleichung (10.3) anwenden. Dafür benötigen wir die Ableitung $F'(s) = \frac{\partial}{\partial s} y(s;b)$. Die unbekannte Funktion $v := \frac{\partial}{\partial s} y(s;\cdot)$ erhalten wir nach Differentiation der Gleichung (10.2) nach s:

$$v'' + Pv' + Qv = 0 \text{ in } , \qquad v(0) = 0, \quad v(a) = 1. \tag{10.4}$$

Dies ist wieder ein Anfangswert-Problem und kann mit einem geeigneten Verfahren gelöst werden.

Somit gilt:

$$F'(s) = v(b).$$

Damit können wir nun einen Lösungsalgorithmus formulieren:

Algorithmus 10.1.

Eingabe: Initial value s_0, Input data: P, Q, f, y_a, y_b.

1: Set $k = 0$.
2: Compute v from (10.4).
3: **while** Stopping criterion not fulfilled **do**
4: Compute $y(s_k)$ from (10.1).

5: Compute
$$\delta s = -\frac{F(s_k)}{F'(s_k)} = \frac{y_b - y(s_k; b)}{v(b)}$$
6: Update $s_{k+1} = s_k + \delta s$, set $k \to k+1$.
7: **end while**

Bemerkung 10.2. (a) Ein sinnvolles Abbruchkriterium wäre $F(s_k) < ATOL$ mit einer absoluten Toleranz $ATOL$ in der Größenordnung 10^{-6} bis 10^{-10}.

(b) Die Gleichung (10.4) muss hier lediglich 1 mal gelöst werden, da diese unabhängig von s ist. Das ist nicht mehr der Fall, wenn das betrachtete Randwertproblem nicht-linear ist.

(c) Für vektorwertige Randwertprobleme muss die Nullstellengleichung (10.3) modifiziert werden. Eine geeignete Wahl wäre
$$F(s) = \|y(s; b) - y_b\|,$$
wobei $\|\cdot\|$ die euklidische Norm ist. Ein entsprechendes Newton-Verfahren kann als Übungsaufgabe hergeleitet werden.

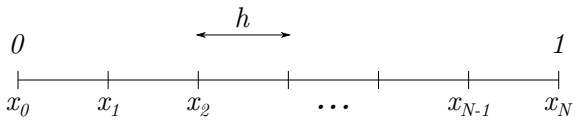
(d) Bei linearen Differentialgleichungen ist auch die Funktion F linear. Dann findet die Schießmethode nach genau einem Newton-Schritt die exakte Lösung. [1]

§ 11 Finite-Differenzen-Verfahren

§ 11.1 Implementierung

In diesem Abschnitt wollen wir ein Randwertproblem der Form (RWP) mit dem sogenannten **Finite-Differenzen-Verfahren** lösen.[2]. Dazu geht man wie folgt vor:

- Ersetze das Gebiet $\Omega = (0,1)$ wird durch ein Gitter $\Omega_h := \{x_i \in \Omega \colon x_i = i\,h, i = 1, \dots, N-1\}$ mit Schrittweite h.



  Der Rand $\Gamma = \{0, 1\}$ wird durch $\Gamma_h = \{x_0, x_N\}$ ersetzt. Analog zu $\overline{\Omega} = \Omega \cup \Gamma$ setzen wir $\overline{\Omega}_h = \Omega_h \cup \Gamma_h$.

- Die gesuchte Funktion $y \colon \overline{\Omega} \to \mathbb{R}$ wird durch eine Gitterfunktion $y_h \colon \overline{\Omega}_h \to \mathbb{R}$ ersetzt.

