Numeriksche Mathematik II

HDoz. Dr. M. Hanke

Universität Karlsruhe

WS 1997/98

Inhaltsverzeichnis

I. Anfangswertaufgaben

1 Einführung

Dieses Kapitel behandelt numerische Verfahren zur Lösung gewöhnlicher Differentialgleichungen der Form

$$y' = f(t,y)\,, \quad y(0) = y_0\,, \qquad t \in [0,T]\,. \tag{1.1}$$

Wegen der Vorgabe von $y(0)$ spricht man bei (1.1) von einem **Anfangswertproblem**.

Die variable Veränderliche t, von der die gesuchte Funktion $y(t)$ abhängt, steht im allgemeinen für die Zeit; man spricht im Zusammenhang mit Anfangswertaufgaben oftmals auch von **Evolutionsprozessen**. Über den zugrundeliegenden Funktionenraum für y ist bislang allerdings noch nichts gesagt; in der Regel soll y mindestens einmal differenzierbar sein. Allerdings braucht y nicht unbedingt skalar zu sein, also $y : [0,T] \to \mathbb{R}$. Ohne größere Schwierigkeiten lassen sich die zu behandelnden Algorithmen auch auf **Systeme** von Differentialgleichungen übertragen: In dem Fall ist y eine vektorwertige Funktion, also $y : [0,T] \to \mathbb{R}^d$.

Beispiel. Ein Standardbeispiel, das uns noch häufiger als "Modellgleichung" begegnen wird, ist die Differentialgleichung

$$y' = \lambda y\,, \quad y(0) = y_0\,, \qquad y_0, \lambda \in \mathbb{R}\,.$$

Hier ist die Funktion $f(t,y) = \lambda y$ von t unabhängig – man spricht in diesem Fall von einer **autonomen** Differentialgleichung. Ihre Lösung ist

$$y(t) = y_0\, e^{\lambda t}\,.$$

Die entsprechende vektorwertige Differentialgleichung lautet

$$y' = Ay\,, \quad y(0) = y_0\,, \qquad A \in \mathbb{R}^{d\times d},\ y_0 \in \mathbb{R}^d\,. \tag{1.2}$$

Ihre Lösung sieht genauso aus, nämlich

$$y(t) = e^{At} y_0\,, \tag{1.3}$$

allerdings handelt es sich hierbei lediglich um eine formale Schreibweise. Der Ausdruck e^{At} ist auf keinen Fall komponentenweise zu verstehen ! Vielmehr wird die rechte Seite von (1.3) über die Potenzreihenentwicklung der Exponentialfunktion definiert:

$$y(t) = \Big(\sum_{k=0}^{\infty} \frac{1}{k!} A^k t^k \Big) y_0 \, .$$

Man beachte hierbei, daß $t \in \mathbb{R}$, $A \in \mathbb{R}^{d \times d}$ und $y_0 \in \mathbb{R}^d$ liegen. Der Ausdruck in der runden Klammer ist also seinerseits eine reelle $d \times d$ Matrix. Die Konvergenz der unendlichen Reihe für jeden möglichen Wert von $t \in \mathbb{R}$ (etwa bezüglich der Spektralnorm $\| \cdot \|_2$) macht man sich leicht wie im eindimensionalen klar. Ebenso ergibt sich durch gliedweises Differenzieren, daß diese unendliche Reihe eine Lösung des Anfangswertproblems darstellt.

Im weiteren sollen verschiedene praktische Anwendungen für dieses wichtige Teilgebiet der numerischen Mathematik angeführt werden.

1.1 Chemische Reaktionskinetik

Wir betrachten das chemische Reaktionsschema

$$A + B \rightleftharpoons C + D \, ,$$

wobei die vier Stoffe A, B, C und D gasförmige Substanzen seien. Grundlage des folgenden mathematischen Modells ist die **Boltzmann'sche kinetische Gastheorie**, nach der bei konstantem Druck, Volumen und Temperatur die Reaktionsgeschwindigkeit proportional zu der Wahrscheinlichkeit ist, daß sich zwei Moleküle der beteiligten Substanzen treffen.

Sind also $c_A, \ldots, c_D$ die Konzentrationen der Gase A bis D bei konstantem Volumen und bezeichnen $\Delta c_A, \ldots, \Delta c_D$ die Konzentrationsänderungen in einem kleinen Zeitintervall $\Delta t > 0$, dann gilt (wenn wir zunächst die Reaktion von rechts nach links ignorieren)

$$\Delta c_A = \Delta c_B = -k_1 c_A c_B \Delta t = -\Delta c_C = -\Delta c_D \, .$$

Hierbei ist k_1 eine positive Proportionalitätskonstante. Durch Grenzübergang $\Delta t \to 0$ werden aus den Konzentrationsänderungen schließlich die Ableitungen $c'_A, \ldots, c'_D$. Finden die beiden Teilreaktionen unabhängig voneinander statt (was wir im weiteren annehmen wollen), dann gilt eine entsprechende Gleichung für die Rückreaktion (mit Proportionalitätskonstante $k_2 > 0$) und die beiden Gleichungen überlagern sich additiv, d.h., es gilt

$$\begin{array}{rcccl} c'_A & = & c'_B & = & -k_1 c_A c_B + k_2 c_C c_D \, , \\ c'_C & = & c'_D & = & k_1 c_A c_B - k_2 c_C c_D \, . \end{array} \tag{1.4}$$

Versehen mit Anfangswerten für die Konzentrationen der Gase zum Zeitpunkt $t = 0$ ergibt (1.4) ein Differentialgleichungssystem von der Form (1.1) mit dem Vektor $y =$

$[c_A, c_B, c_C, c_D]^T \in \mathbb{R}^4$. Ohne Einfluß von außen wird das chemische Verhalten des Gasgemischs für alle Zeiten durch die obigen Gleichungen beschrieben, d.h., für das rechte Ende T des Zeitintervalls kann $T = \infty$ gewählt werden. In der Regel erreicht dabei das Gasgemisch irgendwann einen **Gleichgewichtszustand**, in welchem die Konzentrationen konstant bleiben. Im mathematischen Modell spricht man von dem **stationären Zustand**, dem Grenzzustand für $t \to \infty$. Dort gilt dann $c'_A = c'_B = c'_C = c'_D = 0$, oder äquivalent,

$$\frac{c_A c_B}{c_C c_D} = \frac{k_2}{k_1} .$$

Dies ist das sogenannte **Massenwirkungsgesetz**.

1.2 Himmelsmechanik

Wir bezeichnen mit $x(t) \in \mathbb{R}^3$ die Position eines Körpers x im Weltall (ein Planet, Satellit, etc.). Die Positionsänderung $x'(t)$ gibt dann die Geschwindigkeit und $x''(t)$ die Beschleunigung dieses Körpers an. Ist ferner m die träge Masse des Körpers und $F(t,x)$ die auf den Körper wirkende Kraft zur Zeit t, dann ergibt sich die beschleunigende Wirkung der Kraft zu

$$m\, x''(t) = F(t,x) . \tag{1.5}$$

Ein wichtiges Resultat der Mechanik besagt, daß die Energieerhaltung eines Systems äquivalent zu einer speziellen Form der wirkenden Kraftfelder ist, den sogenannten **Potentialkräften** (daher auch *konservative Kräfte* genannt),

$$F = -\text{grad}\ u .$$

Dabei bezeichnet u das zugehörige Potential.

