Kapitel 3

Numerik gewöhnlicher Differentialgleichungen

Inhalt dieses Kapitels ist die numerische Lösung eines Systems gewöhnlicher Differentialgleichungen

$$y'(x) = f(x, y(x)) \tag{3.1}$$

bzw. ausgeschrieben in Komponenten

$$\begin{aligned} y_1'(x) &= f_1(x, y_1(x), \ldots, y_n(x)), \\ y_2'(x) &= f_2(x, y_1(x), \ldots, y_n(x)), \\ &\vdots \quad\quad \vdots \\ y_n'(x) &= f_n(x, y_1(x), \ldots, y_n(x)). \end{aligned}$$

Man unterscheidet dabei zwischen *Anfangswertproblemen* und *Randwertproblemen*. Bei ersteren fordert man

$$y(x_0) = y_0$$

an einer Stelle x_0, bei letzteren dagegen

$$r(y(a), y(b)) = 0$$

mit einer vorgegebenen Funktion r und einem Intervall $[a, b]$.

3.1 Beispiele und Grundlagen

Bevor wir die Problemklasse näher analysieren, wollen wir zwei Beispiele kennenlernen. Als erstes wird die chemische Reaktionskinetik behandelt, danach folgen elektrische Schaltkreise. In beiden Beispielen verwendet man *mathematische Modelle*, um einen realen Vorgang zu beschreiben. Ohne Vereinfachungen und Modellannahmen ist dies im Allgemeinen nicht möglich, doch wird man von einem vernünftigen Modell verlangen, dass es die interessierenden Phänomene bzw. Effekte hinreichend gut widerspiegelt.

Chemische Reaktionskinetik

Die Reaktionskinetik beschreibt den zeitlichen Ablauf chemischer Reaktionen, und zwar auf einer Makroebene. Man interessiert sich nicht für das Verhalten einzelner Moleküle, sondern für Stoffkonzentrationen in einem Reaktionsgefäß und nimmt an, dass sie dem *Massenwirkungsgesetz* (bzw. Vereinfachungen daraus) unterliegen.

Betrachten wir dazu zwei Gase A und B unter den Annahmen konstanter Druck, konstantes Volumen und konstante Temperatur. Im Fall einer *monomolekularen Reaktion*

$$A \xrightarrow{k} B$$

mit Reaktionsgeschwindigkeit k gilt für die zeitliche Änderung der Konzentrationen $[A]$ und $[B]$

$$\frac{d}{dt}[A] = [\dot{A}] = -k[A], \qquad [\dot{B}] = k[A]$$

Die Exponentialfunktion als Lösung beschreibt also den zeitlichen Verlauf der Reaktion. Wegen $[\dot{A}] + [\dot{B}] = 0$ folgt $[A] + [B] =$ konstant, was man als Massenerhaltung bezeichnet.

Komplexer und in der Anwendung bedeutsamer sind *bimolekulare Reaktio-*

nen. Das Reaktionsschema lautet hier

$$A + B \xrightarrow{k} C + D$$

mit Substanzen A, B, C, D. In diesem Fall ergeben sich die Differentialgleichungen für die Konzentrationen zu

$$\begin{aligned} [\dot{A}] &= -k[A][B], \qquad [\dot{B}] = -k[A][B], \\ [\dot{C}] &= k[A][B], \qquad [\dot{D}] = k[A][B]. \end{aligned} \tag{3.2}$$

Im allgemeinen Fall liegen insgesamt n Substanzen sowie m Reaktionen zwischen ihnen vor,

$$\sum_{i=1}^{n} \alpha_{ij} A_i \xrightarrow{k_j} \sum_{i=1}^{n} \gamma_{ij} A_i, \quad j = 1, \ldots, m.$$

Dabei bezeichnen die ganzen positiven Zahlen α_{ij}, γ_{ij} die Anteile der Substanzen (stöchiometrische Konstanten) und die reellen Parameter k_j die unterschiedlichen Reaktionsgeschwindigkeiten. Als Differentialgleichung für die Konzentration der Substanz A_i hat man dann

$$[\dot{A}_i] = \sum_{j=1}^{m} (\gamma_{ij} - \alpha_{ij}) k_j \prod_{l=1}^{n} [A_l]^{\alpha_{lj}}, \quad i = 1, \ldots, n. \tag{3.3}$$

Die Vorschrift (3.3) stellt ein allgemeines Schema dar, um Differentialgleichungsmodelle für chemische Reaktionen zu erzeugen. Charakteristisch ist dabei die *polynomiale rechte Seite.*

Beispiel 3.1: *Brusselator* (Lefever/Nicolis 1971)
Gegeben sind $n = 6$ Substanzen A, B, D, E, X, Y und die $m = 4$ Übergänge

$$\begin{aligned} A &\xrightarrow{k_1} X, & B + X &\xrightarrow{k_2} Y + D, \\ 2X + Y &\xrightarrow{k_3} 3X, & X &\xrightarrow{k_4} E \end{aligned}$$

mit Reaktionsgeschwindigkeiten $k_1, \ldots, k_4$. Anwendung der Regeln für mono- und bimolekulare Reaktionen ergibt (ohne $[\cdot]$ Notation)

$$\begin{aligned} \dot{A} &= -k_1 A, & \dot{X} &= k_1 A, \\ \dot{B} &= -k_2 BX, & \dot{X} &= -k_2 BX, \\ \dot{Y} &= k_2 BX, & \dot{D} &= k_2 BX, \\ \dot{X} &= -k_4 X, & \dot{E} &= k_4 X. \end{aligned}$$

Die dritte Reaktion ist autokatalytisch trimolekular und liefert nach der allgemeinen Regel (3.3) die Beiträge

$$\dot{X} = k_3 X^2 Y, \qquad \dot{Y} = -k_3 X^2 Y.$$

Mit Sortieren nach Unbekannten und *Aufsummieren der rechten Seiten* folgen die gekoppelten Differentialgleichungen

$$\begin{aligned} \dot{A} &= -k_1 A, \\ \dot{B} &= -k_2 B X, \\ \dot{D} &= k_2 B X, \\ \dot{E} &= k_4 X, \\ \dot{X} &= k_1 A - k_2 B X + k_3 X^2 Y - k_4 X, \\ \dot{Y} &= k_2 B X - k_3 X^2 Y. \end{aligned} \tag{3.4}$$

Fazit: Ein nichtlineares Differentialgleichungssystem mit polynomialer rechter Seite!
Analytische Lösung?

Als Vorbereitung für später werden die Gleichungen (3.4) vereinfacht: Die Differentialgleichungen für D und E werden weggelassen (warum?), A und B werden als konstant angenommen. Wir setzen $k_i = 1$, schreiben x statt t sowie $y_1(x) := X(x)$, $y_2(x) := Y(x)$.
(**Notationsvereinbarung:** $y(t)$ mit Zeit t : $\frac{d}{dt} y = \dot{y}$, $y(x)$ mit unabh. Var. x : $\frac{d}{dx} y = y'$)

Übrig bleibt das sogenannte Brusselatormodell (woher stammt der Name?)

$$\begin{aligned} y_1' &= A + y_1^2 y_2 - (B+1) y_1, \\ y_2' &= B y_1 - y_1^2 y_2, \end{aligned} \tag{3.5}$$

das eine hochinteressante nichtlineare Dynamik aufweist.

Elektrische Schaltkreise

Die Beschreibung einer Schaltung als Netzwerk elektrischer Bauelemente bildet die Grundlage für Modellbildung und numerische Simulation hochintegrierter Systeme (Speicherchips und Prozessoren). Auf diese Art und Weise kann das transiente Verhalten der Ausgangssignale als Reaktion auf die Eingangssignale ohne zeit- und konstenintensive experimentelle Untersuchungen analysiert werden.

Wie bei der Reaktionskinetik existiert auch hier ein auf physikalischen Gesetzmäßigkeiten basierender Kalkül, mit dessen Hilfe die Netzwerkgleichun-

gen aufgestellt werden. Wesentlich dabei sind die *Kirchhoffschen Regeln*

$$\textit{Die Summe der} \left\{ \begin{array}{c} \textit{Teilströme in jedem Knoten} \\ \textit{Teilspannungen in jeder Masche} \end{array} \right\} \textit{ist Null.}$$

Zusammen mit den Gleichungen für die Bauelemente (Widerstände, Kapazitäten, Induktivitäten, Transistoren, Strom- und Spannungsquellen) ergeben sich dann die Netzwerkgleichungen, die den gesamten Schaltkreis beschreiben.

Die am weitesten verbreitete Technik, die modifizierte Knotenspannungsanalyse (MNA, Modified Nodal Analysis), führt auf Differentialgleichungen der Form

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

wobei die Unbekannten y aus Spannungen und Strömen zusammengesetzt sind. Die Einträge in der Matrix C bestehen aus Kapazitäten. Falls keine solchen auftreten (und keine Induktivitäten), ist $C \equiv 0$, und die Gleichungen reduzieren sich auf den stationären Fall.

In vielen Anwendungen ist die Matrix C singulär, und es liegt somit keine gewöhnliche Differentialgleichung vor, sondern ein sogenanntes differentiell-algebraisches System. Einzelheiten dazu in Kapitel 6. Das folgende Beispiel dagegen enthält neben einer Kapazität noch eine Induktivität und wird durch eine Differentialgleichung 2. Ordnung beschrieben.

Beispiel 3.2: *Elektrischer Schwingkreis*
Wir betrachten als einfaches Beispiel einen elektrischen Schwingkreis, der aus einer Spannungsquelle $U(t)$, einer Kapazität C, einer Induktivität L und einem Widerstand R besteht, siehe Abb. 1.

Für die Spannungsabfälle U_R, U_C sowie U_L gelten die Gesetze

$$U_R = RI, \qquad U_C = Q/C, \qquad U_L = L\dot{I}$$

wobei I den fließenden Strom und Q die Ladung der Kapazität bezeichnen. Die Kirchhoffschen Regeln (hier die Maschenregel) ergeben

$$U = U_R + U_C + U_L.$$

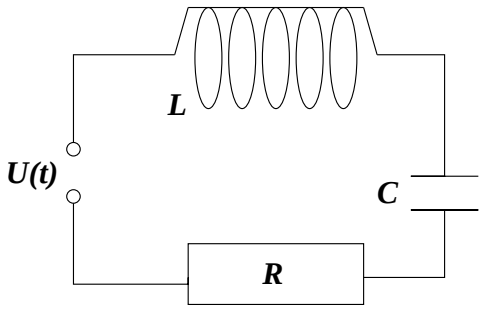


Abbildung 1: Elektrischer Schwingkreis

Mit $I = \dot{Q}$ und Einsetzen erhält man für die an der Kapazität anliegende Spannung U_C die Differentialgleichung

$$LC\,\ddot{U}_C + RC\,\dot{U}_C + U_C = U. \tag{3.6}$$

Insgesamt also eine über $U(t)$ gesteuerte Schwingung, die durch das Produkt RC gedämpft wird.

Aussagen aus der Theorie gewöhnlicher Differentialgleichungen

Als Einstieg und Wiederholung beginnen wir mit einem Blick auf lineare Differentialgleichungen. Für das skalare lineare Anfangswertproblem (AWP)

$$y' = a \cdot y + b, \quad y(x_0) = y_0,$$

liefert der Lösungsansatz $z(x) = e^{a(x-x_0)} \cdot z_0$ eine Lösung des homogenen Falls $z' = az$. Durch *Variation der Konstanten* folgt

$$y(x) = z(x)v(x) \quad \Longrightarrow \quad y(x) = \left(\frac{b}{a} + y_0\right) e^{a(x-x_0)} - \frac{b}{a}.$$

Dieses Vorgehen verallgemeinert man schnell auf Systeme linearer Differentialgleichungen

$$y' = A \cdot y + b, \quad A \in \mathbb{R}^{n\times n},\, b \in \mathbb{R}^n.$$

Wir nehmen an, dass A diagonalisierbar ist (d.h. n l.u. EV besitzt). Mit $z(x) = \exp(A(x - x_0))z_0$ hat man eine Lösung des homogenen Systems

$z' = Az$. Dabei stellen die Spalten der Matrix

$$\exp(A\tau) = \sum_{k=0}^{\infty} \frac{(A\tau)^k}{k!} \quad \textit{Matrizenexponentielle} \tag{3.7}$$

ein Fundamentalsystem dar. Die Variation der Konstanten für den inhomogenen Fall ergibt sich hier zu $y(x) = \exp(A(x - x_0)) \cdot v(x)$.

Im Fall nichtlinearer rechter Seiten (bzw. Differentialgleichungen) kommt man mit analytischen Lösungstechniken dagegen meist nicht mehr weiter.

Beispiel 3.3: *Van–der–Pol–Gleichung*
Wir betrachten zunächst die lineare Differentialgleichung 2. Ordnung $\ddot{z}(t) + \alpha\dot{z}(t) + z(t) = 0$, vergleiche den Schwingkreis (3.6). Für $\alpha > 0$ hat sie fallende (gedämpfte/stabile) Lösungen, für $\alpha > 0$ wachsende (instabile) Lösungen, und für $\alpha = 0$ resultieren periodische Lösungen. Setzt man $\alpha = \alpha(z)$, so dass $\alpha < 0$ für kleine z und $\alpha > 0$ für große z, so kann man ein sich selbst regulierendes (steuerndes) Verhalten erwarten, das in eine periodische Lösung hineinläuft. Die Wahl $\alpha(z) := \mu(z^2 - 1)$ mit konstantem Parameter $\mu > 0$ erfüllt diese Erwartung und liefert die Van-der-Pol-Gleichung

$$\ddot{z} + \mu(z^2 - 1)\dot{z} + z = 0. \tag{3.8}$$

Eine allgemeine geschlossene Lösung ist nicht bekannt!

Es ist vorteilhaft, die Gleichung (3.8) auf die neue Zeit $x = t/\mu$ zu transformieren (die Periode hängt dann nicht mehr von μ ab). Mit $u(x) = z(\mu x) = z(t(x))$, $u' = \mu\dot{z}$, folgt

$$\frac{1}{\mu^2}u'' + (u^2 - 1)u' + u = 0.$$

Von besonderem Interesse ist der Fall $\mu \gg 1$, bei dem wir $\varepsilon := 1/\mu^2 \ll 1$ setzen (ein sogenanntes *singulär gestörtes System*). In einem letzten Schritt führen wir noch die Koordinaten nach Liénhard ein,

$$y_2(x) := u(x), \quad y_1(x) := \varepsilon y_2' + (y_2^3/3 - y_2),$$

und erhalten so das System

$$\begin{aligned} y_1' &= -y_2, \\ \varepsilon y_2' &= y_1 - y_2^3/3 + y_2. \end{aligned} \tag{3.9}$$

Wie verlaufen die Lösungen von (3.9) im oben skizzierten Richtungsfeld?