  Später benötigen wir auch den Restriktionsoperator R_h, welcher kontinuierliche Funktionen $f \colon \overline{\Omega} \to \mathbb{R}$ auf Gitterfunktionen $f_h = R_h f \colon \overline{\Omega}_h \to \mathbb{R}$ abbildet. Häufig verwendet man die Definition
$$[R_h f](x_i) = f(x_i), \qquad i = 0, \dots, N.$$

[1] Hierfür ist ein Matlab-Programm `shooting_test.m` vorhanden.
[2] Angelehnt an das Skript "Differentialgleichungen" von Thomas Apel

- Die in der DGL auftauchenden Ableitungen ersetzt man durch **Differenzenquotienten**: $y'(x)$ kann durch
  - ▷ $[D_h^+ y](x) = \frac{1}{h}(y(x+h) - y(x))$ (Vorwärtsdifferenz)
  - ▷ $[D_h^- y](x) = \frac{1}{h}(y(x) - y(x-h))$ (Rückwärtsdifferenz)
  - ▷ $[D_h^0 y](x) = \frac{1}{2h}(y(x+h) - y(x-h))$ (Zentrale Differenz)

  approximiert werden. Für die zweiten Ableitungen $y''(x)$ verwenden wir häufig
  - ▷ $[D_h^+ D_h^- y](x) = \frac{1}{h^2}(y(x-h) - 2y(x) + y(x+h))$ (Zweite Differenz).

Als Beispiel betrachten wir die Differentialgleichung

$$[Ly](x) := -y''(x) + c(x)\,y(x) = f(x) \qquad \text{in } \Omega := (0,1)$$

mit homogenenen Dirichlet-Randbedingungen $y(0) = y(1) = 0$.

Wir können nun in jedem Gitterpunkt $x_i \in \Omega_h$, $i = 1, \ldots, N-1$, eine Gleichung aufstellen um die $N-1$ Unbekannten $y_i := y_h(x_i)$, $i = 1, \ldots, N-1$, zu berechnen.

Wertet man die Gleichung mit den approximierten Ableitungen im Gitterpunkt x_i, $i = 1, \ldots, N-1$ aus, so erhält man

$$\frac{-y_h(x_i - h) + 2y_h(x_i) - y_h(x_i + h)}{h^2} + c(x_i)y_h(x_i) = f(x_i), \qquad i = 1, \ldots, N-1.$$

Mit $y_h(x_i) = y_i$, $y_h(x_i \pm h) = y_h(x_{i\pm1}) = y_{i\pm1}$ lässt sich diese Gleichung kompakt aufschreiben:

$$\frac{-y_{i-1} + 2y_i - y_{i+1}}{h^2} + c_i y_i = f_i, \qquad i = 1, \ldots, N-1, \tag{11.1}$$

mit der Kurzschreibweise $f_i = f(x_i)$ und $c_i = c(x_i)$. Diese Gleichungen bilden ein lineares Gleichungssystem, welches in Matrix-Vektor-Notation die Gestalt

$$\begin{pmatrix} \frac{2}{h^2} + c_1 & -\frac{1}{h^2} & & & \\ -\frac{1}{h^2} & \frac{2}{h^2} + c_2 & -\frac{1}{h^2} & & \\ & \ddots & \ddots & \ddots & \\ & & -\frac{1}{h^2} & \frac{2}{h^2} + c_{N-2} & -\frac{1}{h^2} \\ & & & -\frac{1}{h^2} & \frac{2}{h^2} + c_{N-1} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N-2} \\ y_{N-1} \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{N-2} \\ f_{N-1} \end{pmatrix} \tag{11.2}$$

besitzt. Beachte, dass wir direkt die Randwerte $y_0 = y_N = 0$ eingesetzt haben.

Die Systemmatrix ist dünn besetzt, d. h. Sie besitzt maximal 3 Nicht-Null-Einträge pro Zeile. Dies sollte bei der Implementierung beachtet werden (siehe Matlab: `sparse` bzw. `spdiags`).

Bemerkung 11.1 (Andere Randbedingungen). (a) Bei inhomogenen Dirichlet-Randbedingungen $y(0) = g_0$, $y(1) = g_1$ setzt man diese direkt in die Gleichung (11.1) ein, "wirft" die entsprechenden Terme auf die rechte Seite. So erhält man beispielsweise für $i = 1$ die Gleichung

$$\frac{2y_1 - y_2}{h^2} + c_1 y_1 = f_1 + \frac{g_0}{h^2}.$$

Vollkommen analog ändert sich die Gleichung im Gitterpunkt x_{N-1}.