Für die Himmelsmechanik hat Newton dieses Potential bestimmt: Sind etwa x_1 und x_2 zwei Körper mit Massen m_1 und m_2, dann lautet das **Gravitationspotential**

$$u(x_1, x_2) = -\gamma \frac{m_1 m_2}{\|x_1 - x_2\|_2} ;$$

γ ist die Gravitationskonstante. Durch Einsetzen in die Bewegungsgleichung (1.5) ergibt sich

$$x_1'' = \frac{\gamma\, m_2}{\|x_1 - x_2\|_2^3} (x_2 - x_1) , \qquad x_2'' = \frac{\gamma\, m_1}{\|x_1 - x_2\|_2^3} (x_1 - x_2) .$$

Diese Gleichungen lassen sich analytisch lösen und es ergeben sich die von Kepler vorhergesagten elliptischen Bahnkurven.

Für unsere numerischen Algorithmen hat dieses Differentialgleichungssystem aber noch nicht die gewünschte Form (1.1). Um diese Form zu erhalten führen wir Hilfsvariablen $v_1 = x_1'$ und $v_2 = x_2'$ ein und erhalten dann das System

$$
\begin{aligned}
x_1' &= v_1\,, \\
v_1' &= \frac{\gamma\, m_2}{\|x_1 - x_2\|_2^3}\,(x_2 - x_1)\,, \\
x_2' &= v_2\,, \\
v_2' &= \frac{\gamma\, m_1}{\|x_1 - x_2\|_2^3}\,(x_1 - x_2)\,.
\end{aligned}
$$

Wir sehen, daß für eine sachgemäße Lösung dieses Differentialgleichungssystems Anfangspositionen *und* Anfangsgeschwindigkeiten der beiden Körper bekannt sein müssen.

Beispiel 1.1 Ein interessanter Spezialfall ergibt sich, wenn wir die Bewegung eines Satelliten relativ zur Erde betrachten. Nehmen wir also an, die Masse m_1 des Satelliten sei gegenüber der Erde vernachlässigbar, und legen wir die Position der Erde zu Beginn ruhend in den Nullpunkt unseres Systems, also $x_2(0) = v_2(0) = 0$. Der Grenzübergang $m_1 \to 0$ ergibt weiterhin $v_2' \equiv 0$, so daß wir die Erde aus unseren weiteren Überlegungen ausklammern dürfen. Das Differentialgleichungssystem vereinfacht sich zu

$$
x_1' = v_1\,, \quad v_1' = -\frac{a}{\|x_1\|_2^3}\,x_1\,, \qquad a > 0\,.
$$

Eine weitere Vereinfachung ergibt sich aus der Annahme, daß sich der Satellit senkrecht zur Erde bewegt: In diesem Fall können wir x_1 und v_1 als skalare Größen annehmen. Wir setzen $x_1(0) = 1$ und $v_1(0) = -1$. Man prüft leicht nach, daß dann die Funktion

$$
x_1^-(t) = (1 - \frac{3}{2}\,t)^{2/3}
$$

die Differentialgleichung für $a = 1/2$ löst. Man beachte, daß in diesem Fall der Satellit zur Zeit $t = 2/3$ auf die Erde aufprallt. Seine Geschwindigkeit ist zu diesem Zeitpunkt $-\infty$. In diesem Fall existiert also *keine* Lösung für alle Zeiten T, sondern nur für $T \le 2/3$.

Auf der anderen Seite ergibt sich für die Anfangsgeschwindigkeit $v_1(0) = 1$ die Lösung

$$
x_1^+(t) = (1 + \frac{3}{2}\,t)^{2/3}\,.
$$

Diese Lösung existiert für alle $t > 0$ mit $\lim_{t\to\infty} x_1^+(t) = \infty$ und $\lim_{t\to\infty} v_1^+(t) = 0$. Dieser Satellit schafft also den "Absprung" von der Erde, und zwar mit minimaler Anfangsgeschwindigkeit – weniger wäre zu wenig gewesen. Für $a = 1/2$ ist $v_1(0) = 1$ daher die sogenannte **Fluchtgeschwindigkeit**.

1.3 Wärmeleitungsgleichung

Zum Abschluß sei noch auf den Zusammenhang zwischen partiellen Differentialgleichungen, die Evolutionsprozesse behandeln und gewöhnlichen Differentialgleichungen hingewiesen. Sei

$u(t,x)$, $-1 \le x \le 1$, die Temperaturverteilung zum Zeitpunkt t in einem Stab der Länge $l = 2$. Bei inhomogener Temperatur $u(t,\cdot) \not\equiv \text{const}$ ergibt sich ein Wärmefluß $j(t,x)$ im Stab mit dem Ziel, Temperaturunterschiede auszugleichen (in dem vorliegenden eindimensionalen Fall ist j eine skalare Größe, die angibt, wieviel Wärmeeinheiten in die positive x-Richtung fließen). Dieser Wärmefluß ist proportional zur Ortsableitung $u_x(t,\cdot)$; die Proportionalitätskonstante σ beschreibt die Wärmeleitfähigkeit. Unter Berücksichtigung der Flußrichtung (Vorzeichen !) ergibt sich somit ein Wärmefluß

$$j(t,x) = -\,\sigma\, u_x(t,x)\,, \qquad -1 < x < 1\,. \tag{1.6}$$

Sei nun $I = [a,b] \subset (-1,1)$ ein beliebiger Teil des Stabs. Dann ist

$$W_I(t) := \int_a^b u(t,x)\,dx \tag{1.7}$$

die gesamte Wärmemenge dieses Stabteils zum Zeitpunkt t, während $j(t,b)$ den Wärmeabfluß am rechten Intervallende und entsprechend $j(t,a)$ den Wärmezufluß am linken Intervallende angibt.

Unter der Annahme, daß keine Wärme verloren geht, ergibt sich notwendigerweise die folgende Bilanzgleichung:

$$-\frac{\partial}{\partial t} W_I(t) \;=\; j(t,b) - j(t,a) \;=\; \int_a^b \frac{\partial}{\partial x}\, j(t,x)\,dx\,.$$

Einsetzen von (1.6) und (1.7) ergibt nach Vertauschung von Differentiation und Integration

$$\begin{aligned} 0 \;&=\; \int_a^b u_t(t,x)\,dx + \int_a^b \frac{\partial}{\partial x}\, j(t,x)\,dx \;=\; \int_a^b \Big(u_t(t,x) \;-\; \sigma\, \frac{\partial}{\partial x}\, u_x(t,x) \Big)\,dx \\ &=\; \int_a^b \Big(u_t(t,x) \;-\; \sigma\, u_{xx}(t,x) \Big)\,dx\,. \end{aligned}$$

Da dies für alle beliebig kleinen Teilintervalle I gelten muß, ergibt sich zwangsläufig die Gültigkeit der **Wärmeleitungsgleichung**

$$u_t = \sigma\, u_{xx}\,, \qquad -1 \le x \le 1\,,\; 0 \le t \le T\,. \tag{1.8}$$

Dies ist eine *partielle Differentialgleichung*, die wir im weiteren durch Diskretisierung der Ortsvariablen x in ein System von gewöhnlichen Differentialgleichungen überführen wollen. Dazu setzen wir der Einfachheit halber $\sigma = 1$.

Durch Multiplikation von (1.8) mit einer Funktion $v(x)$ mit $v(-1) = v(1) = 0$ und Integration über den Ort ergibt sich nach partieller Integration

$$\int_{-1}^{1} u_t(t,x)v(x)\,dx = \int_{-1}^{1} u_{xx}(t,x)v(x)\,dx = -\int_{-1}^{1} u_x(t,x)v_x(x)\,dx\,. \tag{1.9}$$

Dies ist die sogenannte **schwache Form** der Differentialgleichung (1.8), die für jede stückweise differenzierbare Funktion v mit $v(-1) = v(1) = 0$ und alle Zeiten $0 \le t \le T$ erfüllt sein muß.