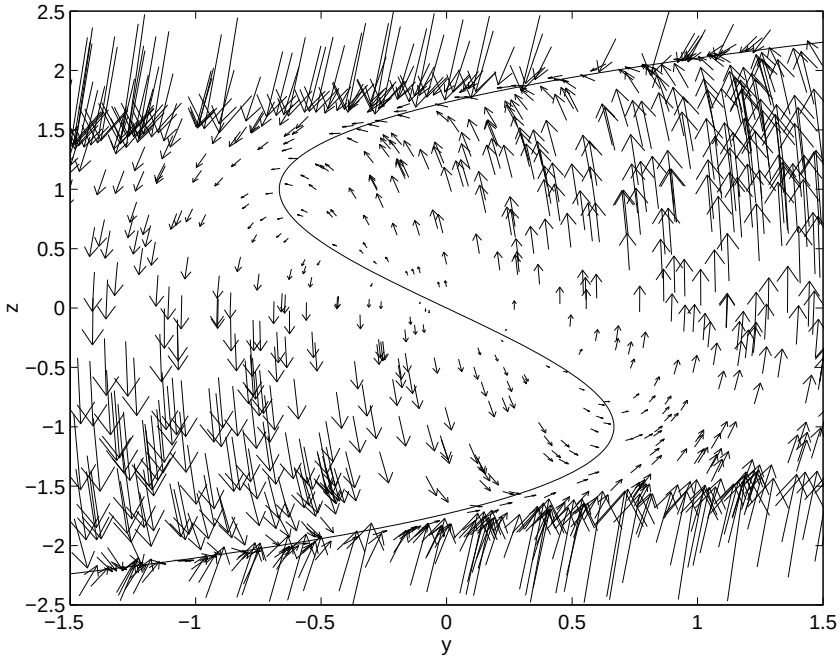


Abbildung 2: Richtungsfeld der Van-der-Pol-Gleichung

Bevor wir zum Existenz- und Eindeutigkeitssatz kommen, seien noch drei oft hilfreiche Bemerkungen erwähnt:

(i) Differentialgleichungen höherer Ordnung lassen sich durch *Hilfsvariablen* auf Differentialgleichungen erster Ordnung zurückzuführen:

$$y'' = -\omega^2 y \quad \Leftrightarrow \quad y' = v,\ v' = -\omega^2 y.$$

(ii) Nichtautonome Gleichungen $y' = f(x, y(x)$ transformiert man mittels $y_{n+1}(x) := x$ und $f_{n+1}(x, y) := 1$ in ein *autonomes System.*

(iii) Durch Integration überführt man $y' = f(x, y)$ in

$$y(x) = y_0 + \int_{x_0}^{x} f(t, y(t))\ dt\ .$$

Falls $f = f(t)$, ist die Quadratur als Sonderfall enthalten!

Wir betrachten nun das allgemeine Anfangswertproblem

$$y'(x) = f(x, y(x)), \quad y(x_0) = y_0, \tag{3.10}$$

wobei $f : [a, b] \times \mathbb{R}^n \to \mathbb{R}^n$ und $x_0 \in [a, b]$, $y_0 \in \mathbb{R}^n$. Die rechte Seite f heißt (global) *Lipschitz–stetig* im Definitionsgebiet (oder "Streifen")

$$D := \{(x, y) : a \leq x \leq b,\ y \in \mathbb{R}^n\},$$

falls

$$\|f(x, y) - f(x, z)\| \leq L \cdot \|y - z\| \quad \forall\, (x, y), (x, z) \in D. \tag{3.11}$$

Einfaches Kriterium: Falls f sowie $\partial f/\partial y$ stetig auf dem Definitionsgebiet D sind und $\partial f/\partial y$ beschränkt ist, $\|\partial f/\partial y\| \leq L$, folgt daraus die Lipschitzstetigkeit.

Beweisskizze: Für je zwei Punkte $(x, y), (x, z)$ in D gilt (MWS der Differentialrechnung)

$$\|f(x, y) - f(x, z)\| \leq sup_{(t,v)\in D} \|\partial f(t, v)/\partial v\|\, \|y - z\|.$$

Mit dem Begriff der Lipschitzstetigkeit zeigt man dann (Picard-Lindelöf, s. Analysis)

Satz 3.1 *Falls die rechte Seite f auf dem Definitionsgebiet D stetig ist und der Lipschitzbedingung (3.11) genügt, besitzt das Anfangswertproblem (3.10) eine eindeutige, stetig differenzierbare Lösung y auf dem ganzen Intervall $[a, b]$.*

Eine höhere Glattheit der Lösung hängt von der Glattheit der rechten Seite f bzw. ihrer Ableitungen $\partial^p f/\partial y^p$ ab. Falls alle partiellen Ableitungen bis zur Ordnung p stetig sind, ist auch y insgesamt p mal stetig differenzierbar.

Lässt man dagegen die Lipschitzstetigkeit als Voraussetzung weg und geht nur von einer stetigen rechten Seite f aus, kann man nur noch die Existenz einer Lösung nachweisen, die Eindeutigkeit geht verloren (Existenzsatz von Peano).

Hinreichend für Existenz und Eindeutigkeit ist im übrigen bereits eine abgeschwächte Form der Lipschitz-Bedingung (3.11), die *lokale Lipschitz-Stetigkeit*. Sie ist gegeben, falls es zu jeder kompakten Teilmenge $K \subset D$ eine Lipschitzkonstante L_K gibt mit

$$\|f(x,y) - f(x,z)\| \leq L_K \cdot \|y - z\| \quad \forall\, (x,y),(x,z) \in K. \tag{3.12}$$

Allerdings kann mit dieser schwächeren Voraussetzung nicht die Existenz auf dem ganzen Intervall $[a,b]$ garantiert werden.

3.2 Einfluß von Störungen

In diesem Abschnitt steht die Kondition des Anfangswertproblems im Mittelpunkt. Wie wirken sich Störungen in den Anfangswerten und in der rechten Seite auf die Lösung aus?

Satz 3.2 *Störung der Anfangswerte*
Falls die rechte Seite f der Lipschitzbedingung (3.11) genügt, gilt für zwei Lösungen $y(x)$ und $z(x)$ mit Anfangswerten $y(x_0)$, $z(x_0)$ die Abschätzung

$$\|y(x) - z(x)\| \leq \|y(x_0) - z(x_0)\| e^{L|x-x_0|}.$$

Beweis: Aus $y' = f(x,y)$ und $z' = f(x,z)$ folgt mit Integration

$$y(x) - z(x) = y(x_0) - z(x_0) + \int_{x_0}^{x} (f(t,y) - f(t,z))\ dt$$

und damit

$$\underbrace{\|y(x) - z(x)\|}_{=:m(x)} \leq \|y(x_0) - z(x_0)\| + L \int_{x_0}^{x} \underbrace{\|y(t) - z(t)\|}_{=m(t)}\ dt$$

bzw.

$$m(x) \leq \|y(x_0) - z(x_0)\| + L \int_{x_0}^{x} m(t)\ dt. \qquad (*)$$

$m(x)$ ist stetig und

$$q(x) := e^{-Lx} \int_{x_0}^{x} m(t)\ dt$$

stetig differenzierbar. Also gilt

$$m(x) = \left(e^{Lx} q(x)\right)' = Le^{Lx} q(x) + e^{Lx} q'(x). \qquad (**)$$

Eingesetzt in $(*)$:

$$\begin{aligned} Le^{Lx} q(x) + e^{Lx} q'(x) &\leq \|y_0 - z_0\| + Le^{Lx} q(x) \\ \Longrightarrow \quad q'(x) &\leq \|y_0 - z_0\| e^{-Lx} \\ \Longrightarrow q(x) = \int_{x_0}^{x} q'(t)\ dt &\leq \|y_0 - z_0\| \left(e^{-Lx_0} - e^{-Lx}\right)/L \end{aligned}$$

Abschätzungen eingesetzt in $(**)$:

$$m(x) \leq \|y_0 - z_0\|(e^{L(x-x_0)} - 1 + 1)$$

(Bei Integration in die "Vergangenheit" mit $x_0 \geq x$ analoges Vorgehen). □

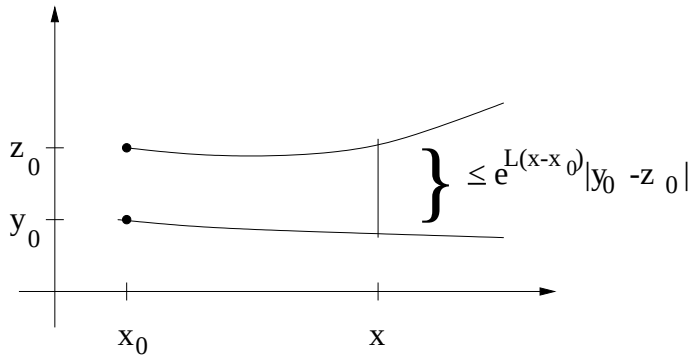


Abbildung 3: Skizze zu Satz 3.2

Warum können sich zwei Lösungskurven nicht schneiden?

Im Beweis von Satz 3.2 ist eine vereinfachte Fassung des Lemmas von Gronwall versteckt, das eine wichtige Rolle bei verschiedenen Abschätzungen spielt. In der vollständigen Fassung lautet es:

Lemma 3.3 *Gronwall*
Sei $m(x)$ eine positive, stetige Funktion sowie $\rho \geq 0$, $\varepsilon \geq 0$. Falls

$$m(x) \leq \rho + \varepsilon(x - x_0) + L \int_{x_0}^{x} m(t)\, dt,$$

gilt die Abschätzung

$$m(x) \leq \rho e^{L(x-x_0)} + \frac{\varepsilon}{L}\left(e^{L(x-x_0)} - 1\right).$$

Mit dem Lemma von Gronwall kann man die Abschätzung über die Auswirkung von Störungen in den Anfangswerten erweitern. Sei y eine exakte Lösung und z eine *approximative Lösung*, die einen *Defekt* $\delta(x)$ aufweist,

$$z'(x) = f(x, z(x)) + \delta(x), \quad \|\delta(x)\| \leq \varepsilon.$$

Mit $m(x) := \|y(x) - z(x)\|$ und $\rho := \|y(x_0) - z(x_0)\|$ sind die obigen Voraussetzungen erfüllt, denn

$$y(x) - z(x) = y(x_0) - z(x_0) + \int_{x_0}^{x} \delta(t)\, dt + \int_{x_0}^{x} \left(f(t, y) - f(t, z)\right)\, dt.$$

Anwendung des Gronwallschen Lemmas liefert

$$\|y(x) - z(x)\| \leq \|y(x_0) - z(x_0)\| e^{L(x-x_0)} + \frac{\varepsilon}{L}\left(e^{L(x-x_0)} - 1\right) \tag{3.13}$$

als Verallgemeinerung von Satz 3.2. Der zweite Term beinhaltet dabei die Auswirkung des Defekts / der Störung $\delta(x)$. Auffallend ist die zentrale Rolle der Exponentialfunktion. Falls die Lipschitzkonstante L sehr groß ist, wird e^{Lx} sehr schnell wachsen, und (3.13) stellt dann eine sehr pessimistische Aussage dar. In den allermeisten Fällen sind jedoch Anfangswertprobleme deutlich besser konditioniert als es (3.13) suggeriert.

3.3 Einschrittverfahren

Dieser Abschnitt stellt eine erste und sehr wichtige Verfahrensklasse zur Lösung von Anfangswertproblemen vor, die sogenannten *Einschrittverfahren*. Allgemein basieren numerische Verfahren für AWPe auf folgendem Ansatz: Man führt ein *Gitter* auf dem betrachteten Intervall $[a, b]$ ein,

$$a = x_0 < x_1 < x_2 < \ldots < x_{n-1} < x_n = b,$$

d.h. man *diskretisiert das Kontinuum* $[a, b]$. Die Schrittweiten $h_i = x_{i+1} - x_i$ müssen nicht äquidistant sein. Dann berechnet man, ausgehend von y_0, sukzessive Nährungen $y_i \doteq y(x_i)$ an den diskreten Punkten,

$$y_0 \longrightarrow y_1 \longrightarrow y_2 \longrightarrow \ldots \longrightarrow y_n.$$

Man spricht dabei allgemein von *Diskretisierungsverfahren* und unterscheidet speziell:

a) **Einschrittverfahren:** Nur die Daten x_i, y_i, h_i gehen in die Berechnung von y_{i+1} ein.

b) **Mehrschrittverfahren:** Hier gehen Daten x_k, y_k, h_k für $k = i-m, \ldots, i$ aus der "Vergangenheit" in die Berechnung von y_{i+1} ein. Siehe Kap. 4.

Beispiele für Einschrittverfahren

Zunächst betrachten wir einige elementare Beispiele (mit $h_i = h$ konstant), um einen ersten Eindruck von Einschrittverfahren zu bekommen. Ältestes und in manchen Anwendungen immer noch gebräuchliches Verfahren ist das *explizite Eulerverfahren*

$$\begin{aligned} y_1 &= y_0 + hf(x_0, y_0), \\ y_2 &= y_1 + hf(x_1, y_1), \\ &\vdots \quad\quad \vdots \\ y_{i+1} &= y_i + hf(x_i, y_i). \end{aligned} \tag{3.14}$$

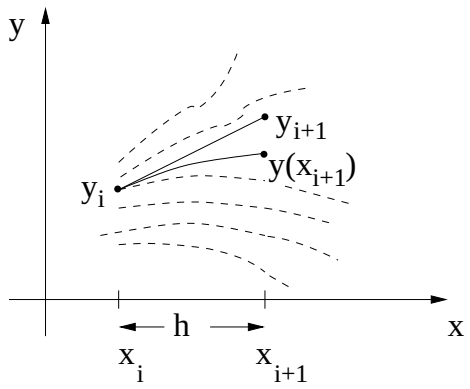


Abbildung 4: Expliziter Euler (Eulerscher Polygonzug)

Die wesentliche Idee besteht in der wiederholten Anwendung der Vorschrift, durch die man eine Folge von diskreten Approximationen erzeugt. Man spricht auch vom Eulerschen Polygonzugverfahren, da die diskreten Werte der y_i aus einem immer länger werdenden Polygonzug folgen. Damit liegt zusätzlich ein linearer Interpolant vor, und man hat de facto auch eine kontinuierliche Approximation konstruiert.