(b) Bei Neumann-Randbedingungen $y'(0) = g_0, y'(1) = g_1$ führt man sogenannte **Geisterpunkte** $x_{-1} = -h$ und $x_{N+1} = (N+1)h$ ein. Diese liegen außerhalb des Gebiets $\overline{\Omega}$. Die Neumann-Randbedingung bringt man dann mit der zentralen Differenz auf und erhält die linearen Gleichungen

$$\frac{y_{-1} - y_1}{2h} = g_0, \qquad \frac{y_{N+1} - y_{N-1}}{2h} = g_1.$$

Somit haben wir $N-1$ Gleichungen für die inneren Punkte, 2 für die Randbedingungen, haben aber nun $N+3$ Unbekannte. Die "fehlenden" 2 Gleichungen sind gerade (11.1) für die Indizes $i = 0$ und $i = N$.

§ 11.2 Konvergenz Finiter-Differenzen-Verfahren

Wir beschränken uns hier auf den Fall

$$Ly = -y'' + cy$$

mit einer positiven Funktion $c(x) > 0$, $x \in \Omega$. Wir betrachten die zugehörige Randwertaufgabe

$$Ly = f \quad \text{in} \quad \Omega := (0,1), \qquad y(0) = g_0,\ y(1) = g_1. \tag{11.3}$$

Für die Lösung dieser Aufgabe gelten 2 wichtige Prinzipien:

- **Maximumsprinzip:**
  Es sei $f \le 0$ und $g_0, g_1 \le 0$. Dann gilt $u \le 0$.
- **Vergleichsprinzip:**
  Es seien y_i, $i = 1, 2$ die Lösungen von (11.3) zu den Daten f_i, $g_{i,0}, g_{i,1}$. unter der Voraussetzung $f_1 \le f_2$ und $g_{1,\cdot} \le g_{2,\cdot}$ gilt $y_1 \le y_2$.

Wir zeigen zunächst, dass diese Prinzipien auch für die Lösungen der diskreten Aufgabe

$$L_h y_h = R_h f \text{ in } \Omega_h, \qquad y_h = g_h \text{ auf } \Gamma_h \tag{11.4}$$

erfüllt sind. Hier ist $L_h y_h = D_h^+ D_h^- y_h + [R_h c] \cdot y_h$.

Lemma 11.2 (Diskretes Maximumsprinzip)**.** Wir nehmen an, dass der Reaktionsparameter strikt positiv ist, d. g. $c \ge c_0 > 0$. Ferner sei $R_h f \le 0$ und $g_h \le 0$. Dann gilt $y_h \le 0$ auf Ω_h.

Beweis: Es sei $j = \arg\max y_j$. Für den Fall $j \in \{0, N\}$ ist die Aussage aufgrund der Annahme an die Randbedingungen gezeigt. Für den Fall $j = 1, \dots, N-1$ setzen wir die Beobachtungen $y_{i+1} \le y_i$ und $y_{i-1} \le y_i$ in die Gleichung (11.1) ein und erhalten

$$c_i y_i = \underbrace{f_i}_{\le 0} - \frac{1}{h^2}(\underbrace{-y_{i+1} + y_i}_{\ge 0} + \underbrace{y_i - y_{i-1}}_{\ge 0}) \le 0.$$

Wegen der Annahme $c_i > 0$ folgt $y_i \le 0$, was der Annahme entspricht. □

Bemerkung 11.3. Die Annahme $c \ge c_0 > 0$ ist nicht zwingend erforderlich, vereinfacht aber den Beweis deutlich. Ein Beweis unter der schwächeren Annahme $c \ge 0$ wird in der Vorlesung "Numerik partieller Differentialgleichungen" diskutiert.