Nehmen wir an, am linken und am rechten Ende des Stabs seien gewisse Randtemperaturen $u_0(t)$ und $u_1(t)$ vorgegeben (zum Beispiel kann der Stab einen Temperaturfühler beschreiben, der links die Temperatur in einem Hochofen mißt und am rechten Ende bei konstanter Temperatur gehalten wird). Dann können wir $u(t,x)$ für jedes t durch einen linearen Spline über einem Gitter $\Delta = \{-1 = x_0 < x_1 < \ldots < x_n = 1\}$ approximieren, d.h.,

$$u(t,x) = \sum_{i=0}^{n} y_i(t) B_i(x) \,, \tag{1.10}$$

wobei B_i wieder die Hutfunktionen aus Paragraph **??** bezeichnen. y_0 und y_n sind bekannt, das sind nämlich gerade die Randtemperaturvorgaben $u_0(t)$ und $u_1(t)$; die restlichen Funktionen $y_i(t)$, $i = 1, \ldots, n-1$, sind zunächst unbekannt.

Durch Einsetzen von (1.10) in (1.9) ergibt sich jedoch ein Differentialgleichungssystem der Form (1.2) für die gesuchten Koeffizientenfunktionen $y_1, \ldots, y_{n-1}$:

$$\sum_{i=0}^{n} y_i'(t) \int_{-1}^{1} B_i(x) v(x)\, dx = -\sum_{i=0}^{n} y_i(t) \int_{-1}^{1} B_i'(x) v_x(x)\, dx \,.$$

Dabei soll v beispielsweise alle Hutfunktionen B_i durchlaufen, die die homogenen Randvorgaben erfüllen, also $v = B_j$ für $j = 1, \ldots, n-1$. Bezeichnet G die innere $(n-1) \times (n-1)$-Untermatrix der Gram'schen Matrix (**??**) aller $n+1$ Hutfunktionen, dann ergibt sich das System

$$Gy' = -Ay + b \qquad \text{mit} \quad A = \left[\langle\, B_i'(x), B_j'(x) \,\rangle \right]_{i,j}$$

und b einem Vektor, der sich aus den bekannten Größen u_0 und u_1 ergibt.

Für ein zulässiges Anfangswertproblem werden noch Anfangswerte für die Funktionen $y_1, \ldots, y_{n-1}$ benötigt, die sich etwa durch Interpolation der Anfangstemperatur $u(0,x)$ im Stab bestimmen lassen.

2 Lösungstheorie

Wir wollen im weiteren annehmen, daß $y \in \mathbb{R}^d$ und daß die Funktion f in einem offenen Rechteck $\Omega = I \times J$ definiert ist mit $(0,T) \subset I$ und $J \subset \mathbb{R}^d$. Dabei darf J auch ein unbeschränktes Intervall sein.

Grundlegend für die folgenden Überlegungen ist der **Existenzsatz von Picard-Lindelöf**:

Satz 2.1 *f sei stetig in Ω und für alle kompakten Teilmengen $K \subset \Omega$ gelte eine (lokale) Lipschitzbedingung der Form*

$$\|f(t,y) - f(t,z)\|_2 \le L_K \|y - z\|_2 \qquad \textit{für alle } (t,y),(t,z) \in K \,. \tag{2.1}$$

Dann existiert für jedes $y_0 \in J$ ein nichtleeres Teilintervall $I_0 \subset I$ mit $0 \in \overline{I_0}$ und eine eindeutig bestimmte stetig differenzierbare Lösung $y : I_0 \to J$ des Anfangswertproblems (1.1). Die Lösungskurve $(t, y(t))$ hat zudem eine eindeutig bestimmte Fortsetzung bis an den Rand des Rechtecks Ω.

Hinreichend für die Gültigkeit der lokalen Lipschitzbedingung (2.1) ist etwa, daß f in Ω stetig differenzierbar ist. Dies folgt unmittelbar aus der mehrdimensionalen Verallgemeinerung des Mittelwertsatzes, wie er bereits im Zusammenhang mit dem Banachschen Fixpunktsatz im $\mathbb{R}^n$ verwendet wurde. Der Banachsche Fixpunktsatz wird auch zum Beweis des Satzes von Picard-Lindelöf eingesetzt; dabei ist die Wahl der zu verwendenden Norm entscheidend; vgl. W. Walter, Gewöhnliche Differentialgleichungen, Springer Verlag.

Ist $I = (0, T)$ das maximale Intervall, in dem f den Voraussetzungen des Satzes 2.1 genügt, dann folgt, daß entweder eine eindeutig bestimmte Lösung der Differentialgleichung im gesamten Intervall $[0, T)$ existiert oder daß die Lösung im Innern des Intervalls gegen ∂J konvergiert (etwa für $t \to t_0 \in (0, T)$); ist $J = \mathbb{R}^d$ und die Funktion f gleichmäßig beschränkt in $I \times J$, dann macht man sich leicht klar, daß die Lösung y in dem gesamten Intervall $[0, T]$ wohldefiniert ist.

Beispiel. Beide Fälle traten in Beispiel 1.1 auf: Die Differentialgleichung

$$y' = u\,, \quad u' = -\frac{1}{2}\, y^{-2}\,,$$

erfüllt die lokale Lipschitz-Bedingung des Satzes im Rechteck $\Omega = \mathbb{R}^+ \times (\mathbb{R}^+ \times \mathbb{R})$. Die Lösung der Differentialgleichung existiert somit in eindeutiger Weise, solange y nicht Null wird, also solange in diesem Beispiel der Satellit nicht auf die Erde stürzt. Bei der Anfangsvorgabe $y(0) = 1$, $u(0) = -1$ war das für $t = 2/3$ der Fall. In diesem Moment erreicht die Lösungskurve den Rand des Rechtecks Ω. Für $y(0) = 1$ und $u(0) = 1$ existiert hingegen eine eindeutige Lösung im gesamten Zeitintervall $(0, \infty)$.

Die nächste Frage ist die nach der stetigen Abhängigkeit der Lösung.

Satz 2.2 *f sei stetig und erfülle die Ungleichung*

$$\langle\, f(t,y) - f(t,z), y - z \,\rangle \le l\, \|y - z\|_2^2 \qquad \textit{für alle } (t,y),(t,z) \in \Omega\,. \tag{2.2}$$

Ferner seien $y, z : I \to J$ Lösungen der Differentialgleichungen $y' = f(t,y)$ und $z' = f(t,z)$ mit verschiedenen Anfangswerten $y_0, z_0 \in J$. Dann gilt

$$\|y(t) - z(t)\|_2 \le e^{lt}\, \|y_0 - z_0\|_2 \qquad \textit{für alle } t \in I\,.$$

Beweis. Da der Beweis sehr einfach ist, soll er hier vorgeführt werden. Wir wählen ein beliebiges $t_0 \in I$ aus und nehmen oBdA an, daß $y(t_0) \neq z(t_0)$. Wegen der Stetigkeit der

beiden Lösungen ist die Funktion $x(t) := \|y(t) - z(t)\|_2^2$ in einer Umgebung um t_0 positiv und daher die Funktion $\log x(t)$ dort wohldefiniert. Dort gilt

$$\begin{aligned} x'(t) &= \frac{d}{dt}\,\|y(t) - z(t)\|_2^2 \;=\; 2\,\langle\, y'(t) - z'(t), y(t) - z(t)\,\rangle \\ &= 2\,\langle\, f(t,y(t)) - f(t,z(t)), y(t) - z(t)\,\rangle \;\le\; 2l\,\|y(t) - z(t)\|_2^2 \\ &= 2l\,x(t)\,, \end{aligned}$$

und daher ist

$$\frac{d}{dt}\,\log x(t) \;=\; \frac{x'(t)}{x(t)} \;\le 2l\;.$$

Aufintegrieren von t bis t_0 ergibt dann

$$\log x(t_0) - \log x(t) \le 2l(t_0 - t)\,,$$

bzw.