Drei Interpretationen des Eulerverfahrens ergeben sich direkt:

(i) Geometrisch: Lege die Tangente in (x_i, y_i) an.

(ii) Vorwärts-Differenzenquotient $\quad (y_{i+1} - y_i)/h \doteq y'(x_i) = f(x_i, y_i)$

(iii) Quadraturformel $\quad y(x_{i+1}) = y(x_i) + \underbrace{\int_{x_i}^{x_{i+1}} f(t, y(t))\, dt}_{\doteq h \cdot f(x_i, y_i)}$

Als zweites Einschrittverfahren betrachten wir das *Implizite Eulerverfahren*

$$y_{i+1} = y_i + h(f(x_{i+1}, y_{i+1}), \tag{3.15}$$

das sich als Rückwärts-Differenzenquotient interpretieren lässt. Während beim expliziten Euler nur eine Auswertung der rechten Seite f nötig ist,

um einen Integrationsschritt durchzuführen, erfordert der implizite Euler die Lösung eines nichtlinearen Systems für y_{i+1}!

Letztes Beispiel ist die *Trapezregel*

$$y_{i+1} = y_i + \underbrace{\frac{h}{2}\left(f(x_i, y_i) + f(x_{i+1}, y_{i+1})\right)}_{\doteq \int_{x_i}^{x_{i+1}} f(t,y(t))\ dt}, \tag{3.16}$$

bei der man deutlich die Verwandschaft zur numerischen Quadratur sieht.

Im Folgenden beschränken wir uns auf explizite Einschrittverfahren; die impliziten spielen hauptsächlich bei sogenannten *steifen Differentialgleichungen* eine Rolle.

Konsistenz und Konvergenz

Als Notation für Einschrittverfahren (ESV) verwendet man das Schema

$$y_{i+1} = y_i + h_i \Phi(x_i, y_i, h_i) \tag{3.17}$$

mit der *Verfahrensfunktion* oder *Inkrementfunktion* Φ. Beim expliziten Euler (3.14) ist $\Phi(x_i, y_i, h_i) = f(x_i, y_i)$.

Wie gut approximiert y_{i+1} die exakte Lösung $y(x_{i+1})$? Zur Analyse der Verfahren führt man verschiedene Begriffe ein. Der erste ist eine lokale Eigenschaft:

Definiton 3.1 *Lokaler Diskretisierungsfehler*
Sei $y(x)$ exakte Lösung des Anfangswertproblems $y' = f(x, y), y(x_0) = y_0$, und $y_1 = y_0 + h\Phi(x_0, y_0, h)$ die numerische Approximation nach einem Schritt. Der lokale Diskretisierungsfehler ist definiert als

$$\tau(h) := \frac{y(x_0 + h) - y_1}{h}\ . \tag{3.18}$$

Nach (3.18) gibt $\tau(h)$ die Differenz von exakter Lösung und numerischer Approximation, skaliert mit h, an. Eine zweite Interpretation ist

$$\tau(h) = \underbrace{\frac{y(x_0+h)-y_0}{h}}_{\substack{\text{Sehnensteigung}\\ \text{exakt}}} - \underbrace{\frac{y_1-y_0}{h}}_{\substack{\text{Steigung}\\ \text{Approx.}}} .$$

Schließlich gibt es eine dritte Interpretation

$$\tau(h) = \frac{y(x_0+h)-y_0}{h} - \Phi(x_0, y_0, h) \ ,$$

die $\tau(h)$ als *Defekt* sieht, der beim Einsetzen der exakten Lösung in die Verfahrensvorschrift entsteht. Formal schreibt man oft auch $\tau(x, y, h)$, um die Abhängigkeit von x sowie y stärker zu betonen.

Beispiel 3.4: Lokaler Diskretisierungsfehler beim expliziten Euler

$$\begin{array}{rrl} & \tau(h) & = \frac{1}{h}(y(x_0+h)-y_1) = \frac{1}{h}(y(x_0+h)-y_0-hf(x_0,y_0)) \\ \text{Taylorentw.} & y(x_0+h) & = y(x_0)+hy'(x_0)+\frac{1}{2}h^2y''(x_0)+\ldots \\ & & = y_0+hf(x_0,y_0)+\frac{1}{2}h^2(f_x+f_yf)(x_0,y_0)+\ldots \\ \Longrightarrow & \tau(h) & = h\cdot\frac{1}{2}(f_x+f_yf)(x_0,y_0)+\mathcal{O}(h^2) \end{array}$$

Man spricht beim Euler von einem *Verfahren der Konsistenzordnung 1.*

Nach dem lokalen Diskretisierungsfehler führen wir einen weiteren Begriff ein, die *Konsistenz*:

Definition 3.2 *Konsistenzordnung*
Ein Verfahren heißt konsistent, falls der lokale Diskretisierungsfehler für $h \to 0$ ebenfalls gegen 0 strebt:

$$\|\tau(h)\| \leq \gamma(h) \ \textit{mit} \ \lim_{h\to 0}\gamma(h) = 0 \, .$$

Das Verfahren hat die Konsistenzordnung p, falls

$$\|\tau(h)\| = O(h^p) \ .$$

Die Konsistenzordnung beschreibt die Qualität der numerischen Approximation in einem Schritt. Eigentlich interessiert man sich aber für die Qualität nach n Schritten. Dazu definiert man drittens

Definition 3.3 *Globaler Diskretisierungsfehler*
Der globale Diskretisierungsfehler eines Verfahrens ist die Differenz

$$e_n(X) = y(X) - y_n \quad \textit{mit} \;\; X = x_n \;\; \textit{fest, } n \textit{ variabel.}$$

Ein Verfahren ist konvergent, falls

$$\lim_{n\to\infty} \|e_n(X)\| = 0 \; .$$

Konvergenz bedeutet also:

Für feiner werdendes Gitter strebt der globale Fehler gegen 0.

Wie hängen Konsistenz und Konvergenz zusammen? Die Analysetechnik ist in Abb. 5 skizziert.

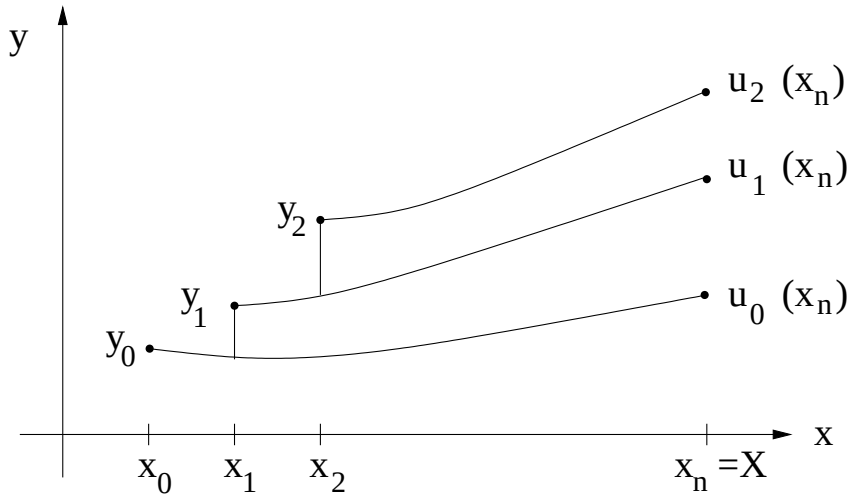


Abbildung 5: Analyse des globalen Fehlers

Zum diskreten Wert y_i sei $u_i(x)$ die exakte Lösung der Differentialgleichung mit Anfangswert $u_i(x_i) = y_i$ für $i = 0, \ldots, n$. Insbesondere ist $u_0 = y$ die exakte Lösung zum Anfangswert y_0. Der globale Fehler lautet dann

$$y(x_n) - y_n = u_0(x_n) - y_n$$

und lässt sich durch die Differenzen $u_i - u_{i+1}$ ausdrücken:

$$\begin{aligned} u_0(x_n) - y_n &= u_0(x_n) - u_1(x_n) + u_1(x_n) - u_2(x_n) + \ldots \\ &\quad \ldots - u_{n-1}(x_n) + u_{n-1}(x_n) - y_n \\ &= \sum_{i=0}^{n-2} [u_i(x_n) - u_{i+1}(x_n)] + u_{n-1}(x_n) - y_n. \end{aligned}$$

Damit folgt

$$\begin{aligned} \|u_0(x_n) - y_n\| &\leq \|u_{n-1}(x_n) - y_n\| + \sum_{i=0}^{n-2} \|u_i(x_n) - u_{i+1}(x_n)\| \\ &\leq \|u_{n-1}(x_n) - y_n\| + \sum_{i=0}^{n-2} \|u_i(x_{i+1}) - u_{i+1}(x_{i+1})\| e^{L|x_n - x_{i+1}|}. \end{aligned}$$

Bei der letzten Abschätzung wurde Satz 3.2 angewandt, um die Differenz $\|u_i(x_n) - u_{i+1}(x_n)\|$ auf die Differenz der Anfangswerte $\|u_i(x_{i+1}) - u_{i+1}(x_{i+1})\|$ zurückzuspielen. Wegen $y_{i+1} = u_{i+1}(x_{i+1})$ hat man nun

$$\|u_0(x_n) - y_n\| \leq \sum_{i=0}^{n-1} \|u_i(x_{i+1}) - y_{i+1}\| e^{L|x_n - x_{i+1}|}.$$

Die $\|\cdot\|$–Beiträge auf der rechten Seite sind lokale Fehler nach einem Schritt. Für ein konsistentes Verfahren ist demnach

$$\|u_i(x_{i+1}) - y_{i+1}\| \leq c \cdot h_i^{p+1} \leq c \cdot h_{\max}^p \cdot h_i$$

mit $h_{max} = \max_i h_i$. Schließlich bekommen wir für den globalen Fehler

$$\begin{aligned} \|u_0(x_n) - y_n\| &\leq c \cdot h_{\max}^p \sum_{i=0}^{n-1} h_i e^{L|x_n - x_{i+1}|} \\ &\leq c \cdot h_{\max}^p \int_{x_0}^{x_n} e^{L(x_n - t)} \, dt \end{aligned}$$

unter der Annahme $x_n > x_0$. Mit Ausintegrieren lautet das Resultat:

Satz 3.4 *Konvergenz von Einschrittverfahren*
Sei f stetig sowie Lipschitzstetig mit Konstante L nach (3.11). Weiter habe das Einschrittverfahren Konsistenzordnung p, d.h.

$$\|\tau(h)\| = O(h^p) \ .$$

Dann gilt für den globalen Diskretisierungsfehler

$$\|e_n(X)\| \leq c h_{\max}^p \frac{e^{L|X-x_0|} - 1}{L}$$

wobei $h_{\max} = \max_i h_i$.

Diskussion von Satz 3.4

1) Ordnung globaler Diskretisierungsfehler = Ordnung lokaler Diskretisierungsfehler. Man sagt bei ESV auch kurz: *Konsistenz* $\Longrightarrow$ *Konvergenz*

2) Variable Schrittweiten sind zugelassen. Der Verstärkungsfaktor $\frac{1}{L}(e^{L(x-x_0)} - 1)$ ist jedoch von den Schrittweiten unabhängig.

4) Manchmal findet man die Schreibweise $e(h, x)$ statt $e_n(x)$ für den globalen Diskretisierungsfehler, um die Abhängigkeit von der Schrittweite zu betonen. Oft besitzt $e(h, x)$ eine asymptotische Entwicklung in h.

Einschrittverfahren sind demnach einfach zu analysieren: Man ermittelt den lokalen Fehler (durch Taylorreihenentwicklung) und bekommt aus der Konsistenz bereits globale Konvergenz.

3.4 Runge–Kutta–Verfahren

Runge–Kutta–Verfahren sind spezielle ESV, die in jedem Schritt die rechte Seite mehrmals "ausloten" und die daraus gewonnenen Zwischenergebnisse oder *Zuwächse / Korrekturen* linear kombinieren.

Beispiel 3.5: Heun–Verfahren

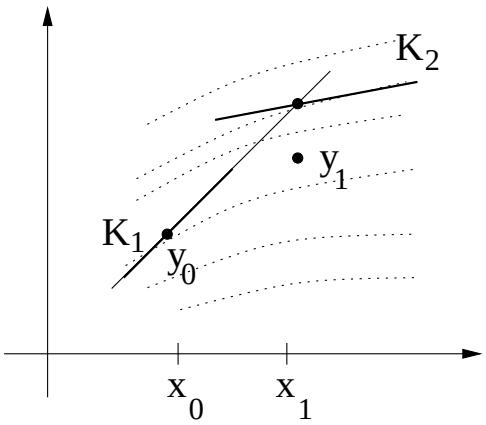


Abbildung 6: Ansatz für das Verfahren nach Heun

Diskretisierungsvorschrift

$$\begin{array}{rrl} & y_1 & = y_0 + \frac{1}{2}h(K_1 + K_2) \\ \text{mit Korrekturen} & K_1 & = f(x_0, y_0), \\ & K_2 & = f(x_1, y_0 + h \cdot K_1), \end{array}$$

bzw. mit Verfahrensfunktion Φ:

$$\begin{array}{rll} y_1 = & y_0 + h \cdot \Phi(x_0, y_0, h) \ , & \\ & \Phi(x, y, h) & = \frac{1}{2}\left[f(x, y) + f(x + h, y + hf(x, y))\right] \\ & & = \frac{1}{2}(K_1 + K_2) \end{array}$$

Das allgemeine Diskretisierungsschema für einen Schritt eines *Runge–Kutta–Verfahrens* lautet

$$y_1 = y_0 + h(b_1K_1 + b_2K_2 + \ldots + b_sK_s) \tag{3.19}$$

mit Korrekturen / Zuwächsen

$$K_i = f\left(x_0 + c_ih, y_0 + h\sum_{j=1}^{i-1} a_{ij}K_j\right), \quad i = 1, \ldots, s. \tag{3.20}$$

Die Verfahrenskoeffizienten b_i, c_i für $i = 1, \ldots, s$ sowie a_{ij} für $i = 1, \ldots, s$ und $j \leq i - 1$ legen die Methode fest, zusammen mit der *Stufenzahl* s.