Lemma 11.4 (Diskretes Vergleichsprinzip)**.** Angenommen, $c \geq c_0 > 0$. Es seien v_h, w_h Lösungen der Gleichungen

$$L_h v_h = R_h f_1 \quad \text{und} \quad L_h w_h = R_h f_2$$

Unter der Annahme $f_1 \leq f_2$ und $v_0 \leq w_0$ und $v_N \leq w_N$ gilt

$$v_h \leq w_h \qquad \text{in } \Omega.$$

Beweis: Wende das diskrete Maximumprinzip auf die Gleichung

$$L_h(v_h - w_h) = R_h f_1 - R_h f_2$$

und folgere $(v_h - w_h) \leq 0$. □

Um Konvergenz der Finite-Differenzen-Verfahren zu untersuchen benötigen wir eine geeignete Norm für Gitterfunktionen $v_h \colon \overline{\Omega}_h \to \mathbb{R}$. Wir verwenden die Maximumnorm

$$\|v_h\|_{\Omega_h} = \max_{i=1,\dots,N-1} |v_i|, \qquad \|v_h\|_{\overline{\Omega}_h} = \max_{i=0,\dots,N} |v_i|.$$

Definition 11.5 (Konsistenz, Konvergenz)**.**

Es seien y und y_h die Lösungen von (11.3) bzw. (11.4).

(a) Ein Differenzenverfahren heißt **konvergent mit Ordnung** p, falls eine gitterunabhängige Konstante $C > 0$ existiert, so dass

$$\|R_h y - y_h\|_{\Omega_h} \leq C h^p$$

für alle $h > 0$ erfüllt ist.

(b) Ein FDV heißt **konsistenz mit Ordnung** p, falls $C > 0$ $(C \neq C(h))$ existiert, so dass

$$\|L_h R_h y - R_h \underbrace{Ly}_{=f}\|_{\Omega_h} \leq C h^p$$

für alle $h > 0$ erfüllt ist.

(c) Ein FDV heißt **stabil**, falls $C > 0$ existiert, so dass für alle Gitterfunktionen $v_h \colon \Omega_h \to \mathbb{R}$ mit $v_0 = v_N = 0$ die Abschätzung

$$\|v_h\|_{\overline{\Omega}_h} \leq C \|L_h v_h\|_{\Omega_h}$$

erfüllt ist.

Satz 11.6 (Beziehung zwischen Konsistenz und Konvergenz)**.** Ist ein FDV stabil und konsistent (mit Ordnung p), dann ist es auch konvergent (mit ordnung p).

Beweis: Es gilt

$$\begin{aligned} \|R_h y - y_h\|_{\Omega_h} &\leq C \, \|L_h(R_h y - y_h)\|_{\Omega_h} && \text{(Stabilität)} \\ &\leq C \, \|L_h R_h y - R_h f\| && (L_h y_h = R_h f) \\ &\leq C h^p && \text{(Konsistenz).} \end{aligned}$$

□

Im Folgenden werden wir Konsistenz und Stabilität des Verfahrens (11.1) nachweisen.

Lemma 11.7. Angenommen, die exakte Lösung y von (RWP) ist 4-mal stetig differenzierbar. Dann ist das FDV aus (11.1) **konsistent mit Ordnung 2**, d. h. es exisiert $C > 0$, sodass

$$\|L_h R_h y - R_h f\|_{\Omega_h} \leq C\, h^2$$

erfüllt ist.