$$x(t_0) \le x(t)\,e^{2l(t_0 - t)}\,.$$

Diese Ungleichung gilt für jedes beliebige t aus dem größtmöglichen offenen Intervall um t_0, in dem $x(t)$ positiv bleibt. Da aber $x(t_0)$ nach unserer Annahme positiv ist, kann es kein $t \in [0, t_0)$ mit $x(t) = 0$ geben, und daher kann in obiger Ungleichung der Grenzübergang $t \to 0$ durchgeführt werden. □

Aus Satz 2.2 folgt insbesondere, daß unter der Bedingung (2.2) Lösungen des Anfangswertproblems $y' = f(t,y)$ mit $y(0) = y_0 \in J$ eindeutig bestimmt sind.

Die Voraussetzung (2.2) ist gleichzeitig schwächer und stärker als die des Satzes von Picard-Lindelöf. Sie ist schwächer, da aus einer Lipschitz-Bedingung

$$\|f(t,y) - f(t,z)\|_2 \le L\,\|y - z\|_2 \qquad \text{für alle } (t,y),(t,z) \in \Omega$$

sofort die Bedingung (2.2) des Satzes mit $l = L$ folgt. Wir sprechen daher in Zukunft bei (2.2) von einer **schwachen Lipschitz-Bedingung**. Andererseits ist die schwache Lipschitz-Bedingung stärker als die Voraussetzung von Satz 2.1, da die Bedingung (2.2) *gleichmäßig* für alle Punkte in Ω benötigt wird.

Die Abschwächung gegenüber dem Satz von Picard-Lindelöf hat jedoch den entscheidenden Vorteil, daß negative l in der Abschätzung von Satz 2.2 möglich sind, während L zwangsläufig immer positiv ist. Differentialgleichungen, die einer schwachen Lipschitz Bedingung (2.2) mit einem negativen l genügen, nennt man **strikt dissipativ**.

Beispiel. Die Differentialgleichung $y' = \lambda y$, $y(0) = y_0$ hat die Lösung $y(t) = y_0 e^{\lambda t}$. Wegen

$$\langle\, f(t,y) - f(t,z), y - z\,\rangle = \lambda\,\|y - z\|_2^2$$

ist die Voraussetzung von Satz 2.2 für $l = \lambda$ in ganz $\mathbb{R}^+ \times \mathbb{R}^d$ erfüllt. Für negative Werte von λ werden Fehler in den Startwerten also mit dem Faktor $e^{\lambda t}$ gedämpft; alle Lösungen laufen gegen Null (**asymptotisch stabil**). Für $\lambda > 0$ werden Fehler in den Startwerten verstärkt; die Lösungen sind instabil.

Satz 2.2 besagt, daß die Zuordnung $y_0 \mapsto y(t)$ stetig ist, genauer Lipschitz-stetig mit Lipschitz-Konstante e^{lt}. Wir können daher in Ω die Größe $\kappa = e^{l\Delta t}$ als ein Maß für die lokale Fehlerverstärkung des absoluten Datenfehlers ansehen. κ übernimmt die Rolle einer absoluten (lokalen) **Konditionszahl** der Abbildung

$$y_0 \mapsto y(t), \qquad 0 \le t \le \Delta t\,.$$

3 Das Euler-Verfahren

Als erstes numerisches Verfahren zur Lösung von (1.1) betrachten wir das klassische **Euler-Verfahren**. Dazu beachte man, daß die Funktion f in jedem Punkt $(t, y) \in \Omega$ die Steigung der Lösungskurve definiert. Sobald eine Lösungskurve $(t, y(t))$ durch diesen Punkt läuft, hat die Tangente an die Kurve in diesem Punkt die Steigung $f(t, y(t))$.

Das Euler-Verfahren (auch **Polygonzugverfahren**) macht sich diesen Sachverhalt wie folgt zunutze. In einem vorgegebenen Gitter $\Delta = \{0 = t_0 < t_1 < \ldots < t_n\} \subset I$ wird derjenige lineare Spline $y_\Delta \in S^d_{1,\Delta}$ als Approximation an y gewählt, dessen *rechtsseitige Ableitung* in dem jeweiligen Gitterknoten mit der vorgegebenen Steigung $f(t, y(t))$ übereinstimmt.

Da durch y_0 und $f(0, y_0)$ am linken Rand der Funktionswert und die Anfangssteigung des Splines festgelegt sind, lassen sich die Koeffizienten $y_i \in \mathbb{R}^d$ des Splines $y_\Delta(t) = \sum_{i=0}^n y_i B_i(t)$ in **expliziter** Weise rekursiv von links nach rechts bestimmen:

$$y_{i+1} = y_i + (t_{i+1} - t_i) f(t_i, y_i)\,.$$

Beispiel. $y' = y$, $y(0) = 1$; die exakte Lösung ist $y(t) = e^t$. Mit dem Euler-Verfahren ergibt sich in einem äquidistanten Gitter $(t_i = ih)$

$$\begin{aligned} y_0 &= 1\,, \qquad y_1 = 1 + h \cdot 1 = 1 + h\,,\\ y_2 &= 1 + h + h(1+h) = (1+h)^2\,,\\ y_3 &= (1+h)^2 + h(1+h)^2 = (1+h)^3\,. \end{aligned}$$

Man sieht leicht durch Induktion, daß $y_n = (1+h)^n$. Für $t_n = T$ (fest) bedeutet dies

$$y_n = \left(1 + \frac{T}{n}\right)^n \longrightarrow e^T = y(T)\,, \qquad n \to \infty\,.$$

Für die folgende Fehlerabschätzung beschränken wir uns auf äquidistante Gitter Δ mit konstanter Gitterweite $h = t_{i+1} - t_i$, $i = 0, \ldots, n$.

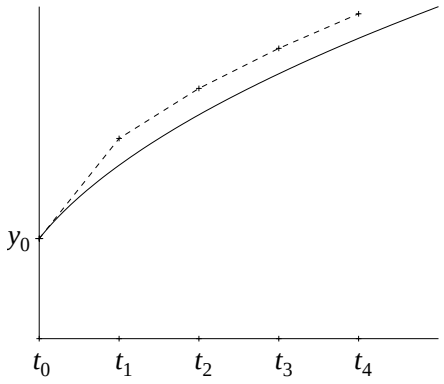


Fig. 3.1. Euler-Polygonzugverfahren

Satz 3.1 *Sei $I = [0,T]$ und $f : I \times \mathbb{R}^d \to \mathbb{R}^d$ stetig differenzierbar und bezüglich y global Lipschitz stetig,*

$$\|f(t,y) - f(t,z)\|_2 \le L\|y - z\|_2 \qquad \text{für alle } t \in I \text{ und } y, z \in \mathbb{R}^d .$$

Ist dann y die eindeutig bestimmte Lösung des Anfangswertproblems (1.1) und sind y_i, $i = 1, \ldots, n$, die Näherungen des Euler-Polygonzugverfahrens an den Gitterpunkten $t_i \in I$, dann gilt

$$\|y(t_i) - y_i\|_2 \;\le\; \frac{(1+Lh)^i - 1}{2L}\,\|y''\|_{[0,T]}\,h \;\le\; \frac{e^{LT}-1}{2L}\,\|y''\|_{[0,T]}\,h\,, \qquad i = 0, \ldots, n\,.$$

Hierbei sei $\|y''\|_{[0,T]} = \max_{0\le t\le T}\|y''\|_2$ (beachte: $y'' \in \mathbb{R}^d$).