Beispiel 3.6: Heun–Verfahren
Stufenzahl $s = 2$ und Koeffizienten

$$b_1 = \frac{1}{2},\ b_2 = \frac{1}{2},\ c_1 = 0,\ c_2 = 1,\ a_{21} = 1.$$

Beispiel 3.7: Modifiziertes Euler–Verfahren

$$\begin{array}{rll} & y_1 & = y_0 + h \cdot K_2 \\ \text{mit} & K_1 & = f(x_0, y_0), \\ & K_2 & = f\left(x_0 + \frac{h}{2}, y_0 + \frac{h}{2} \cdot K_1\right). \end{array}$$

Also $s = 2$ Stufen und Koeffizienten $b_1 = 0,\ b_2 = 1,\ c_1 = 0,\ c_2 = \frac{1}{2},\ a_{21} = \frac{1}{2}$.

Praktischerweise fasst man die Koeffizienten in einem Tableau, dem sogenannten *Butcher–Tableau*, zusammen:

$$\begin{array}{c|cccc} c_1 & 0 & & & \\ c_2 & a_{21} & \ddots & 0 & \\ \vdots & \vdots & \ddots & \ddots & \\ c_s & a_{s1} & \ldots & a_{ss-1} & 0 \\ \hline & b_1 & b_2 & \ldots & b_s \end{array} \quad \text{bzw.} \quad \begin{array}{c|c} c & A \\ \hline & b^T \end{array}$$

Die Verfahren sind dann gegeben über z.B. die Koeffizienten

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \quad \text{Heun} \qquad \begin{array}{c|cc} 0 & & \\ \frac{1}{2} & \frac{1}{2} & \\ \hline & 0 & 1 \end{array} \quad \text{mod. Euler}$$

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

"klassisches Runge–Kutta–Verfahren"
(Runge 1895, Kutta 1901)

Koeffizientenbestimmung

Wie kommt man auf die Koeffizienten? Die Koeffizienten sind zunächst freie Parameter des Verfahrens. Sie werden so bestimmt, dass das Verfahren "möglichst gut" ist, d.h. eine möglichst hohe Ordnung erreicht. Wesentliches Hilfsmittel bei der Bestimmung sind Taylorreihenentwicklungen der exakten Lösung und der numerischen Näherung.

Als Beispiel für eine Koeffizientenbestimmung (den sogenannten *Abgleich*) betrachte man den Fall $s = 2$:

Das Runge–Kutta–Verfahren lautet

$$\begin{aligned} y_1 = \;& y_0 + h(b_1 K_1 + b_2 K_2) \;, \\ & K_1 = f(x_0 + c_1 h, y_0) \;, \quad K_2 = f(x_0 + c_2 h, y_0 + h a_{21} K_1). \end{aligned}$$

Taylorentwicklung liefert

$$\begin{aligned} K_1 &= f(x_0, y_0) + f_x(x_0, y_0) \cdot c_1 \cdot h + O(h^2) \\ K_2 &= f(x_0, y_0) + f_x(x_0, y_0) \cdot c_2 \cdot h + f_y(x_0, y_0) h a_{21} K_1 + O(h^2) \\ &= f(x_0, y_0) + f_x(x_0, y_0) \cdot c_2 \cdot h + f_y f(x_0, y_0) h a_{21} + O(h^2) \\ \Longrightarrow \; y_1 &= y_0 \;\; + h(b_1 + b_2) f(x_0, y_0) \\ &\qquad + h^2 (b_1 c_1 + b_2 c_2) f_x(x_0, y_0) \\ &\qquad + h^2 b_2 a_{21} f_y f(x_0, y_0) + O(h^3) \end{aligned}$$

Exakte Lösung im Vergleich (s. Beispiel 3.4)

$$y(x_0 + h) = y_0 + h f(x_0, y_0) + \frac{1}{2} h^2 (f_x + f_y f)(x_0, y_0) + O(h^3)$$

Als Bedingungsgleichungen für Konsistenzordnung $p = 2$ ergeben sich demnach

$$b_1 + b_2 = 1, \;\; b_1 c_1 + b_2 c_2 = \frac{1}{2}, \;\; b_2 a_{21} = \frac{1}{2}. \tag{3.21}$$

Als spezielle Lösungen von (3.21) hat man u.a. das Heun–Verfahren und den modifizierten Euler. Mit $s = 2$ Stufen ist jedoch die Ordnung $p = 3$ nicht erreichbar!

Offensichtlich wird bereits bei $s = 3$ Stufen der Abgleich sehr aufwändig. Das Vorgehen vereinfacht sich etwas, falls man nur autonome Differentialgleichungen betrachtet. Man wird von einem für autonome Differentialgleichungen hergeleiteten Verfahren natürlich verlangen, dass es auch im nichtautonomen Fall wohldefiniert ist (d.h. *invariant* unter Autonomisierung ist).

Wir vergleichen dazu $y' = f(x, y)$ mit der autonomisierten Gleichung

$$Y' = F(Y), \quad Y := \begin{pmatrix} y \\ x \end{pmatrix}, \quad F(Y) := \begin{pmatrix} f(x,y) \\ 1 \end{pmatrix}.$$

Der Runge–Kutta–Ansatz ist für beide Gleichungen nur dann äquivalent, falls

$$c_i = \sum_{j=1}^{i-1} a_{ij}, \tag{3.22}$$

denn wegen der letzten Zeile in F ergibt sich $x_0 + c_i h = x_0 + h \sum_{j=1}^{i-1} a_{ij} \cdot 1$ als Forderung.

Damit genügt es, den Abgleich nur im autonomen Fall durchzuführen und nachträglich mit den Koeffizienten a_{ij} die Zwischenstützstellen oder Knoten c_i mittels (3.22) zu definieren. Aber diese Erleichterung hilft nicht allzuviel, um Verfahren mit $s \geq 3$ Stufen und entsprechend höherer Ordnung zu gewinnen. Auch wenn die einzelnen Schritte nur aus wiederholter Taylorentwicklung bestehen, ist der Abgleich auf diese Art und Weise schnell nicht mehr handhabbar. Der folgende Satz gibt die Bedingungsgleichungen bis Ordnung 4 an.

Satz 3.5 *Ordnungsbedingungen*
Seien $f \in C^p(D)$ und A, b, c die Koeffizienten des Runge-Kutta-Verfahrens. Das Verfahren hat die Ordnung p, falls die Bedingung (3.22) und die folgenden Bedingungsgleichungen erfüllt sind:

$$p = 1: \quad \sum_{i=1}^{s} b_i = 1 \qquad (1)$$

$$p = 2: \quad \sum_{i=1}^{s} b_i c_i = 1/2 \qquad (2) \qquad \text{sowie (1)}$$

$$p = 3: \quad \sum_{i=1}^{s} b_i c_i^2 = 1/3, \quad \sum_{i,j=1}^{s} b_i a_{ij} c_j = 1/6 \quad (3) \quad \text{sowie (1)} - (2)$$

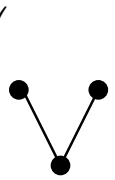
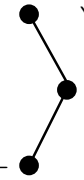
$$p = 4: \quad \sum_{i=1}^{s} b_i c_i^3 = 1/4, \quad \sum_{i,j=1}^{s} b_i c_i a_{ij} c_j = 1/8,$$

$$\sum_{i,j=1}^{s} b_i a_{ij} c_j^2 = 1/12, \quad \sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} c_k = 1/24 \quad \text{sowie (1)} - (3)$$

Für Ordnungen $p \geq 5$ hat Butcher ab 1963 ein graphentheoretische Hilfsmittel eingeführt, die sogenannten *Butcher–Bäume*, um die auftretenden *elementaren Differentiale* systematisch zu beschreiben. Beispielsweise ist (autonomer Fall)

$$y(x_0 + h) = y(x_0) + hf(y_0) + \frac{1}{2}h^2 f'f(y_0) + \frac{1}{3!}h^3 \left(f''(f,f) + f'f'f\right)(y_0) + \ldots$$

Die Darstellung über Butcher–Bäume sieht dann folgendermaßen aus:

$y(x_0 + h) = y(x_0) + h \cdot$  $+ \frac{1}{2}h^2$ $+ \frac{1}{3!}h^3 \Bigg($  $+$  $\Bigg) + \ldots$

Auf diese Art und Weise gelingt es, die fortgesetzte Taylorentwicklung systematisch und elegant darzustellen, und zwar sowohl für die exakte Lösung als auch für die numerische Approximation. (Näheres in Deuflhard/Bornemann und Hairer/Wanner)

Verfahrenskonstruktion

Die in Satz 3.5 angegebenen Bedingungsgleichungen sind nichtlinear, und deswegen stellt die Bestimmung geeigneter Koeffizienten, die sogenannte *Verfahrenskonstruktion*, eine schwierige Aufgabe dar. Wir wollen im folgenden die Situation bei $s = 4$ Stufen näher betrachten und Verfahren der Ordnung $p = 4$ konstruieren. (Zeige: Ordnung $p = 4$ ist mit $s = 3$ Stufen nicht erreichbar!)

Insgesamt hat man mit der Konvention $c_i = \sum a_{ij}$ die 10 Koeffizienten $b_1, b_2, b_3, b_4, a_{21}, a_{31}, a_{32}, a_{41}, a_{42}, a_{43}$ zu bestimmen, so dass sämtliche Bedingungen erfüllt sind.

Die Bedingung $\sum b_i = 1$ erinnert sofort an die Quadratur und legt nahe, $b_1, \ldots, b_4$ als *Gewichte* einer Quadraturformel zu interpretieren. Fassen wir die c_i zusätzlich als Stützstellen im Intervall $[0, 1]$ auf, so folgt aus den Bedingungen

$$\sum b_i c_i = 1/2, \quad \sum b_i c_i^2 = 1/3, \quad \sum b_i c_i^3 = 1/4,$$

dass diese Quadraturformel exakt für alle Polynome in $\mathbb{P}_3$ sein muss.

Zwei Quadraturformeln können wir nun heranziehen:

a) Die Newtonsche 3/8-Regel mit

$$c = (0,\, 1/3,\, 2/3,\, 1)^T, \quad b = (1/8,\, 3/8,\, 3/8,\, 1/8)^T.$$

Mit dieser Festlegung liefern $\sum b_i a_{ij} c_j = 1/6$, $\sum b_i c_i a_{ij} c_j = 1/8$ sowie $\sum b_i a_{ij} a_{jk} c_k = 1/24$ die Beziehungen

$$a_{32} = 1, \qquad a_{42} c_2 + a_{43} c_3 = 1/3, \qquad b_4 a_{43} a_{32} c_2 = 1/24.$$

Damit ist $a_{43} = 1$ und $a_{42} = -1$. Aus der Definition der c_i ergeben sich schließlich noch $a_{21} = 1/3, a_{31} = -1/3, a_{41} = 1$.
Resultat ist die *Kuttasche 3/8-Regel*

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

(Zu überprüfen ist noch die übriggebliebene Bedingung $\sum b_i a_{ij} c_j^2 = 1/12$, die von diesen Koeffizienten tatsächlich erfüllt wird.)

b) Die Simpson-Regel mit

$$c = (0,\, 1/2,\, 1/2,\, 1)^T, \quad b = (1/6,\, 2/6,\, 2/6,\, 1/6)^T,$$

wobei diese eigentlich nur 3 Knoten besitzt und deswegen der mittlere verdoppelt wurde. Ein analoges Vorgehen wie unter a) führt dann auf das "klassische" Runge-Kutta-Verfahren auf S. 37.

Wie oben erwähnt kann man mit Hilfe des Abgleichs über die Butcher-Reihen die Bedingungsgleichungen für Verfahren hoher Ordnung (≥ 5) zwar elegant aufstellen, die Zahl der Bedingungsgleichungen nimmt jedoch stark zu:

Ordnung p	1	2	3	4	5	6	7
Mindeststufen s	1	2	3	4	6	7	9
# Bed.gleich.	1	2	4	8	17	37	85
# Zahl Koeff.	1	3	6	10	21	28	45

Die Koeffizientenbestimmung ist dann nur noch mit weiteren vereinfachenden Annahmen (vergl. den Zugang über die Quadratur) und mit Computer–Algebra–Unterstützung (Maple, Mathematica) möglich. In der folgenden Tabelle ist aufgeführt, wieviele Stufen notwendig sind, um eine bestimmte Ordnung zu erreichen. Offensichtlich gilt die Ungleichung $s \geq p$.

Stufenzahl s	1	2	3	4	5	6	7	8
max. Ordnung p	1	2	3	4	4	5	6	6

Fehlerschätzung und Schrittweitensteuerung

Die bisher vorgestellten Verfahren wurden für ein gegebenes Gitter

$$x_0 < x_1 < \ldots < x_{n-1} < x_n$$

formuliert. Das einfachste Gitter ist das äquidistante, doch diese Wahl ist in den meisten Anwendungen wenig effizient. Ziel sollte es vielmehr sein, das Gitter so zu wählen, dass

(i) eine vorgegebene Genauigkeit der numerischen Lösung erreicht wird,

(ii) der dazu notwendige Rechenaufwand möglichst minimal wird.

Im folgenden wollen wir die Grundlagen für eine *adaptive* Gittersteuerung kennenlernen, vergleiche dazu den Algorithmus des adaptiven Simpson bei der numerischen Quadratur.

Verschiedene Faktoren sind bei der Gittersteuerung zu berücksichtigen. Bild 7 gibt eine qualitative Darstellung des Gesamtfehlers im äquidistanten Fall. Bei Vergrößern der Schrittweite verringert sich der Aufwand, möglicherweise steigt aber der Fehler an (vgl. Satz 3.4). Zu kleine Schrittweiten bedeuten dagegen mehr Aufwand und größeren Einfluss von Rundungsfehlern. Insbesondere gibt es eine untere Schranke, die die Schrittweite nicht unterschreiten sollte.

Schauen wir uns nun eine spezielle Lösungskurve an, so wird man offensichtlich fordern, dass starke Änderungen in der Lösung ein lokal feines Gitter verlangen, während annähernd konstante Abschnitte mit großen Schritten durchlaufen werden können, Abb. 8.