Beweis: Wir werten den Konsistenzfehler wieder mit der Taylorformel, siehe Theorem 3.2, im Entwicklungspunkt x_i, $i = 1, \ldots, N-1$ aus. Daraus folgt, es existieren Zwischenstellen $\xi_1 \in [x_i - h, x_i]$, $\xi_2 \in [x_i, x_i + h]$, sodass

$$\begin{aligned}
[L_h R_h y](x_i) &= \frac{1}{h^2}(-y(x_i - h) + 2y(x_i) - y(x_i + h) + c_i y(x_i) \\
&= \frac{1}{h^2}\Big(y'(x_i)\, h - \frac{1}{2}y''(x_i)\, h^2 + \frac{1}{6}y'''(x_i)\, h^3 - \frac{1}{24}y^{(4)}(\xi_1)\, h^4 \\
&\quad - y'(x_i)\, h - \frac{1}{2}y''(x_i)\, h^2 - \frac{1}{6}y'''(x_i)\, h^3 - \frac{1}{24}y^{(4)}(\xi_2)\, h^4\Big) + c(x_i)y(x_i) \\
&= -y''(x_i) + c(x_i)y(x_i) - \frac{1}{24}\left(y^{(4)}(\xi_1) + y^{(4)}(\xi_2)\right) h^2.
\end{aligned}$$

Wir wissen außerdem, dass $-y''(x_i) + c(x_i)y(x_i) = f(x_i) = [R_h f](x_i)$ und somit folgt

$$\|L_h R_h y - R_h f\|_{\Omega_h} \leq Ch^2,$$

wobei C von $y^{(4)}$ abhängt. □

Lemma 11.8. Es sei $c \geq c_0 = 0$.[3] Das FDV aus (11.1) ist stabil, d. h. es gilt

$$\|v_h\|_{\Omega_h} \leq C_S \|L_h v_h\|_{\Omega_h}$$

für alle $v_h \colon \overline{\Omega}_h \to \mathbb{R}$ mit $v_0 = V_N = 0$.

Beweis: Für den Fall $L_h v_h = 0$ folgt sofort (siehe Vergleichsprinzip) $v_h = 0$, und die Aussage ist gezeigt. Andernfalls konstruieren wir eine Vergleichslösung

$$w_h := \frac{\|L_h v_h\|_\Omega}{c_0}$$

und zeigen die Aussage über das Vergleichsprinzip. Wegen $w_h \equiv \text{const}$ gilt

$$[L_h w_h](x_i) = \underbrace{c_i}_{\geq c_0} w_i \geq \|L_h v_h\|_{\Omega_h} \geq \pm[L_h v_h](x_i).$$

Also haben wir gezeigt:

$$-L_h w_h \leq L_h v_h \leq L_h w_h.$$

Ferner gilt $-w_h \leq v_h \leq w_h$ auf Γ_h. Mit dem Vergleichsprinzip (Lemma 11.4) folgt dann

$$-w_h = -\frac{\|L_h v_h\|_{\Omega_h}}{c_0} \leq v_h \leq \frac{\|L_h v_h\|_{\Omega_h}}{c_0} = w_h$$

auf Ω_h und damit die Behauptung. Die Stabilitätskonstante lautet hier also $C_S := \frac{1}{c_0}$.

Folgerung 11.9. Das FDV aus (11.1) ist konvergent mit der Ordnung 2.

□

[3] Auch hier ist diese Annahme nicht notwendig, vereinfacht aber den Beweis dieses Satzes.

§ 12 Numerische Methoden für Anfangs-Randwert-Probleme

Wir wollen hier unsere Untersuchungen zu Anfangswert- und Randwert-Problemen miteinander verbinden und untersuchen sogenannte **Anfangs-Randwert-Probleme**. Der Einfachheit halber beschränken wir uns auf den linearen Fall und untersuchen das Problem

$$\begin{cases} \partial_t y(x,t) - Ly(x,t) &= f(x,t) \quad \text{für } x \in \Omega := (0,1),\ t \in (0,T] \\ y(x,t) &= g(x,t) \quad \text{für } x \in \Gamma := \partial\Omega,\ t \in (0,T], \\ y(x,0) &= y_0(x) \quad \text{für } x \in \Omega. \end{cases} \tag{12.1}$$

Hierbei handelt es sich bereits um eine **partielle Differentialgleichung**, das bedeutet, die gesuchte Lösung hängt von mehr als einer Variablen (hier: Zeit und Ort) ab. Der Operator L ist ein Differentialoperator bezüglich der Ortsvariablen, wie z. B. in (9.5). Wir beschränken uns auf den Fall

$$Ly := -y'' + c\,y$$

mit einer positiven Funktion $c\colon \Omega \to \mathbb{R}$, sowie auf homogene Dirichlet-Randbedingungen

$$g \equiv 0.$$

Die Gleichung (12.1) wird häufig als **Wärmeleitungsgleichung** bezeichnet. Die Lösung $y(x,t)$ ist die Temperatur des Stabes Ω an der Stelle x zum Zeitpunkt t. Die rechte Seite f ist ein Quellterm.