Beachte: Die Voraussetzungen an f garantieren, daß y zweimal stetig differenzierbar ist, denn es gilt

$$y'' = \underbrace{\frac{\partial f}{\partial t}}_{\in \mathbb{R}^d} + \underbrace{\frac{\partial f}{\partial y}}_{\in \mathbb{R}^{d\times d}} y' = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y} f\,.$$

Beweis. Der Beweis ist in drei Schritte gegliedert.

1. Lokaler Fehler: Nehmen wir zunächst an, zur Zeit t_i würde das Polygonzugverfahren auf dem Punkt $\big(t_i, y(t_i)\big)$ auf der exakten Lösungskurve starten und ausgehend von $y(t_i)$ eine Approximation z_{i+1} für $y(t_{i+1})$ berechnen. Dann ergibt sich der absolute Fehler

$$\|y(t_{i+1}) - z_{i+1}\| \;=\; \big\| y(t_{i+1}) - \big(y(t_i) + hf(t_i, y(t_i))\big) \big\| \underset{Dgl.}{=} \big\| y(t_{i+1}) - y(t_i) - hy'(t_i) \big\|$$

$$\underset{Hauptsatz}{=} \Big\| \int_{t_i}^{t_{i+1}} \big(y'(\tau) - y'(t_i)\big)\, d\tau \Big\| \underset{MWS}{\le} \|y''\|_{[0,T]} \int_{t_i}^{t_{i+1}} (\tau - t_i)\, d\tau \;=\; \frac{1}{2}\,\|y''\|_{[0,T]}\,h^2\,.$$

2. Lokale Fehlerfortpflanzung: Tatsächlich ist das Verfahren nach i Schritten nicht auf der exakten Lösungskurve, sondern hat statt dessen eine Näherung y_i von $y(t_i)$ berechnet. Daher

müssen wir noch untersuchen, wie der Fehler $y_i - y(t_i)$ im $(i+1)$-ten Schritt fortgepflanzt wird. Mit der Rechenvorschrift des Eulerverfahrens ergibt sich

$$y_{i+1} = y_i + hf(t_i, y_i)\,, \qquad \text{bzw.} \qquad z_{i+1} = y(t_i) + hf(t_i, y(t_i))\,.$$

Nach diesem Schritt ist also

$$\|y_{i+1} - z_{i+1}\|_2 \;\le\; \|y_i - y(t_i)\|_2 + h\|f(t_i, y_i) - f(t_i, y(t_i))\|_2 \;\le\; (1+hL)\,\|y_i - y(t_i)\|_2\,. \tag{3.1}$$

3. Kumulierter Fehler: Im weiteren suchen wir eine Oberschranke für die Norm ε_i des Gesamtfehlers nach i Zeitschritten. Ziel und Aussage des Satzes ist die Ungleichung

$$\varepsilon_i \le \frac{(1+Lh)^i - 1}{2L}\,\|y''\|_{[0,T]}\,h\,, \qquad i = 0, \ldots, n\,, \tag{3.2}$$

die für $i = 0$ wegen des exakt vorgegebenen Anfangswerts natürlich erfüllt ist. Aus den ersten beiden Beweisschritten ergibt sich (mit den gleichen Bezeichnungen wie oben) mit der Dreiecksungleichung für die $(i+1)$-te Fehlergröße induktiv die Ungleichung

$$\begin{aligned}
\|y_{i+1} - y(t_{i+1})\| &\le \|y_{i+1} - z_{i+1}\| \;+\; \|z_{i+1} - y(t_{i+1})\| \\
&\le (1+hL)\varepsilon_i \;+\; \frac{1}{2}\,\|y''\|_{[0,T]}\,h^2 \qquad (3.3)\\
&\underset{(3.2)}{\le} \frac{1}{2L}\,\big((1+hL)^{i+1} - 1 - hL + hL\big)\,\|y''\|_{[0,T]}\,h \\
&= \frac{(1+hL)^{i+1} - 1}{2L}\,\|y''\|_{[0,T]}\,h\,,
\end{aligned}$$

was zu zeigen war. Wegen $1 + hL \le e^{hL}$ und $t_i = ih \in (0, T]$ folgt daraus unmittelbar auch die zweite Behauptung. □

Aus dieser Fehlerabschätzung folgt, daß der Fehler des Euler-Verfahrens linear in h gegen Null geht, falls das Gitter sukzessive verfeinert wird. Diese Verfeinerung wird aber in dem Moment kritisch, in dem der jeweilige Rechenfehler die Größenordnung des lokalen Fehlers erreicht. Eine heuristische Überlegung mag das belegen: Nehmen wir an, im $(i+1)$-ten Schritt kommt zu den bereits untersuchten Fehlern (lokaler Fehler und fortgepflanzter Fehler) noch ein additiver Rundefehler der Größenordnung ϵ, also der Maschinengenauigkeit hinzu. Dann erhalten wir anstelle von (3.3) die Ungleichung

$$\varepsilon_{i+1} \;\le\; (1+hL)\varepsilon_i \;+\; \frac{1}{2}\,\|y''\|_{[0,T]}\,h^2 \;+\epsilon\,,$$

und induktiv ergibt sich entsprechend

$$\varepsilon_i \le \frac{e^{Lih} - 1}{2L}\Big(\|y''\|_{[0,T]}\,h + 2\,\frac{\epsilon}{h}\Big)\,, \qquad i = 0, \ldots, n\,. \tag{3.4}$$

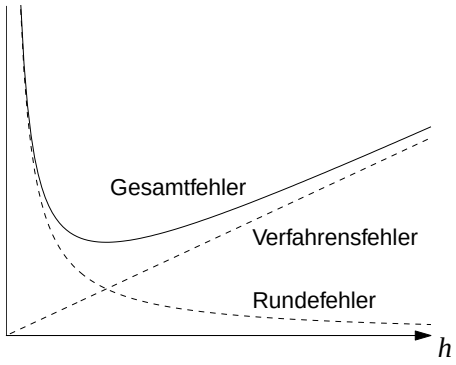


Fig. 3.2. Verfahrensfehler und Rundefehler

Mit anderen Worten: Der Gesamtfehler des Euler-Verfahrens setzt sich aus einem (für $h \to 0$ konvergenten) Verfahrensfehler und einem (für $h \to 0$ divergenten) fortgepflanztem Rundefehler zusammen. Man sieht leicht, daß die Schranke auf der rechten Seite von (3.4) für $h \sim \sqrt{\epsilon}$ ihren minimalen Wert von der Größenordnung $\sqrt{\epsilon}$ annimmt.

Bemerkung 3.2 Es macht also keinen Sinn, Schrittweiten zu wählen, die kleiner als $\sqrt{\epsilon}$ sind, falls ϵ die Rechengenauigkeit in einem Zeitschritt ist. Im gerade betrachteten Fall war ϵ die Maschinengenauigkeit und $\sqrt{\epsilon}$ etwa die halbe, zur Verfügung stehende Mantissenlänge (relativ zur Größe von y). Später werden wir hierauf zurückkommen und dann ϵ andere Bedeutungen zuordnen.