Da das Lösungsverhalten aber a priori nicht bekannt ist (im gewissen Ge-

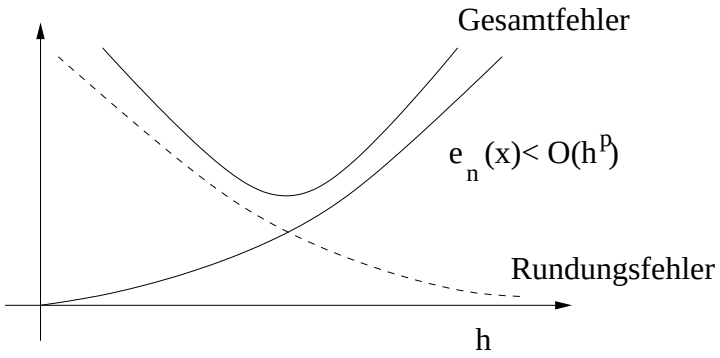


Abbildung 7: Verhalten des globalen Fehlers mit Rundungsfehlern

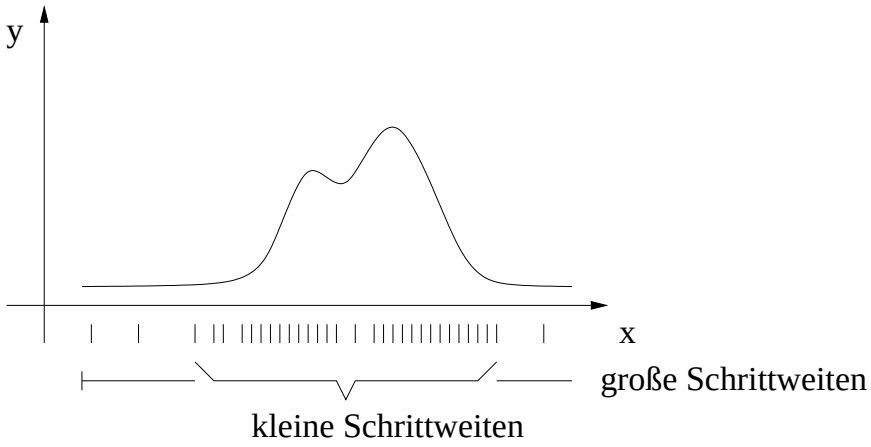


Abbildung 8: Nicht äquidistantes Gitter

gensatz zur Quadratur, bei der man den Integranden kennt), kann die Gitterstruktur nicht zu Beginn der numerischen Integration festgelegt werden. Stattdessen wird man die Gitterpunkte während des Lösungsprozesses schrittweise anpassen, d.h. ausgehend von (x_i, y_i) wird eine neue Schrittweite h_i bestimmt und damit x_{i+1} festgelegt. Man spricht deswegen von einer *Schrittweitensteuerung*, kurz SWS.

Das Ziel der SWS kann nun so formuliert werden: Wähle h_i so, dass die nächste Näherung y_{i+1} eine vorgegebene Fehlertoleranz erfüllt.

Allerdings steuern alle gängigen Verfahren nur den *lokalen Fehler* und nicht den globalen! D.h., die Schrittweite h_i wird so gewählt, dass für den aktuellen Schritt gilt

$$\|u(x_{i+1})|_{u(x_i)=y_i} - y_{i+1}\| \leq \epsilon \tag{3.23}$$

mit vorgegebener Toleranz ϵ. Die exakte Lösung $u(x)$ zum AW $u(x_i) = y_i$ ist im Kriterium (3.23) natürlich nicht bekannt und muss durch eine Approximation $\hat{y}_{i+1}$ ersetzt werden. Man spricht deswegen von der *Fehlerschätzung*, vergleiche das Vorgehen beim adaptiven Simpson.

Bei Runge–Kutta–Verfahren ist die folgende Technik am weitesten verbreitet: Kombiniere ein Verfahren der Ordnung p (für y_{i+1}) mit einem der Ordnung $p+1$ (für $\hat{y}_{i+1}$). Man bezeichnet das Verfahren für $\hat{y}_{i+1}$ als *eingebettetes Verfahren*.

Formal im Butcher–Tableau

$$\begin{array}{c|c} c & A \\ \hline & b^T \\ \hline & \hat{b}^T \end{array} \quad \begin{array}{l} \\ \\ \longrightarrow \; y_1 = y_0 + h \sum_{i=1}^{s} b_i K_i \\ \longrightarrow \; \hat{y}_1 = y_0 + h \sum_{i=1}^{\hat{s}} \hat{b}_i K_i \end{array}$$

Die Idee der Einbettung geht auf Fehlberg zurück, und die von ihm entwickelten adaptiven Verfahren nennt man deswegen *Runge–Kutta–Fehlberg–*

Verfahren. Details dazu und eine moderne Implementierung werden weiter unten besprochen. Im folgenden konzentrieren wir uns auf die Grundidee der Schrittweitensteuerung und nehmen an, eine zweite Approximation $\hat{y}_{i+1}$ der Ordnung $p+1$ stehe zur Verfügung.

Mit den Näherungen y_{i+1} und $\hat{y}_{i+1}$ setzt man nun folgendermaßen an, um eine "optimale" Schrittweite $h_{\text{opt.}}$ zu bestimmen: Es gilt

$$\|\hat{y}_{i+1} - y_{i+1}\| \doteq c \cdot h^{p+1} \quad (y_{i+1} \text{ Konsistenzord. } p).$$

Die Wunschschrittweite $h_{\text{opt.}}$ ist dagegen über

$$c \cdot h_{\text{opt.}}^{p+1} = \epsilon$$

gegeben. Durch Division folgt

$$h_{\text{opt.}}^{p+1} = h^{p+1} \frac{\epsilon}{\|\hat{y}_{i+1} - y_{i+1}\|}$$

bzw.

$$h_{\text{opt.}} = h \sqrt[p+1]{\frac{\epsilon}{\|\hat{y}_{i+1} - y_{i+1}\|}}. \tag{3.24}$$

Die Vorschrift (3.24) bildet die Grundlage für die Schrittweitensteuerung bei Einschrittverfahren. Wichtig dabei ist, dass die Idee der Herleitung eigentlich nur für kleine Schrittweiten gültig ist, denn nur dann macht die Darstellung des Fehlerschätzers als ch^{p+1} Sinn.

Für den Einsatz in der Praxis sind zahlreiche Heuristiken notwendig, um die Leistungsfähigkeit der SWS zu optimieren und Sonderfälle abzufangen. Wesentlich sind z.B. die Begrenzung der neuen Schrittweite nach oben/unten und die Einführung eines Sicherheitsfaktors.

Im folgenden Algorithmus ist diese Heuristik in den Grundzügen berücksichtigt. Die einzelnen Schritte sind allgemein gehalten und greifen mit Φ auf ein Verfahren der Ordnung p zurück sowie mit $\hat{\Phi}$ auf eines der Ordnung $p+1$.

Algorithmus 3.1 *Einschrittverfahren adaptiv*

Start: $x_0,\ y_0,\ h_0,\ \epsilon,\ x_{\text{end}}$ gegeben;
$i = 0$;
while $x_i < x_{\text{end}}$
    $x = x_i + h_i$;
    $y = y_i + h_i\Phi(x_i, y_i, h_i)$;
    $\hat{y} = y_i + h_i\hat{\Phi}(x_i, y_i, h_i)$;
    $e = \|\hat{y} - y\|$;
    $h_{\text{opt.}} = h\sqrt[p+1]{\frac{\epsilon}{e} \cdot \rho}$; % Sicherheitsf. ρ
    $h_{\text{opt.}} = \min(\alpha \cdot h_i,\ \max(\beta h_i,\ h_{\text{opt.}}))$; % Schranken α, β
    if $e \leq \epsilon$
        $x_{i+1} = x$;
        $y_{i+1} = y$;
        $h_{i+1} = \min(h_{\text{opt.}}, x_{\text{end}} - x_{i+1})$;
        $i = i + 1$;
    else
        $h_i = h_{\text{opt.}}$; % Wiederhole Schritt
    end
end

Bemerkungen:

1) Die Konstanten ρ, α, β sind abhängig vom konkreten Verfahren. Bei einem Verfahren höherer Ordnung kann eine größere Änderung von $h_{\text{opt.}}$ zugelassen werden als bei einem Verfahren niedriger Ordnung.

2) Geeignete Norm zur Fehlerschätzung:

$$\text{ERR} \quad = \sqrt{\frac{1}{n}\sum_{j=1}^{n}\left(\frac{\hat{y}_j - y_j}{ATOL + RTOL \cdot WT_j}\right)^2}\ ,\ WT = |\hat{y}|$$

mit absoluter/relativer Toleranz $ATOL,\ RTOL$. Statt $e \leq \epsilon$ vergleicht man dann den Fehler gegen 1, ERR ≤ 1.

In Algorithmus 3.1 wird die weniger genaue Lösung als neue Approximation herangezogen, denn für sie liegt mit der Differenz $\hat{y} - y$ ein Fehlerschätzer vor. Diese Argumentation gilt heute als überholt, und man rechnet in den gängigen Verfahren mit der *besseren Lösung* weiter, ein Vorgehen, das man auch als lokale Extrapolation bezeichnet. Begründung: Der Fehlerschätzer bezieht sich auf den lokalen Fehler und liefert kaum Information über den eigentlich wichtigeren globalen Fehler. Stattdessen wird über die SWS das Gitter an den lokalen Lösungsverlauf angepasst, und diese Anpassung funktioniert, wie die Erfahrung zeigt, mit der lokalen Extrapolation genauso gut, mit dem zusätzlichen Plus der höheren Ordnung.

Damit führen wir die folgende *Verbesserung* im Algorithmus 3.1 ein: Φ ist ein Verfahren der Ordnung $p+1$ und $\hat{\Phi}$ eines der Ordnung p. Dann arbeitet der Algorithmus automatisch im Modus der lokalen Extrapolation; alle anderen Schritte bleiben wie gehabt.

Eingebettete Verfahren

Wie oben bereits erwähnt, stellt die Einbettung eines zweiten RK-Verfahrens, das möglichst wenig zusätzlichen Aufwand erfordert, die gängige Technik zur SWS dar. Am Beispiel des klassischen RK-Verfahrens mit $p = 4$ wollen wir die prinzipielle Vorgehensweise studieren. Im Tableau haben wir also eine zusätzliche Zeile

$$
\begin{array}{c|cccccc}
0 & & & & & & \\
\frac{1}{2} & \frac{1}{2} & & & & & \\
\frac{1}{2} & 0 & \frac{1}{2} & & & & \\
1 & 0 & 0 & 1 & & & \\
\hline
 & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} & & \\
\hline
 & \hat{b}_1 & \hat{b}_2 & \hat{b}_3 & \dots & & \hat{b}_{\hat{s}}
\end{array}
$$

Erster Ansatz: Wir nehmen $\hat{s} = 4$ an und versuchen, die vier Koeffizienten $\hat{b}_1, \ldots, \hat{b}_4$ so zu bestimmen, dass

$$\hat{y}_1 = y_0 + h(\hat{b}_1 K_1 + \ldots + \hat{b}_4 K_4)$$

eine Approximation der Ordnung 3 darstellt. (Vergleiche die Diskussion oben: Wir sehen das gegebene Verfahren als das bessere an und suchen dazu ein zweites, das Information für die SWS liefert.) Aus den Bedingungsgleichungen für Ordnung $p = 3$ folgt (Satz 3.5)

$$\begin{aligned} \hat{b}_1 + \hat{b}_2 + \hat{b}_3 + \hat{b}_4 &= 1, & \hat{b}_2/2 + \hat{b}_3/2 + \hat{b}_4 &= 1/2, \\ \hat{b}_2/4 + \hat{b}_3/4 + \hat{b}_4 &= 1/3, & \hat{b}_3/4 + \hat{b}_4/2 &= 1/6, \end{aligned}$$

wobei die Koeffizienten a_{ij} und c_i bereits eingesetzt wurden. Einzige Lösung des linearen Gleichungssystems ist $\hat{b} = b$, d.h. mit $\hat{s} = 4$ ist es nicht möglich, solch ein eingebettetes Verfahren zu konstruieren!

Zweiter Ansatz: Mit $\hat{s} = 5$ bekommt man mehr Freiheitsgrade, dies sollte aber nicht zu einem höheren Aufwand führen. Eine weitere Auswertung einer Korrektur K_5 stellt einen solchen Aufwand dar, dieser fällt aber durch den *Fehlberg-Trick* nicht ins Gewicht: Wir bestimmen die neu ins Spiel kommenden Koeffizienten $c_5, a_{51}, \ldots, a_{54}$ so, dass

$$K_5(\text{alter Schritt}) = K_1(\text{neuer Schritt}). \tag{3.25}$$

Man nennt diese Technik auch FSAL (FirstSameAsLast). Konkret bedeutet dies

$$K_5(\text{a.S.}) = f(x_0 + c_5 h, y_0 + h \sum a_{5j} K_j) = f(x_0 + h, y_0 + h \sum b_i K_i) = K_1(\text{n.S.}).$$

Also ist $c_5 = 1$ und $a_{5j} = b_j$ für $j = 1 : 4$. Mit diesen Koeffizienten geht man nochmals in die Bedingungsgleichungen für die Ordnung $p = 3$. Es zeigt sich: Nun existiert eine ganze Familie von Gewichten $\hat{b}$, die die Ordnung 3

liefern. Ein Beispiel zeigt das folgende Butcher-Tableau:

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

Die Auswertung der Differenz $\hat{y} - y$ reduziert sich hier auf die einfache Formel $\hat{y} - y = (K_1(\text{n.S.}) - K_4(\text{a.S.}))/6$, d.h. die Größe $\hat{y}$ muss gar nicht explizit berechnet werden.

Bemerkung:

Eine in der Herleitung einfache, aber in der Auswertung teure Alternative zu den eingebetteten Verfahren ist die *Richardson-Extrapolation*, bei der ein Schritt $y_1 = y_0 + 2h\Phi(x_0, y_0, 2h)$ mit doppelter Schrittweite und zwei Schritte $\tilde{y}_{1/2} = y_0 + h\Phi(x_0, y_0, h),\ \tilde{y}_1 = \tilde{y}_{1/2} + h\Phi(x_0 + h, \tilde{y}_{1/2}, h)$ mit einfacher Schrittweite miteinander kombiniert werden, um mittels Extrapolation eine Approximation höherer Ordnung zu erhalten. Vergleiche den adaptiven Simpson.