Um eindeutige Lösbarkeit zu garantieren müssen wir uns sowohl Randwerte, als auch eine Anfangsbedingung vorgeben.

§ 12.1 Finite-Differenzen-Diskretisierung

Grundidee: Wende die Finite-Differenzen-Methode auf ein Gitter $\overline{Q}_h = \{(x_i, t_k)\colon i = 0,\dots,N,\ k = 0,1,\dots\}$ des Orts-Zeit-Zylinders an. Die Zeit- und Ortsschrittweiten sind äquidistant angenommen, d. h. $\tau = t_k - t_{k-1}$, $k = 1,2,\dots$, und $h = x_i - x_{i-1}$, $i = 1,\dots,N$.

Approximiere nun auch die Zeitableitung durch

$$\partial_t y(x,t_k) \approx \frac{1}{\tau}(y(x,t_k) - y(x,t_{k-1})) \qquad \text{(Rückwärtsdifferenz), oder}$$

$$\partial_t y(x,t_k) \approx \frac{1}{\tau}(y(x,t_{k+1}) - y(x,t_k)) \qquad \text{(Vorwärtsdifferenz)}$$

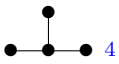
Die gesuchte Gitterfunktion $y_h\colon \overline{Q}_h \to \mathbb{R}$ kann durch die Unbekannten

$$y_i^k := y_h(x_i,t_k), \qquad i = 0,\dots,N,\ k = 0,1,\dots$$

eindeutig beschrieben werden. Der Differentialoperator bezüglich der Ortsvariablen L wird wieder wie in (11.1) diskretisiert:

$$[L_h y_h](x_i,t_k) = \frac{-y_{i-1}^k + 2y_i^k - y_{i+1}^k}{h^2}.$$

Die Lösung zum Zeitpunkt $t = 0$ ist durch die Anfangsbedingung $y(x,0) = y_0(x)$ vorgegeben. Wir wählen also $y_h(x,0) = R_h y_0$. Je nach Wahl der Approximation der Zeitableitung ergeben sich unterschiedliche Verfahren:

(a) **Das explizite Euler-Verfahren:**  [4]

Dieses erhält man bei Wahl Vorwärtsdifferenz:

$$\frac{1}{\tau}(y_i^{k+1} - y_i^k) + \frac{1}{h^2}\left(-y_{i-1}^k + 2y_i^k - y_{i+1}^k\right) + c_i^k y_i^k = f_i^k, \tag{12.2}$$

für $i = 1, \ldots, I-1$ und $k = 0, 1, \ldots$, mit $c_i^k = c(x_i, t_k)$ und $f_i^k = f(x_i, t_k)$.

Die Lösung im $k+1$-ten Zeitschritt ($k = 0, 1 \ldots$) lässt sich explizit aus (12.2) berechnen:

$$y_i^{k+1} = y_i^k + \frac{\tau}{h^2}\left(-y_{i-1}^k + 2y_i^k - y_{i+1}^k\right) - \tau c_i^k y_i^k + \tau f_i^k.$$

Dabei lassen sich y_0^k und y_N^k jeweils aus der Randbedingung $y(x,t) = g(x,t)$ für $x \in \Gamma$ bestimmen.