4 Das implizite Euler-Verfahren

Im Gegensatz zum vorangegangenen Abschnitt wollen wir nun annehmen, daß die rechte Seite $f(t,y)$ der Differentialgleichung einer schwachen Lipschitz-Bedingung

$$\langle\, f(t,y) - f(t,z), y - z \,\rangle \le l \,\|y - z\|_2^2 \qquad \text{für alle } (t,y),(t,z) \in \Omega \tag{4.1}$$

genügt. Wie wir am Ende von Paragraph 2 gesehen haben, ergibt sich dann $\kappa = e^{l\Delta t}$ als Konditionszahl für die lokale Abhängigkeit der Lösung vom Anfangswert.

Andererseits ergibt sich für die Näherung des Euler-Verfahrens in Analogie zu (3.1), daß

$$\begin{aligned} \|y_1 - z_1\|^2 &= \| \,(y_0 - z_0) + h\big(f(t,y_0) - f(t,z_0)\big)\,\|^2 \\ &= \|y_0 - z_0\|^2 + 2h\,\langle\, f(t,y_0) - f(t,z_0), y_0 - z_0 \,\rangle + h^2 \|f(t,y_0) - f(t,z_0)\|^2 \\ &\le (1 + 2hl)\,\|y_0 - z_0\|^2 + h^2\,\|f(t,y_0) - f(t,z_0)\|^2 . \end{aligned}$$

Beachtet man noch, daß (nach der Cauchy-Schwarz Ungleichung)

$$\|f(t,y)-f(t,z)\| \;=\; \sup_{w\neq 0}\frac{\langle\, f(t,y)-f(t,z),w\,\rangle}{\|w\|} \;\geq\; \max\Big\{\pm\frac{\langle\, f(t,y)-f(t,z),y-z\,\rangle}{\|y-z\|}\Big\}\,,$$

dann ergibt sich als gute Schätzung für den lokalen fortgepflanzten Datenfehler:

$$\|y_1-z_1\|^2 \lesssim (1+2hl+h^2l^2)\,\|y_0-z_0\|^2 = (1+hl)^2\,\|y_0-z_0\|^2\,.$$

Für $h=\Delta t$ ist also $\kappa_E \approx |1+l\Delta t|$ unter den genannten Voraussetzungen eine gute Approximation der Konditionszahl des Euler-Verfahrens. Zwei Fälle gilt es nun zu unterscheiden: Ist l positiv, dann ist $\kappa_E<\kappa$ und das Eulerverfahren kann als *vorwärts stabil* betrachtet werden. Ist hingegen l negativ, dann sind κ_E und κ nur dann von vergleichbarer Größenordnung, falls $|l|\Delta t = O(1)$, denn ansonsten ist $\kappa_E \gg 1 > \kappa$.

Die Bedingung $|l|\Delta t = O(1)$ kann umgekehrt als Anforderung an die Schrittweite $h=\Delta t$ verstanden werden. Demnach ist das Euler-Verfahren bei negativem l in aller Regel nur dann stabil, wenn $h \lesssim 1/|l|$; für stark negative l führt das zu einem unzulässig hohem Arbeitsaufwand. Differentialgleichungen, für die dies von Bedeutung ist, werden **steife Differentialgleichungen** genannt, vgl. Abschnitt 7 für Beispiele.

Bei steifen Differentialgleichungen wählt man besser eine Variante des Euler-Verfahrens, die als **implizites Euler-Verfahren** bekannt ist. Wieder approximiert man die exakte Lösung durch einen linearen Spline, doch im Unterschied zum expliziten Euler-Verfahren fordert man nun, daß die *linksseitige Ableitung* des Splines in jedem Gitterknoten mit dem Wert von $f(t_i,y_i)$ übereinstimmt. Wie der Name des Verfahrens allerdings bereits andeutet, kann die Bestimmung dieses Splines nicht mehr explizit erfolgen; statt dessen ergibt sich y_{i+1} aus

$$y_{i+1} = y_i + hf(t_{i+1},y_{i+1})\,. \tag{4.2}$$

In jedem Schritt ist daher zur Bestimmung von y_{i+1} ein (i.A. nichtlineares) Gleichungssystem zu lösen.

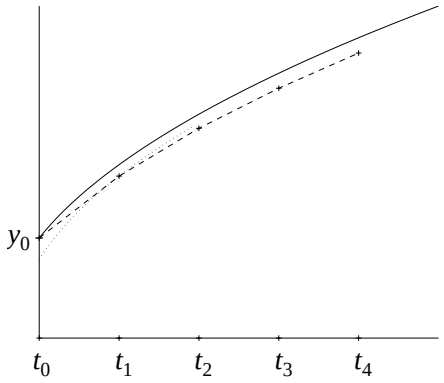


Fig. 4.1. Implizites Euler-Verfahren

Beispiel. Betrachte $y'=\lambda y$, $y(0)=1$, mit $\lambda<0$; die exakte Lösung ist $y(t)=e^{\lambda t}$. (4.2) hat in diesem Fall die einfache Form

$$y_{i+1} = y_i + h\lambda y_{i+1} \quad \rightsquigarrow \quad y_{i+1} = \frac{1}{1-h\lambda}\, y_i\,, \tag{4.3}$$

also

$$y_n = \left(\frac{1}{1-h\lambda}\right)^n .$$

Mit $T = nh$ ergibt sich

$$y_n = \left(1 - \frac{T}{n}\lambda\right)^{-n} \longrightarrow e^{\lambda T} = y(T)\,, \qquad n \to \infty\,.$$

In (4.3) erkennt man, daß die lokale Fehlerverstärkung in einem Zeitschritt durch $\kappa_{IE} = (1-\lambda h)^{-1} \in (e^{\lambda h}, 1) = (\kappa, 1)$ gegeben ist (beachte: $\lambda < 0$) und diese Gleichung gilt *unabhängig* von der Größe von h. Das Verfahren ist also für alle $h > 0$ gut konditioniert und für moderate Zeitschritte (vorwärts) stabil, falls $\lambda < 0$ ist.

Dieses Beispiel ist typisch für die allgemeine Situation. Bevor wir jedoch die Konvergenz des impliziten Euler-Verfahrens untersuchen, diskutieren wir zunächst die Lösbarkeit der nichtlinearen Gleichung (4.2).

Satz 4.1 *Sei $I = [0,T]$, und $f : I \times \mathbb{R}^d \to \mathbb{R}^d$ sei stetig differenzierbar und erfülle die schwache Lipschitz-Bedingung* (4.1) *für ein $l \in \mathbb{R}$. Dann existiert für jedes $y \in \mathbb{R}^d$ und jedes $t \in (0,T]$ eine eindeutige Lösung Y der nichtlinearen Gleichung*

$$Y = y + hf(t,Y)\,, \tag{4.4}$$

vorausgesetzt, daß $hl < 1$ ist.

Beweis. Lösungen der Gleichung (4.4) sind offensichtlich Nullstellen der Funktion

$$F(Y) := y + hf(t,Y) - Y\,.$$

Dabei erfüllt die Funktion F ebenfalls eine schwache Lipschitz-Bedingung:

$$\begin{aligned} \langle\, F(Y) - F(Z), Y - Z \,\rangle &= h\,\langle\, f(t,Y) - f(t,Z), Y - Z \,\rangle \;-\; \|Y - Z\|_2^2 \\ &\leq -\underbrace{(1-hl)}_{>0}\, \|Y - Z\|_2^2\,. \end{aligned}$$

Hieraus folgt bereits unmittelbar, daß F höchstens eine Nullstelle Y haben kann, also die Eindeutigkeit der Lösung Y von (4.4).