Das Dormand-Prince-Verfahren DOPRI5(4)

Das zur Zeit wohl am weitesten verbreitete und für mittlere Genauigkeitsanforderungen effizienteste explizite Runge-Kutta-Verfahren geht auf Dormand und Prince (1980) zurück.

Das Verfahren hat die Ordnung 5 mit einem eingebetteten Verfahren der Ordnung 4, daher stammt die Namensgebung DOPRI5(4). Erreicht wird dies mit $s = 6$ und $\hat{s} = 7$ Stufen, wobei durch den FSAL-Ansatz die zusätzliche Stufe nicht ins Gewicht fällt. Man hat also effektiv nur 6 Stufen

pro Schritt auszuwerten. Der Koeffizientensatz lautet

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

Die Bestimmung der Koeffizienten erfolgte nach mehreren Kriterien:

(i) Der führende Fehlerterm $\delta_6(y)$ des Verfahrens 5. Ordnung soll möglichst klein sein, der führende Fehlerterm $\hat{\delta}_5(y)$ des Verfahrens 4. Ordnung dagegen soll möglichst dominant sein im Vergleich zu den weiteren Termen der zugehörigen Fehlerentwicklung.
(ii) Der Term $\hat{\hat{\delta}}_5(y)$, der sich ja aus elementaren Differentialen von f zusammensetzt, soll im Sonderfall der Quadratur $f = f(x)$ nicht verschwinden.
(iii) Die Koeffizienten c_i sollen in $[0, 1]$ liegen und verschieden sein.

In die Bestimmung gehen daneben auch viel Erfahrung und Fingerspitzengefühl ein!

Einen guten Integrationscode zeichnen aber nicht nur die Koeffizienten aus. Zwei weitere Aspekte sind von besonderer Bedeutung:

Kontinuierliche Lösungsdarstellung oder Dense Output: Die SWS sorgt dafür, dass der Integrator möglichst große Schritte macht, um eine vorgegebene Genauigkeit einzuhalten. Oft will man aber die Lösung zu sehr vielen Zeitpunkten auswerten, und dies ohne die Integration mit entsprechend verkleinerten Schritten durchführen zu müssen. Hilfsmittel dazu ist eine kontinuierliche Lösungsdarstellung (engl. Dense Output), d.h. ein Interpolant, der die diskreten Daten (x_i, y_i) interpoliert und dazwischen die

exakte Lösung möglichst gut approximiert. Grundsätzlich geht man dabei intervallweise vor und konstruiert nach jedem Schritt den Interpolanten auf $[x_i, x_{i+1}]$.

Die einfachste Methode bei ESV verwendet die Daten y_i, f_i sowie y_{i+1}, f_{i+1} und konstruiert das Hermite–Interpolationspolynom vom Grad 3

$$u(\theta) = (1-\theta)y_i + \theta y_{i+1} + \theta(\theta-1)\left((1-2\theta)(y_{i+1}-y_i) + (\theta-1)hf_i + \theta h f_{i+1}\right)$$

mit $0 \le \theta \le 1$. Probe: $u(0) = y_i,\ u'(0) = y_i' = f_i,\ u(1) = y_{i+1},\ u'(1) = f_{i+1}$

Falls das Verfahren Ordnung $p \ge 3$ hat, stellt der Hermite-Interpolant eine stetige Erweiterung des Runge-Kutta-Verfahrens dar, die die exakte Lösung im aktuellen Intervall mit Ordnung 3 approximiert und global C^1 ist. Damit kann der Integrator nach jedem Schritt überprüfen, ob Ausgabepunkte erreicht wurden und nachträglich an diesen Stellen das Polynom u auswerten.

Zu DOPRI5(4) wurde eine stetige Erweiterung der Ordnung 4 entwickelt, die auf einem Polynom vom Grad 5 basiert.

Verbesserte SWS mit PI-Regler: Die Steuerung der Schrittweite wurde im Jahre 1988 durch eine regelungstheoretische Analyse (Gustafsson / Lundh / Söderlind) auf ein solideres Fundament gestellt. Insbesondere verbesserte man die Formel (3.24) für die neue Schrittweite um einen Korrekturterm, der für glattere Schrittweitenverläufe und weniger Schrittverwerfungen sorgt.

Die Formel (3.24) ist in diesem Zusammenhang ein *I-Regler*, d.h. sie berücksichtigt die Summe der bisherigen Abweichungen vom Sollwert (das Integral). Ein *PI-Regler* ist zusätzlich Proportional zur letzten Abweichung vom Sollwert und aus regelungstechnischer Sicht besser. Man kommt so auf die Schrittweitenformel

$$h_{i+1} = h_i \left(\frac{\epsilon}{\|\hat{y}_{i+1} - y_{i+1}\|}\right)^{\alpha} \left(\frac{\|\hat{y}_i - y_i\|}{\epsilon}\right)^{\beta} \tag{3.26}$$

mit Konstanten $\alpha = 0.7/p$, $\beta = 0.4/p$.

3.5 Mehrschrittverfahren

Mehrschrittverfahren (MSV) bilden neben den ESV die zweite große Verfahrensklasse. Bei ihnen fließen Daten / Lösungspunkte aus der Vergangenheit in die Berechnungvorschrift mit ein. Genauer:
In einem *k–Schrittverfahren* berechnet man die Approximation $y_{i+1} \doteq y(x_{i+1})$ aus den Daten

$$(x_{i-k+1}, y_{i-k+1}), (x_{i-k+2}, y_{i-k+2}), \ldots, (x_{i-1}, y_{i-1}), (x_i, y_i).$$

Die Daten stammen aus den vorherigen Schritten oder einer *Anlaufrechnung*. Neben der Konsistenz ist die Eigenschaft der Stabilität bei MSV notwendig, um die Konvergenz zu zeigen. Insofern zeigt sich hier ein wesentlicher Unterschied zu den ESV.

Verfahrensklassen

Adams–Verfahren. Zur Herleitung der Adams–Verfahren integriert man die Differentialgleichung $y' = f(x, y)$ von x_i bis x_{i+1} und erhält

$$y(x_{i+1}) = y(x_i) + \int_{x_i}^{x_{i+1}} f(x, y(x))\ dx\ . \tag{3.27}$$

Der Integrand in (3.27) hängt von der (unbekannten) Lösung y ab und wird nun durch ein Interpolationspolynom p ersetzt. Zwei Fälle werden unterschieden:

a) Das Polynom p interpoliert $f_{i-k+j} = f(x_{i-k+j}, y_{i-k+j})$, $j = 1, \ldots, k$

Über die Lagrange-Interpolation hat man die Darstellung

$$p(x) = \sum_{j=1}^{k} f_{i-k+j} \cdot l_{i,j}(x) \quad \text{mit} \quad l_{i,j}(x) = \prod_{\substack{\nu=1 \\ \nu \neq j}}^{k} \frac{x - x_{i-k+\nu}}{x_{i-k+j} - x_{i-k+\nu}}.$$

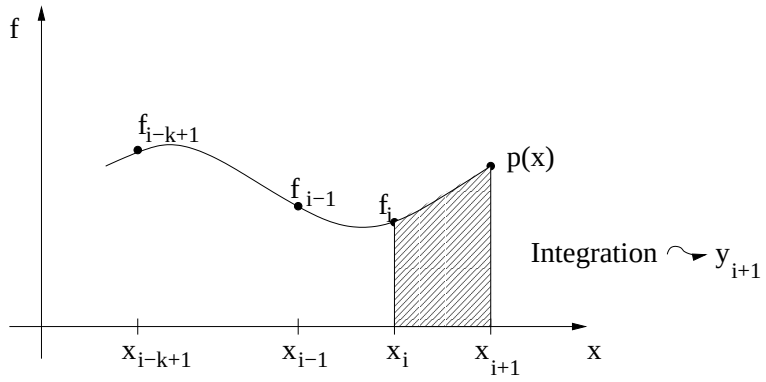


Abbildung 9: Idee der Adams-Bashforth-Verfahren

Mit Einsetzen von $p(x)$ im Integral in (3.27) folgt

$$y_{i+1} = y_i + h_i \sum_{j=1}^{k} \beta_{i,j} f_{i-k+j}, \qquad \beta_{i,j} := \frac{1}{h_i} \int_{x_i}^{x_{i+1}} l_{i,j}(x)\, dx.$$

Für äquidistantes Gitter $x_i = x_0 + i \cdot h$ sind die Koeffizienten $\beta_{i,j} =: \beta_{j-1}$, $j = 1, \ldots, k$ unabhängig vom Schritt i,

$$\begin{aligned} \int_{x_i}^{x_{i+1}} l_{i,j}(x)\, dx &= h \int_0^1 \prod_{\substack{\nu=1 \\ \nu \neq j}}^{k} \frac{x_0 + ih + sh - (x_0 + (i-k+\nu)h)}{x_0 + (i-k+j)h - (x_0 + (i-k+\nu)h)}\, ds \\ &= h \underbrace{\int_0^1 \prod_{\substack{\nu=1 \\ \nu \neq j}}^{k} \frac{k - \nu + s}{j - \nu}\, ds}_{=:\beta_{j-1}}. \end{aligned}$$

Die Verfahrensvorschrift schreibt sich dann folgendermaßen:

$$y_{i+1} = y_i + h(\beta_0 f_{i-k+1} + \beta_1 f_{i-k+2} + \ldots + \beta_{k-1} f_i) \tag{3.28}$$

As spezielle Verfahren hat man

$$\begin{aligned} \underline{k=1} \qquad & y_{i+1} = y_i + h f_i, \\ \underline{k=2} \qquad & y_{i+1} = y_i + h\left(-\tfrac{1}{2} f_{i-1} + \tfrac{3}{2} f_i\right), \\ \underline{k=3} \qquad & y_{i+1} = y_i + h\left(\tfrac{5}{12} f_{i-2} - \tfrac{16}{12} f_{i-1} + \tfrac{23}{12} f_i\right). \end{aligned}$$

Die so gewonnenen expliziten MSV nennt man *Adams–Bashforth–Verfahren.*

b) Ein alternativer Zugang fordert, dass das Polynom p die Werte f_{i-k+j} für $j = 1, \ldots, k, k+1$ interpoliert. Da p damit von y_{i+1} abhängt, liegt ein implizites Verfahren vor.

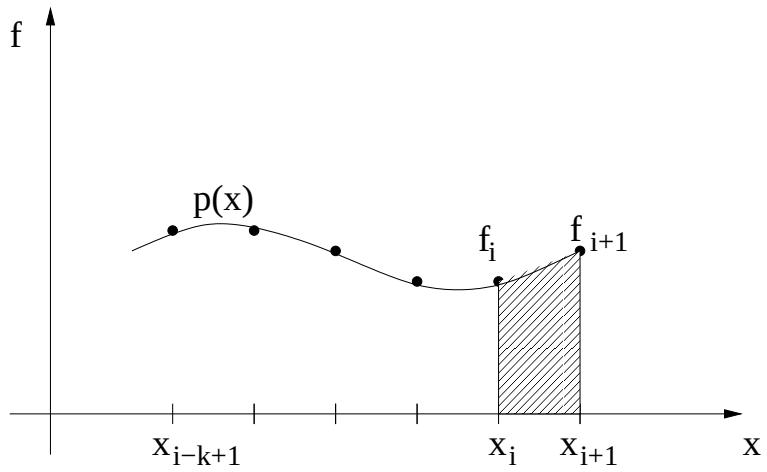


Abbildung 10: Idee der Adams-Moulton-Verfahren

Analoge Konstruktion des Interpolanten wie unter a) liefert

$$p(x) = \sum_{j\equiv 1}^{k+1} f_{i-k+j} \cdot l^*_{i,j}(x), \qquad l^*_{i,j}(x) = \prod_{\substack{\nu=1 \\ \nu\neq j}}^{k+1} \frac{x - x_{i-k+\nu}}{x_{i-k+j} - x_{i-k+\nu}}.$$

Bei äquidistantem Gitter erhält man die Koeffizienten

$$\beta^*_{j-1} = \int_0^1 \prod_{\substack{\nu=1 \\ \nu\neq j}}^{k+1} \frac{k-\nu+s}{j-\nu}\, ds, \quad j = 1, \ldots, k+1$$

und als Verfahren

$$y_{i+1} = y_i + h(\beta^*_0 f_{i-k+1} + \ldots + \beta^*_{k-1} f_i + \beta^*_k f_{i+1}). \tag{3.29}$$

Spezielle Vertreter sind

$$\begin{array}{ll} \underline{k=1} & y_{i+1} = y_i + h\left(\frac{1}{2}f_i + \frac{1}{2}f_{i+1}\right) \quad \text{Trapezregel,} \\ \underline{k=2} & y_{i+1} = y_i + h\left(-\frac{1}{12}f_{i-1}+, \frac{8}{12}f_i + \frac{5}{12}f_{i+1}\right). \end{array}$$

Diese Klasse von impliziten MSV bezeichnet man als *Adams–Moulton–Verfahren*. Der Fall $k = 0$ macht auch Sinn, daraus ergibt sich

$$y_{i+1} = y_i + hf_{i+1} \quad \text{Impliziter Euler.}$$

Adams–Verfahren sind in der Praxis von großer Bedeutung und werden meist als *Prädiktor–Korrektor–Schema* implementiert:

(i) $y_{i+1}^{(0)} = y_i + h(\beta_0 f_{i-k+1} + \ldots + \beta_{k-1} f_i)$
(Startschritt mit explizitem Adams–Bashforth–Verfahren)

(ii) $y_{i+1}^{(l+1)} = \Psi(y_{i+1}^{(l)}),\ l = 0, 1, \ldots$
Fixpunktiteration Korrektorschritt; mit implizitem Adams–Moulton–Verfahren, wobei

$$\Psi(y) := h\beta_k^* f(x_{i+1}, y) + y_i + h(\beta_0^* f_{i-k+1} + \ldots + \beta_{k-1}^* f_i).$$

Die Integration im Korrektor konvergiert für h hinreichend klein gegen einen Fixpunkt mit $y = \Psi(y)$, da dann

$$\|\Psi'(y)\| = h|\beta_k^*|\ \|f_y(x_{i+1}, y)\| \leq M < 1$$

gezeigt werden kann (vgl. Banachscher Fixpunktsatz).