(b) **Das implizite Euler-Verfahren:** 

Wählt man die Rückwärtsdifferenz für die Approximation der Zeitableitung erhält man folgendes Schema:

$$y_i^k + \frac{\tau}{h^2}\left(-y_{i-1}^k + 2y_i^k - y_{i+1}^k\right) + \tau\, c_i^k\, y_i^k = y_i^{k-1} + \tau\, f_i^k$$

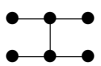
für $i = 1, \ldots, N-1$ und $k = 1, 2, \ldots$. Wir erkennen hier einen Teil der Systemmatrix aus (11.2) wieder:

$$L_h^k := \begin{pmatrix} \frac{2}{h^2} + c_1^k & -\frac{1}{h^2} & & & \\ -\frac{1}{h^2} & \frac{2}{h^2} + c_2^k & -\frac{1}{h^2} & & \\ & \ddots & \ddots & \ddots & \\ & & -\frac{1}{h^2} & \frac{2}{h^2} + c_{N-2}^k & -\frac{1}{h^2} \\ & & & -\frac{1}{h^2} & \frac{2}{h^2} 2 + c_{N-1}^k \end{pmatrix}.$$

Zusammenfassend lässt sich diese Gleichung dann schreiben als

$$(I + \tau\, L_h^k)\vec{y}^k = \vec{y}^{k-1} + \tau\, \vec{f}^k, \quad k = 1, 2, \ldots, \tag{12.3}$$

mit der Einheitsmatrix $I \in \mathbb{R}^{N-1 \times N-1}$ und $\vec{y}^k = (y_1^k\ y_2^k\ \ldots\ y_{N-1}^k)^T$ und $\vec{f}^k = (f(x_1, t_k)\ \ldots f(x_{N-1}, t_k)$. Wir müssen also in jedem Zeitschritt ein Gleichungssystem der Dimension $N-1 \times N-1$ lösen. Das entspricht dem Aufwand, den wir auch zur Lösung der zugehörigen Randwertaufgabe investieren müssen. Ist c unabhängig von t ändert sich die Systemmatrix allerdings nicht, eine ggf. berechnete Faktorisierung kann wiederverwendet werden.

(c) **Die Crank-Nicolson-Methode:** 

[4] Das Schema gibt an, welche Punkte in einer einzelnen Differenzengleichung vorkommen.

Dieses Verfahren ergibt sich, wenn man $L_h y$ und f zwischen den Zeitpunkten, also in $t_{k+1/2} = \frac{1}{2}(t_k + t_{k+1})$ auswertet. Damit ergibt sich

$$[L_h y](x_i, t_{k+1/2}) = \frac{1}{2\,h^2}\left(\left[-y_{i-1}^k + 2y_i^k - y_{i+1}^k\right] + \left[-y_{i-1}^{k+1} + 2y_i^{k+1} - y_{i+1}^{k+1}\right]\right).$$

Daraus resultiert das Schema

$$(I + \frac{\tau}{2}L_h^k)\vec{y}^k = (I - \frac{\tau}{2}L_h^{k-1})\vec{y}^{k-1} + \vec{f}^{k+1/2}, \quad k = 1, 2, \ldots.$$

§ 12.2 Zeitdiskretisierung mittels unstetigem Galerkin-Verfahren

Grundidee: Führe zunächst eine Semidiskretisierung bezüglich der Zeitvariablen durch. Anschließend erhalten wir für jeden Zeitschritt ein Randwertproblem, welches wir mit der Finite-Differenzen-Methode aus Abschnitt 11.2 lösen können.