Nullstellen Y der Funktion F sind automatisch auch stationäre Lösungen, $u(t) = Y$ für alle $t > 0$, der Differentialgleichung

$$u' = F(u) \tag{4.5}$$

und umgekehrt. Mit f ist auch F stetig differenzierbar und daher insbesondere lokal Lipschitz-stetig. Also existieren zu jedem $u_0, v_0 \in \mathbb{R}^d$ Lösungen u und v der Differentialgleichung (4.5) mit Anfangswerten $u(0) = u_0$, bzw. $v(0) = v_0$. Ferner gilt nach Satz 2.2 für beliebiges $t_0 > 0$ die Ungleichung

$$\|u(t_0) - v(t_0)\|_2 \;\le\; \underbrace{e^{-(1-hl)t_0}}_{=:\,q\,<\,1} \|u_0 - v_0\|_2 \,. \tag{4.6}$$

Daher ist die Abbildung $u_0 \mapsto u(t_0)$ eine kontrahierende Selbstabbildung des $\mathbb{R}^d$, und nach dem Banachschen Fixpunktsatz existiert ein eindeutiger Fixpunkt dieser Abbildung, den wir mit Y bezeichnen wollen. Beachte: Für verschiedene t_0 können sich prinzipiell verschiedene Y ergeben !

Da die Differentialgleichung (4.5) autonom ist, ist u zwangsläufig t_0-periodisch: Mit $v(t) = u(t + t_0)$ gilt nämlich

$$v'(t) = u'(t + t_0) = F\big(u(t + t_0)\big) = F\big(v(t)\big) \,,$$

also sind u und v beides Lösungen von (4.5) mit gleichem Anfangswert $u(0) = v(0) = u(t_0) = Y$. Demnach stimmen u und v überein, d.h. $u(t) = u(t + t_0)$.

Genauso sieht man, daß für festes $t_1 > 0$ die Funktion $v_1(t) = u(t + t_1)$ eine Lösung der Differentialgleichung (4.5) mit Anfangswert $v_1(0) = u(t_1)$ ist; mit u ist dann natürlich auch v_1 t_0-periodisch. Also folgt aus Satz 2.2:

$$\begin{aligned} \|u(t_1) - Y\|_2 \quad &= \quad \|v_1(0) - u(0)\|_2 \;=\; \|v_1(t_0) - u(t_0)\|_2 \\ &\underset{(4.6)}{\le} \quad e^{-(1-hl)t_0} \|v_1(0) - u(0)\|_2 \;=\; q\,\|u(t_1) - Y\|_2 \,. \end{aligned}$$

Da $q < 1$ ist, muß also $u(t_1) = Y$ sein. Da aber $t_1 > 0$ beliebig war, ergibt sich $u \equiv Y$ und damit ist Y die gesuchte Nullstelle von F. □

Bemerkung. Die Bedingung $lh < 1$ ist insbesondere dann erfüllt, wenn l negativ ist (steife Differentialgleichungen) – unabhängig von der Größe der Konstanten L in einer *starken* Lipschitz-Bedingung. Dafür ergeben sich Einschränkungen an die Schrittweite bei positiven Werten von l.

Nun zu dem angekündigten Konvergenzsatz.

Satz 4.2 *Es gelten die Voraussetzungen von Satz 4.1 an f und für das besagte l aus* (4.1) *gelte die Schrittweitenbedingung $hl < 1$. Dann gilt für das implizite Euler-Verfahren die Fehlerabschätzung*

$$\|y(t_i) - y_i\|_2 \;\le\; \frac{1}{2l}\Big(\big(\frac{1}{1-lh}\big)^i - 1\Big)\,\|y''\|_{[0,T]}\,h \,, \qquad t_i = ih \in I \,. \tag{4.7}$$

Beweis. Der Aufbau des Beweises ist in weiten Strecken ähnlich zum Beweis von Satz 3.1.

1. Lokaler Fehler: Wir betrachten zunächst einen Schritt des impliziten Euler-Verfahrens, ausgehend von einem Punkt $\big(t_i, y(t_i)\big)$ auf der exakten Lösungskurve. Mit dem Satz von Taylor,

$$y(t_i) = y(t_{i+1}) - hy'(t_{i+1}) + r_i\,, \qquad \|r_i\|_2 \le \frac{1}{2}\,\|y''\|_{[0,T]} h^2\,,$$

ergibt sich für die nächste Approximation z_{i+1} der Fehler

$$\begin{aligned} z_{i+1} - y(t_{i+1}) &= y(t_i) + hf(t_{i+1}, z_{i+1}) - y(t_{i+1}) \\ &= y(t_{i+1}) - hy'(t_{i+1}) + r_i + hf(t_{i+1}, z_{i+1}) - y(t_{i+1}) \\ &= h\big(f(t_{i+1}, z_{i+1}) - f(t_{i+1}, y(t_{i+1}))\big) + r_i\,. \end{aligned}$$

Nach Multiplikation mit $z_{i+1} - y(t_{i+1})$ ergibt sich aus der schwachen Lipschitz-Bedingung dann die Ungleichung

$$\|z_{i+1} - y(t_{i+1})\|_2^2 \;\le\; lh\,\|z_{i+1} - y(t_{i+1})\|_2^2 \;+\; \|r_i\|_2\,\|z_{i+1} - y(t_{i+1})\|_2\,,$$

und daraus folgt schließlich

$$\|z_{i+1} - y(t_{i+1})\|_2 \;\le\; \frac{1}{2(1-lh)}\,\|y''\|_{[0,T]}\,h^2\,. \tag{4.8}$$

2. Lokale Fehlerfortpflanzung: Ausgehend von y_i, bzw. $y(t_i)$ ergeben sich im $(i+1)$-ten Schritt zwei verschiedene Näherungen für $y(t_{i+1})$,

$$y_{i+1} = y_i + hf(t_{i+1}, y_{i+1})\,, \qquad \text{bzw.} \qquad z_{i+1} = y(t_i) + hf(t_{i+1}, z_{i+1})\,.$$

Dabei ist

$$y_{i+1} - z_{i+1} = h\big(f(t_{i+1}, y_{i+1}) - f(t_{i+1}, z_{i+1})\big) \;+\; \big(y_i - y(t_i)\big)\,,$$

und wie im ersten Beweisschritt ergibt sich nach Multiplikation mit $y_{i+1} - z_{i+1}$ schließlich

$$\|y_{i+1} - z_{i+1}\|_2 \;\le\; \frac{1}{1-lh}\,\|y_i - y(t_i)\|_2\,. \tag{4.9}$$

3. Kumulierter Fehler: Für den Gesamtfehler ε_i nach i Zeitschritten ergibt sich daher beim impliziten Euler-Verfahren die Rekursion

$$\varepsilon_{i+1} \;\le\; \frac{1}{1-lh}\,\varepsilon_i \;+\; \frac{1}{2(1-lh)}\,\|y''\|_{[0,T]}\,h^2\,.$$

Die Behauptung (4.7) folgt nun wieder durch einen einfachen Induktionsbeweis. □

Daraus folgt unmittelbar das

Korollar 4.3 *Es gelten die Voraussetzungen von Satz 4.2 mit einem $l < 0$. Dann gilt für alle $t_i \in [0,T]$ die Abschätzung*

$$\|y(t_i) - y_i\|_2 \leq \frac{1}{2|l|} \|y''\|_{[0,T]} h \,.$$

Für eine effiziente Implementierung des impliziten Euler-Verfahrens bilden die nichtlinearen Gleichungssysteme (4.2) das Hauptproblem. Sie können beispielsweise mit dem Newton-Verfahren gelöst werden:

$$y_{i+1}^{(n+1)} = y_{i+1}^{(n)} - \left(I - hf_y(t_{i+1}, y_{i+1}^{(n)})\right)^{-1} \left(y_{i+1}^{(n)} - y_i - hf(t_{i+1}, y_{i+1}^{(n)})\right), \qquad k = 0, 1, 2, \ldots .$$

Wegen der sehr aufwendigen Berechnung der **Jacobi-Matrix** $J = f_y(t_{i+1}, y)$ und der Inversen von $I - hJ$ ersetzt man das Newton-Verfahren im allgemeinen durch ein sogenanntes **Quasi-Newton-Verfahren**, bei dem in der Jacobi-Matrix immer die Näherung $y = y_i$ aus dem vorangegangenen Zeitschritt eingesetzt wird. Dadurch muß in jedem Zeitschritt nur eine Jacobi-Matrix aufgebaut werden.