Verfahren nach Nyström und Milne. Analog zur Herleitung der Adams–Verfahren kann man weitere MSV konstruieren, indem man $y' = f(x, y)$ von x_{i-1} bis x_{i+1} integriert, d.h. über 2 Intervalle. Der Fall a) mit explizit gegebenem Interpolanten p führt dann auf die *Nyström–Verfahren*. Ein Beispiel ist der Fall

$$\underline{k = 2} \qquad y_{i+1} = y_{i-1} + 2hf_i,$$

der die Mittelpunktregel liefert.

Der Fall b) mit implizitem Ansatz definiert die *Verfahren nach Milne*. Als Beispiel betrachte

$$\underline{k = 2} \qquad y_{i+1} = y_{i-1} + h/3(f_{i-1} + 4f_i + f_{i+1}),$$

eine Verallgemeinerung der Keplerschen Fassregel aus der Quadratur (Verfahren von Milne–Simpson).

BDF–Verfahren. Die Herleitung der *BDF (Backward Difference Formulas)–Verfahren* basiert auf der *Differentation* (und nicht der Integration wie bei Adams). Man konstruiert einen Interpolanten $q(x)$ durch

$$(x_{i-k+1}, y_{i-k+1}), \ldots, (x_i, y_i), (x_{i+1}, y_{i+1})$$

und fordert, dass $q(x_{i+1})$ die Differentialgleichung erfüllt, d.h.

$$q'(x_{i+1}) = f(x_{i+1}, q(x_{i+1})) = f(x_{i+1}, y_{i+1}).$$

In der Lagrange-Darstellung ist

$$q(x) = \sum_{j=1}^{k+1} y_{i-k+j} l^*_{i,j}(x) \quad \Rightarrow \quad q'(x_{i+1}) = \sum_{j=1}^{k+1} y_{i-k+j} l^{*,\prime}_{i,j}(x_{i+1}).$$

Im Fall äquidistanter Schrittweiten gilt weiter

$$\frac{d}{dx} l^*_{i,j}(x)|_{x=x_{i+1}} = \frac{1}{h} \underbrace{\left(\frac{d}{ds} \prod_{\substack{\nu=1 \\ \nu \neq j}}^{k+1} \frac{k-\nu+s}{j-\nu} \right) |_{s=1}}_{=:\alpha_{j-1}}$$

Die Koeffizienten $\alpha_0, \ldots, \alpha_k$ sind wiederum unabhängig vom Schritt i, und allgemein lautet das k-Schritt-BDF-Verfahren dann

$$\alpha_0 y_{i-k+1} + \ldots + \alpha_{k-1} y_i + \alpha_k y_{i+1} = h f_{i+1}. \tag{3.30}$$

BDF–Verfahren sind implizit und finden vor allem bei steifen Differentialgleichungen Anwendung. Beispiele sind

$\underline{k=1}$	$y_{i+1} - y_i = h f(x_{i+1}, y_{i+1})$	(Impliziter Euler)
$\underline{k=2}$	$\frac{3}{2} y_{i+1} - 2y_i + \frac{1}{2} y_{i-1} = h f_{i+1}$	(BDF-2)
$\underline{k=3}$	$\frac{11}{6} y_{i+1} - 3y_i + \frac{3}{2} y_{i-1} - \frac{1}{3} y_{i-2} = h f_{i+1}$	(BDF-3)

Im Folgenden werden alle bisher eingeführten Verfahren unter einer einheitlichen Notation zusammengefaßt und analysiert. Statt

$$\alpha_0 y_{i-k+1} + \ldots + \alpha_k y_{i+1} \;=\; h(\beta_0 f_{i-k+1} + \ldots + \beta_{k-1} f_i + \beta_k f_{i+1})$$

schreibt man ein *lineares k-Schritt-Verfahren* bequemer als

$$\alpha_0 y_i + \alpha_1 y_{i+1} + \ldots + \alpha_k y_{i+k} = h(\beta_0 f_i + \ldots + \beta_k f_{i+k})$$

bzw. kurz

$$\sum_{l=0}^{k} \alpha_l y_{i+l} = h \sum_{l=0}^{k} \beta_l f_{i+l} \tag{3.31}$$

mit zunächst beliebigen reellen Verfahrenskoeffizienten α_l und β_l.

3.6 Konsistenz von MSV

Analog zu den ESV untersuchen wir nun den Fehler eines MSV in einem Schritt. Dazu gibt es verschiedene Ansätze, die im Wesentlichen äquivalent sind. Wir verwenden hier den *Defekt* als lokale Fehlerdefinition, vgl. Definition 3.1.

Definition 3.4 *Lokaler Diskretisierungsfehler MSV*
Sei $y(x)$ exakte Lösung des AWP $y' = f(x,y)$, $y(x_0) = y_0$. Der lokale Diskretisierungsfehler des MSV (3.31) mit konstanter Schrittweite h ist definiert als Defekt

$$\tau(h) := \frac{1}{h} \left(\sum_{l=0}^{k} \alpha_l y(x_{i+l}) - h \sum_{l=0}^{k} \beta_l f(x_{i+l}, y(x_{i+l})) \right).$$

Das Fehlermaß $\tau(h)$ ergibt sich also durch Einsetzen der exakten Lösung in die Differenzengleichung (3.31). Im Sonderfall exakter Daten

$$y_i = y(x_i), \ldots, y_{i+k-1} = y(x_{i+k-1})$$

hat man für die numerische Lösung unter der Annahme $\beta_k = 0$ (explizites Verfahren)

$$\alpha_k y_{i+k} + \sum_{l=0}^{k-1} \alpha_l y_{i+l} = h \sum_{l=0}^{k-1} \beta_l f_{i+l}$$

und für die exakte Lösung

$$\begin{aligned} \alpha_k y(x_{i+k}) + \sum_{l=0}^{k-1} \alpha_l y(x_{i+l}) &= h \sum_{l=0}^{k-1} \beta_l f_{i+l} + h \cdot \tau(h) \\ \Longrightarrow \quad \tau(h) &= \frac{\alpha_k}{h} \left(y(x_{i+k}) - y_{i+k} \right) . \end{aligned} \tag{3.32}$$

Die Darstellung (3.32) ist analog zur Definition 3.1 des lokalen Fehlers bei ESV. Oft normiert man das MSV noch durch $\alpha_k := 1$, so dass (3.32) unabhängig von Verfahrenskoeffizienten geschrieben werden kann.

Definition 3.5 *Konsistenzordnung MSV*
Ein MSV heisst konsistent, falls der lokale Diskretisierungsfehler $\tau(h)$ für $h \to 0$ ebenfalls gegen 0 strebt:

$$\|\tau(h)\| \leq \gamma(h) \quad mit \quad \lim_{h \to 0} \gamma(h) = 0 \, .$$

Das MSV hat Konsistenzordnung p, falls

$$\|\tau(h)\| = O(h^p) \ .$$

Zur Bestimmung der Konsistenzordnung eines MSV führt man ähnlich den RK–Verfahren einen Abgleich durch und ermittelt Bedingungsgleichungen für die Koeffizienten $\alpha_l, \ \beta_l$. Man setzt an

$$\frac{1}{h} \left(\sum_{l=0}^{k} \alpha_l y(x + lh) - h \sum_{l=0}^{k} \beta_l y'(x + lh) \right) = \tau(h)$$

$$\begin{aligned} \text{mit} \qquad y(x + lh)) &= y(x) + y'(x) lh + \frac{1}{2} y''(x) (lh)^2 + \ldots \\ y'(x + lh)) &= y'(x) + y''(x) lh + \frac{1}{2} y'''(x) (lh)^2 + \ldots \end{aligned}$$

Sortieren nach h–Potenzen ergibt

$$\begin{aligned}\tau(h) = \frac{1}{h}y(x)\sum_{l=0}^{k}\alpha_l \;&+\; y'(x)\sum_{l=0}^{k}(\alpha_l\cdot l-\beta_l)\\ &+\; \frac{h}{2}y''(x)\sum_{l=0}^{k}(\alpha_l\cdot l^2-2\beta_l l)\\ &\vdots\\ &+\; \frac{h^{p-1}}{p!}y^{(p)}(x)\sum_{l=0}^{k}(\alpha_l\cdot l^p-p\cdot\beta_l l^{p-1})\\ &+\; O(h^p)\end{aligned} \tag{3.33}$$

Ein MSV ist demnach konsistent, falls

$$\sum_{l=0}^{k}\alpha_l = 0 \quad\text{und}\quad \sum_{l=0}^{k}(\alpha_l\cdot l-\beta_l)=0. \tag{3.34}$$

Für die Ordnung $p\geq 1$ müssen auch die höheren Terme in der Entwicklung (3.33) herausfallen, d.h.

$$\sum_{l=0}^{k}\alpha_l l^q = q\sum_{l=0}^{k}\beta_l l^{q-1} \quad\text{für}\quad q=1,\ldots,p. \tag{3.35}$$

Beispiel 3.9: Gegeben ist das MSV

$$y_{i+2}+4y_{i+1}-5y_i = h(4f_{i+1}+2f_i).$$

Die Verfahrenskoeffizienten sind also $\alpha_2=1$, $\alpha_1=4$, $\alpha_0=-5$, $\beta_1=4$, $\beta_0=2$

$$\begin{aligned}\text{Konsistent:}\quad &\textstyle\sum\alpha_l=0,\quad \sum\alpha_l\cdot l-\beta_l=(-5\cdot 0-2)+(4\cdot 1-4)+(1\cdot 2)=0\\ p=2:\quad &\textstyle\sum\alpha_l\cdot l^2-2\beta_l l=(-5\cdot 0-2\cdot 0)+(4\cdot 1-8)+(1\cdot 4)=0\\ p=3:\quad &\textstyle\sum\alpha_l\cdot l^3-3\beta_l l^2=(-5\cdot 0-2\cdot 0)+(4\cdot 1-12)+(1\cdot 8)=0\end{aligned}$$

Also liegt ein 2–Schritt–Verfahren der Konsistenzordnung $p=3$ vor.

Alternativ zu den Bedingungsgleichungen (3.34) und (3.35) benutzt man die *erzeugenden Polynome*

$$\rho(\zeta) : = \alpha_k\zeta^k+\ldots+\alpha_1\zeta+\alpha_0\ , \tag{3.36}$$

$$\sigma(\zeta) : = \beta_k\zeta^k+\ldots+\beta_1\zeta+\beta_0\ , \tag{3.37}$$

um die Konsistenzordnung zu charakterisieren. Es folgt unmittelbar die Äquivalenz der Konsistenzbedingung (3.34) mit

$$\rho(1) = 0 \quad \text{und} \quad \rho'(1) = \sigma(1). \tag{3.38}$$

Die weiteren Bedingungen erhält man über den Ansatz

$$\begin{aligned} \rho(e^h) = \sum_{l=0}^{k} \alpha_l e^{lh} &= \sum_{l=0}^{k} \alpha_l (1 + lh + \frac{1}{2}(lh)^2 + \ldots) \\ \sigma(e^h) = \sum_{l=0}^{k} \beta_l e^{lh} &= \sum_{l=0}^{k} \beta_l (1 + lh + \frac{1}{2}(lh)^2 + \ldots) \end{aligned}$$

Daraus schließt man auf

$$\begin{aligned} \rho(e^h) - h\sigma(e^h) &= \sum_{l=0}^{k} \alpha_l + h \sum_{l=0}^{k} (\alpha_l \cdot l - \beta_l) \\ &\quad + \frac{1}{2} h^2 \sum_{l=0}^{k} (\alpha_l \cdot l^2 - 2\beta_l \cdot l) \\ &\quad \vdots \\ &\quad + \frac{h^p}{p!} \sum_{l=0}^{k} (\alpha_l \cdot l^p - p\beta_l \cdot l^{p-1}) + O(h^{p+1}). \end{aligned} \tag{3.39}$$

D.h., die Entwicklung (3.33) lässt sich auch über die Exponentialfunktion kompakt darstellen. Setzt man noch $h = \ln \mu$, dann ist $h = 0 \Leftrightarrow \mu = 1$ und

$\rho(e^h) - h\sigma(e^h) = \mathcal{O}(h^{p+1}) \Leftrightarrow \rho(\mu) - \ln \mu \cdot \sigma(\mu)$ hat $\mu = 1$ als $p+1$–fache NST

Satz 3.6 *Konsistenzordnung MSV*
Das MSV $\sum \alpha_l y_{i+l} = h \sum \beta_l f_{i+l}$ *hat die Konsistenzordnung* p*, falls eine der drei äquivalenten Bedingungen erfüllt ist:*

(i) $\sum_{l=0}^{k} \alpha_l = 0$ *und* $\sum_{l=0}^{k} \alpha_l l^q = q \sum_{l=0}^{k} \beta_l l^{q-1}$ *für* $q = 1, \ldots, p$

(ii) $\rho(e^h) - h\sigma(e^h) = O(h^{p+1})$ *für* $h \to 0$.

(iii) $\rho(\mu) - \ln \mu \cdot \sigma(\mu)$ *hat* $\mu = 1$ *als* $p+1$*–fache Nullstelle*

Bemerkungen

1) In der Literatur schreibt man oft $\tau(x, y, h)$ statt $\tau(h)$, um die Abhängigkeit vom Gitter und der Lösung zu betonen. Bedingung (ii) oben bedeutet dann $\tau(x, \exp, h) = O(h^p)$. Ein anderer Ansatz verwendet statt $\tau(x, y, h)$ den Differenzenoperator

$$L(x, y, h) := \sum_{l=0}^{k} \alpha_l y(x + lh) - h \sum_{l=0}^{k} \beta_l y'(x + lh) = h\tau(x, y, h).$$

2) Einordnung Verfahrensklassen:

	k–Schritt–Adams–Bashforth	Adams–Moulton	BDF
p	k	$k+1$	k

Was ist die maximal erreichbare Konsistenzordnung? Insgesamt hat man $2k + 2$ Parameter $\alpha_0, \ldots, \alpha_k$, $\beta_0, \ldots, \beta_k$. Nach Normierung $\alpha_k = 1$ stehen noch $2k + 1$ Parameter zur Verfügung. Wie man zeigen kann, existiert ein implizites MSV der Ordnung $p = 2k$ und ein explizites mit $p = 2k - 1$. Die maximal erreichbare Ordnung ist also deutlich höher als bei Adams- oder BDF–Verfahren. Allerdings sind solche Verfahren nicht mehr *stabil*, siehe den nächsten Abschnitt.