Der Einfachheit halber betrachten wir hier lediglich das **DG(0)**-Verfahren. In Beispiel 5.5 haben wir bereits gezeigt, dass die Anwendung von **DG(0)** auf die Gleichung (12.1) auf die Randwertaufgabe

$$\begin{cases} (I + \tau\, L)y(\cdot, t_{k+1}) = y(\cdot, t_k) + \tau f(\cdot, t_{k+1}) & \text{in } \Omega \\ y(0, t_{k+1}) = y(1, t_{k+1}) = 0 \end{cases} \tag{12.4}$$

führt (Beachte: dort wurde die Zeitschrittweite mit h bezeichnet). Dieses Problem hat die Gestalt (RWP) und ist bezüglich der Ortsvariablen noch nicht diskretisiert (man spricht von einer **Semidiskretisierung**). Unter Verwendung einer Finite-Differenzen-Diskretisierung im Ort erhalten wir die voll-diskrete Formulierung

$$(I + \tau\, L_h^{k+1})\,\vec{y}^{k+1} = \vec{y}^k + \tau\,\vec{f}^{k+1}, \quad k = 0, 1, \ldots.$$

Bemerkung 12.1.

(a) Auch hier erkennen wir wieder, dass das diskrete Schema mit dem des impliziten Euler-Verfahren (12.3) übereinstimmt.

(b) Zur Lösung des semi-diskreten Problems (12.4) kann auch eine andere Methode fur Ortsdiskretisierung verwendet werden, z. B. Schießmethoden, finite Elemente, ...

§ 12.3 Die vertikale Linienmethode

Darüber hinaus gibt es noch weitere Verfahren zur Lösung von Anfangs-Randwert-Problemen, die in der Praxis häufiger eingesetzt werden. Beispielsweise lässt sich das ARWP (12.1) in ein Variationsproblem

$$(\partial_t y + Ly, v) = (f, v) \qquad \forall v \in C^1(\Omega) \cap C(\overline{\Omega}),$$

überführen. Hier ist $(u, v) = \int_\Omega u\, v$ das $L^2(\Omega)$-Skalarprodukt. Analog zu den unstetigen Galerkin-Verfahren[5] verwendet man nun einen Galerkin-Ansatz um eine Ortsdiskretisierung zu realisieren, d. h. man approximiert Ansatz- und Testraum durch

[5] Diese Verfahren werden erst bei partiellen Differentialgleichungen ($n \geq 2$) richtig Interessant. Diese Verfahren werden in der Vorlesung "Numerik partieller Differentialgleichungen" intensiver diskutiert.

einen endlich-dimensionalen Funktionenraum $V_h := \operatorname{span}\{\varphi_i\}_{i=1}^N$ mit ortsabhängigen Basisfunktionen $\varphi_i = \varphi_i(x)$, und sucht eine Lösung der Gestalt

$$y_h(x,t) = \sum_{i=1}^{N} y_i(t)\varphi_i(x). \tag{12.5}$$

Dies führt auf die Gleichung

$$\sum_{j=1}^{N} \dot{y}_j \,(\varphi_j, \varphi_i) + \sum_{j=1}^{N} y_j \,(L\varphi_j, \varphi_i) = (f, \varphi_j), \quad j = 1, \ldots, N. \tag{12.6}$$

Die Skalarprodukte

$$m_{ij} = (\varphi_j, \varphi_i), \quad a_{ij} = (L\varphi_j, \varphi_i), \quad f_i = (f, \varphi_i)$$

lassen sich (ggf. unter Verwendung von Quadraturformeln) berechnen. Dann lässt sich (12.6) kompakt schreiben als

$$M\dot{\vec{y}} + A\vec{y} = \vec{f} \tag{12.7}$$

mit dem unbekannten Lösungsvektor $\vec{y}(t) = (y_1(t) \ \ldots y_N(t))^\top$, welcher in (12.5) eingesetzt werden muss.

Beachte: (12.7) ist ein Anfangswertproblem mit einer gewöhnlichen Differentialgleichung. Hierfür kann ein beliebiges RKV zur Lösung verwendet werden.

Diese hier beschriebene Methode wird als **vertikale Linienmethode** bezeichnet. Die Bezeichnung beruht auf der Tatsache, dass nach der Semidiskretisierung die Unbekannten y_i noch kontinuierliche Funktionen in der Zeit sind.

Bemerkung 12.2. Analog dazu gibt es auch die **horizontale Linienmethode**, auch **Rothe-Methode** genannt. Diese werden wir aber nicht weiter diskutieren.