Vor der Stabilitätsanalyse noch die Definition des globalen Fehlers:

Definition 3.7 *Globaler Diskretisierungsfehler*
Der globale Diskretisierungsfehler eines MSV ist die Differenz

$$e_n(X) = y(X) - y_n \quad \textit{mit} \quad X = x_n \quad \textit{fest, } n \textit{ variabel.}$$

3.7 Stabilität von MSV

Wie hängen lokaler Fehler (d.h. die Konsistenz) und globaler Fehler (d.h. Konvergenz) zusammen? Anders als bei ESV braucht man bei MSV eine zusätzliche Stabilitätsbedingung, um Konvergenz zu zeigen.

Beispiel 3.10 Als Beispiel für die Problematik betrachte das MSV mit $k = 2$ und $p = 3$ aus Beispiel 4.1,

$$y_{i+2} + 4y_{i+1} - 5y_i = h(4f_{i+1} + 2f_i).$$

Erzeugende Polynome sind $\rho(\zeta) = \zeta^2 + 4\zeta - 5$ sowie $\sigma(\zeta) = 4\zeta + 2$. Das MSV wird angewendet auf die triviale (!) Differentialgleichung

$$y' = 0\ , \quad \text{AW} \quad y_0 = 1\ , \quad \text{sowie} \quad y_1 = h\epsilon. \tag{3.40}$$

Daraus ergibt sich die *3–Term–Rekursion*

$$y_{i+2} + 4y_{i+1} - 5y_i = 0 \quad \text{Start} \quad y_0 = 1,\ y_1 = 1 + h\epsilon. \tag{3.41}$$

Ansatz zur Lösung: $\rho(\zeta) = (\zeta - 1)(\zeta + 5) = (\zeta - \zeta_1)(\zeta - \zeta_2)$ mit NST $\zeta_1 = 1$, $\zeta_2 = -5$. Aus $y_i = \zeta_1^i$ folgt

$$\zeta_1^{i+2} + 4\zeta_1^{i+1} - 5\zeta_1^i = 0 \iff \zeta_1^i \cdot \rho(\zeta_1) = 0.$$

ζ_1^i ist also *spezielle Lösung* der Differenzengleichung (3.41), genauso ζ_2^i. Als allgemeine Lösung setzen wir an $y_i = A\zeta_1^i + B\zeta_2^i$, und damit gilt

$$y_{i+2} + 4y_{i+1} - 5y_i = 0 \quad \iff \quad A\zeta_1^i\rho(\zeta_1) + B\zeta_2^i\rho(\zeta_2) = 0.$$

Die noch freien Parameter $A,\ B$ ergeben sich aus den AW

$$\begin{aligned} y_0 &= A\zeta_1^0 + B\zeta_2^0 & \qquad y_1 &= A\zeta_1^1 + B\zeta_2^1 \\ &= A + B & &= A - 5B \\ \Longrightarrow A &= 1 + \tfrac{h\epsilon}{6}, \quad B = \tfrac{-h\epsilon}{6}. \end{aligned}$$

Die allgemeine Lösung der Rekursion (3.41) lautet somit

$$y_i = 1 + \frac{h\epsilon}{6} + \frac{-h\epsilon}{6} \cdot (-5)^i. \tag{3.42}$$

Diskussion: Im Fall $\epsilon = 0$ liegen die Startwerte auf der exakten Lösung $y(x) \equiv 1$ von (3.40)). Die numerische Lösung $y_i \equiv 1$ stimmt dann mit der exakten überein. Im Fall $\epsilon \neq 0$ ist y_1 leicht gestört. Diese Störung wird duch den Faktor $(-5)^i$ *dramatisch verstärkt – das MSV ist instabil!*

Die im Beispiel diskutierte Problematik tritt bei vielen MSV auf. Verfahren mit scheinbar guten Eigenschaften wie hoher Konsistenzordnung und kleiner Fehlerkonstante liefern schon bei einfachsten Testbeispielen extrem schlecht Resultate.

Zur Analyse des *Stabilitätsproblems* wird wie im Beispiel 4.2 die triviale Testgleichung $y' = 0,\ y(x_0) = 1$ betrachtet. Ein darauf angewandtes MSV liefert die *homogene Rekursion* oder *Differenzengleichung*

$$\alpha_0 y_i + \alpha_1 y_{i+1} + \ldots + \alpha_k y_{i+k} = 0 \tag{3.43}$$

mit Startwerten $y_0, \ldots, y_{k-1}$.

Offensichtlich ist die Folge der y_i über die Rekursion eindeutig festgelegt. Man kann aber noch mehr aussagen, und zwar mit Hife der *Theorie der Differenzengleichungen*. Beachte die Analogie zu linearen Differentialgleichungen k-ter Ordnung!

Satz 3.7
Sei λ eine m-fache Nullstelle des charakteristischen Polynoms $\rho(\zeta)$ aus (3.36), d.h. $\rho(\lambda) = \rho'(\lambda) = \ldots = \rho^{(m-1)}(\lambda) = 0$. Dann gilt:

(i) $y_i^{(1)} = \lambda^i,\ y_i^{(2)} = D\lambda^i = i\lambda^{i-1},\ y_i^{(3)} = D^2\lambda^i = i(i-1)\lambda^{i-2}, \ldots,$
$y_i^{(m)} = D^{m-1}\lambda^i = i(i-1)(i-2)\cdot\ldots\cdot(i-m+2)\lambda^{i-m+1}$
sind spezielle Lösungen der homogenen Rekursion (3.43).

(ii) Die allgemeine Lösung von (3.43) lässt sich als Linearkombination der insgesamt k speziellen Lösungen nach (i) schreiben.

Beweis: Betrachte spezielle Lösung $y_i^{(j)}$ mit $1 \leq j \leq m$,

$$\alpha_0 y_i^{(j)} + \alpha_1 y_{i+1}^{(j)} + \ldots + \alpha_k y_{i+k}^{(j)} = D^{j-1}(\alpha_0\lambda^i + \alpha_1\lambda^{i+1} + \ldots + \alpha_k\lambda^{i+k}) = D^{j-1}(\rho(\lambda)\cdot\lambda^i).$$

Mit der Produktregel nach Leibniz

$$D^l(f \cdot g) = \sum_{j=0}^{l} \binom{l}{s} D^s f D^{l-s} g$$

folgt

$$D^{j-1}(\rho(\lambda)\lambda^i) = \underbrace{\rho(\lambda)}_{=0} D^{j-1}\lambda^i + \ldots + \binom{j-s}{s} \underbrace{D^s\rho(\lambda)}_{=0} D^{l-s}\lambda^i + \ldots + \underbrace{D^{j-1}\rho(\lambda)}_{=0} \lambda^i = 0.$$

Damit ist (i) gezeigt. Offensichtlich gibt es insgesamt k spezielle Lösungen entsprechend dem Grad von ρ, und jede Linearkombination ist ebenfalls Lösung der homogenen Rekursion. Zu zeigen bleibt, dass solch eine Linearkombination durch die Startwerte $y_0, \ldots, y_{k-1}$ eindeutig festgelegt ist. Beweis dazu ist "elementar, aber mühselig" (Stoer/Bulirsch, S. 148) □

Welche Folgerung erlaubt Satz 3.7? Wir haben die Fallunterscheidung:

Falls λ eine NST von ρ ist mit $|\lambda| > 1$:
$\{y_i\} = \{\lambda^i\}$ wächst exponentiell!
Falls λ eine NST von ρ ist mit $|\lambda| < 1$:
$\{y_i\} = \{\lambda^i\}$ fällt exponentiell!
Falls λ eine NST ist mit $|\lambda| = 1$:

$y_i = \lambda^i$ liefert $|y_i| = 1$,
$y_i^{(j)} = i(i-1)(1i-2)\cdot\ldots\cdot(i-j+2)\lambda^i$ wächst polynomial!
(nur falls λj-fache NST mit $j > 1$)

Die homogene Rekursion wird also vollständig durch die Nullstellen des charakteristischen oder erzeugenden Polynoms ρ bestimmt. Dies motiviert

Definition 3.8 *Stabilitätsbedingung MSV*
Ein MSV erfüllt die Stabilitätsbedingung, falls alle NST des charakteristischen Polynoms $\rho(\zeta) = \alpha_k\zeta^k + \ldots + \alpha_1\zeta + \alpha_0$ im abgeschlossenen Einheitskreis von $\mathbb{C}$ liegen und diejenigen auf dem Rand nur einfach sind:

$$\begin{aligned} &\textit{Falls } \rho(\lambda) = 0 \implies |\lambda| \le 1, \\ &\textit{Falls } \left.\begin{array}{rl} \rho(\lambda) & = 0, \\ |\lambda| & = 1 \end{array}\right\} \implies \lambda \textit{ ist einfache NST.} \end{aligned} \tag{3.44}$$

Das MSV heißt stark stabil*: Für alle NST gilt* $|\lambda| < 1$ *außer für* $\lambda = 1$*, ansonsten* schwach stabil*.*

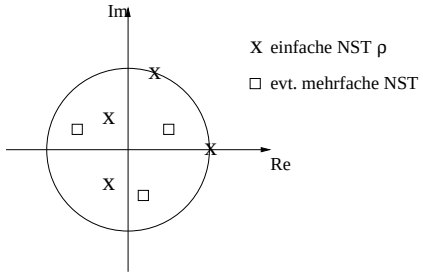


Abbildung 11: Nullstellen des ρ-Polynoms bei Erfülltsein der Stabilitätsbed.

Abb. 11 illustriert die Lage der Nullstellen bei Erfülltsein der Stabilitätsbedingung.

Bei konsistenten MSV ist $\lambda = 1$ immer eine NST von ρ, denn

$$\rho(1) = \alpha_0 + \alpha_1 + \ldots + \alpha_k = 0 \quad \text{nach Satz 4.1}$$

Was bedeutet die Stabilitätsbedingung für die oben eingeführten Verfahrensklassen? Bei Adams–Bashforth und Adams–Moulton ist

$$\alpha_k = 1 \ , \ \alpha_{k-1} = -1 \ , \ \alpha_{k-2} = \ldots = \alpha_0 = 0$$

und damit

$$\rho(\zeta) = \zeta^k - \zeta^{k-1} = \zeta^{k-1}(\zeta - 1) \quad \text{mit NST } \ \lambda = 1 \ \text{ einfach} \ , \ \lambda = 0 \ \ k-1\text{-fach.}$$

Damit sind die Adams-Verfahren stark stabil.

Die BDF-Verfahren sind *stark stabil nur für* $k \leq 6$, ansonsten instabil (o. Bew.).

Die Verfahren nach Nyström und Milne haben Koeffizienten $\alpha_k = 1$, $\alpha_{k-1} = 0$, $\alpha_{k-2} = -1$, $\alpha_{k-3} = \cdots = \alpha_0 = 0$. Daher

$$\rho(\zeta) = \zeta^k - \zeta^{k-2} = \zeta^{k-2}(\zeta^2 - 1) = \zeta^{k-2}(\zeta - 1)(\zeta + 1)$$

mit NST $\zeta_1 = 0, \ \zeta_2 = 1, \ \zeta_3 = -1$. Demnach sind diese MSV nur *schwachstabil.*

Falls das MSV die Stabilitätsbedingung erfüllt, gilt im Übrigen

$$\lim_{i\to\infty} \frac{y_i}{i} = 0$$

für alle Lösungen der homogenen Rekursion (3.43).

Die bisherige Analyse des Stabilitätsproblems kann wie folgt zusammengefasst werden: *Die Konsistenz eines MSV reicht nicht aus, um Konvergenz für die Testgleichung $y' = 0$ zu erzielen. Zusätzlich notwendig ist die Stabilitätsbedingung (3.44)!*

Wie sieht nun der allgemeine Fall, d.h. die Konvergenz für $y' = f(x, y)$, aus? Es zeigt sich, dass die anhand von $y' = 0$ ermittelte Stabilitätsbedingung (3.44) *hinreichend* ist, um die Konvergenz für ein konsistentes Verfahren zu zeigen! Ohne Bew. sei angegeben:

Satz 3.8 *Konvergenz MSV*
Das MSV $\sum \alpha_l y_{i+l} = h \sum \beta_l f_{i+l}$ habe die Konsistenzordnung p und erfülle die Stabilitätsbedingung (2.41). Dann ist es für hinreichend glattes f auch konvergent von der Ordnung p, d.h.

$$e_n(X) = Y(X) - y_n = O(h^p)$$

Kurz und prägnant:

Stabilität + Konsistenz	$\Longleftrightarrow$	Konvergenz

Offen bleibt noch die Frage, welche maximale Konsistenzordnung ein stabiles MSV haben kann. Ohne Berücksichtigung der Stabilitätsbedingung war die maximale Ordnung $2k$ bzw. $2k-1$, s.o. Unter Berücksichtigung gilt der berühmte

Satz 3.9 *Dahlquist–Barriere (1956/59)*
Ein lineares k-Schritt–Verfahren $\sum_{l=0}^{k} \alpha_l y_{i+l} = h \sum_{l=0}^{k} \beta_l f_{i+l}$, *das die Stabilitätsbedingung (3.44) erfüllt, hat maximale Konsistenzordnung*

$$\begin{array}{ll} k+1 & \textit{falls } k \textit{ ungerade} \\ k+2 & \textit{falls } k \textit{ gerade} \end{array}$$

Die Ordnung $k+2$ *kann nur erzielt werden, wenn alle NST des charakteristischen Polynoms* ρ *auf dem Rand des Einheitskreises liegen (schwach stabiles Verfahren).*

Beispiele:
$k = 1$: Maximal $p = 2$, wird erreicht von Adams–Moulton
$y_{i+1} - y_i = \frac{h}{2}(f_{i+1} + f_i)$ Trapezformel
$k = 2$: Maximal $p = 4$, wird erreicht von Milne
(aber nur schwach stabiles Verfahren)

Bemerkungen:

1) Der Beweis der Dahlquist–Barriere verlangt viel funktionentheoretische Hilfsmittel, s. Hairer/Wanner S. 384.

2) Aufgrund der Testgleichung $y' = 0$ spricht man manchmal auch von 0–Stabilität.

Zum Abschluss der MSV noch ein paar Worte zur Implementierung. Im Gegensatz zu den RK–Verfahren ist diese sehr komplex und lässt viel Spielraum für Heuristiken. Die heute verfügbaren Codes gleichen in gewisser Hinsicht “ nervösen Rennpferden”.