Meist sind wenige (ein bis drei) Iterationsschritte ausreichend, um eine Genauigkeit $\|y_{i+1}^{(n)} - y_{i+1}\|_2 \approx h^2$ zu erreichen. Letzteres ist gerade die Größenordnung des lokalen Fehlers, vgl. (4.8), und wie wir im vergangenen Abschnitt in Feststellung 3.2 gesehen haben, ist eine höhere Genauigkeit nicht erforderlich. Zusammenfassend ergibt sich dann der folgende Algorithmus zur Lösung von (4.2):

Algorithmus 4.4

- Berechne $J = f_y(t_{i+1}, y_i)$
- Setze $y_{i+1}^{(0)} = y_i \qquad [\text{oder } y_{i+1}^{(0)} = y_i + (y_i - y_{i-1})\,]$
- `for` $n = 0, 1, \ldots$ `do`

  Löse $(I - hJ) z^{(n)} = y_i + hf(t_{i+1}, y_{i+1}^{(n)}) - y_{i+1}^{(n)}$

  Setze $y_{i+1}^{(n+1)} = y_{i+1}^{(n)} + z^{(n)}$

  `until` $\|y_{i+1} - y_{i+1}^{(n+1)}\|_2 \approx h^2$.

Bemerkung. Die Abbruchbedingung ist natürlich in dieser Form nicht verwendbar, da y_{i+1} ja gerade die gesuchte, unbekannte Größe ist. Es gilt also, diesen Fehler in der Praxis zu schätzen, und dafür kann man die aus dem Banachschen Fixpunktsatz bekannte a posteriori Abschätzung

$$\|y_{i+1} - y_{i+1}^{(n+1)}\|_2 \leq \frac{q}{1-q} \|z^{(n)}\|_2$$

vewenden. Sie setzt voraus, daß die Quasi-Newton-Iteration in dem relevanten lokalen Bereich kontrahierend ist mit Kontraktionskonstante $q \ll 1$. Das unbekannte q kann seinerseits durch den Konvergenzfaktor der Folge $\{z^{(n)}\}$ mit $z^{(n)} \to 0$ geschätzt werden, etwa $q \approx \|z^{(n)}\|_2 / \|z^{(n-1)}\|_2$. Damit ergibt sich der implementierbare Fehlertest

$$\frac{\|z^{(n)}\|_2^2}{\|z^{(n-1)}\|_2 - \|z^{(n)}\|_2} \lesssim h^2$$

für die Abbruchbedingung von Algorithmus 4.4.

Moderne Implementierungen verwenden ein festes J über mehrere Zeitschritte hinweg und entscheiden adaptiv, wann J neu berechnet wird.

5 Runge-Kutta Verfahren

Der entscheidende Nachteil der beiden Euler-Verfahren ist ihre langsame Konvergenz. Betrachtet man den Beweis, so stellt man fest, daß hierfür allein der lokale Fehler verantwortlich ist. Anhand der Abbildungen 3.1 und 4.1 mag es einleuchten, daß die schlechte Konvergenz daran liegt, daß die Tangentensteigung an den Randpunkten des Intervalls $[t_i, t_{i+1}]$ die *Sekante* durch die optimalen Punkten $(t_i, y(t_i))$ und $(t_{i+1}, y(t_{i+1}))$ zu schlecht approximiert. Es ist daher naheliegend, zur Konvergenzverbesserung einen Ansatz der Form

$$y_{i+1} = y_i + h \sum_{j=1}^{s} b_j f(t_i + c_j h, \eta_j)\,, \qquad \sum_{j=1}^{s} b_j = 1\,, \tag{5.1}$$

zu wählen, mit Näherungen η_j für y_i bzw. y_{i+1} ; s nennt man dabei die **Stufenzahl** des Verfahrens. Speziell bei den beiden Euler-Verfahren ist jeweils $s = 1$ und $c_1 = 0, \eta_1 = y_i$ (explizites Euler-Verfahren), bzw., $c_1 = 1, \eta_1 = y_{i+1}$ (implizites Euler-Verfahren).

Da bei (5.1) jeweils ausgehend von $y_i \approx y(t_i)$ die nächste Näherung $y_{i+1} \approx y(t_{i+1})$ berechnet wird, spricht man bei Verfahren dieser Art von **Einschrittverfahren**. Im Gegensatz dazu verwenden **Mehrschrittverfahren** auch ältere Näherungen $y_{i-1}, \ldots$, zur Berechnung von y_{i+1}.

Nehmen wir nun an, daß $y_i = y(t_i)$ auf der exakten Lösungskurve liegt und bestimmen davon ausgehend wie im ersten Schritt des Beweises von Satz 3.1 den lokalen Fehler des Verfahrens (5.1): Dann ergibt sich mit dem Hauptsatz der Differentialrechnung

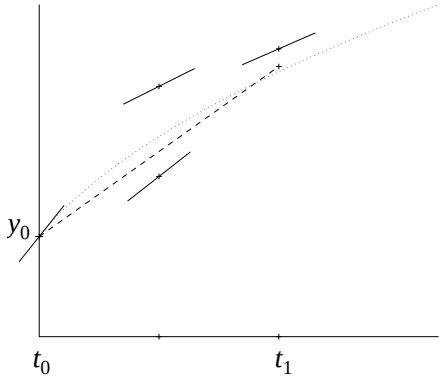


Fig. 5.1. Runge-Kutta Ansatz

$$\begin{aligned} y(t_{i+1}) - y_{i+1} &= y(t_{i+1}) - y(t_i) - h\sum_{j=1}^{s} b_j f(t_i + c_j h, \eta_j) \\ &= \int_{t_i}^{t_{i+1}} y'(t)\,dt \;-\; h\sum_{j=1}^{s} b_j f(t_i + c_j h, \eta_j) \\ &\underset{Dgl.}{=} \int_{t_i}^{t_{i+1}} f(t, y(t))\,dt \;-\; h\sum_{j=1}^{s} b_j f(t_i + c_j h, \eta_j)\,. \end{aligned}$$

Wir sehen daher, daß der lokale Fehler in dem Moment klein wird, in dem die Summe $h\sum_{j=1}^{s} b_j f(t_i + c_j h, \eta_j)$ eine gute Approximation des entsprechenden Integrals $\int_{t_i}^{t_{i+1}} f(t, y(t))\,dt$ ist.

Daher liegt es nahe, *Quadraturformeln* zur Wahl der Parameter $\{b_j\}$, $\{c_j\}$ und $\{\eta_j\}$ heranzuziehen.

Beispiel. Mit der Mittelpunktsformel (**??**) ergibt sich beispielsweise der Ansatz

$$y_{i+1} = y_i + h f(t_i + \tfrac{h}{2}, \eta_1)\,, \tag{5.2}$$

wobei idealerweise $\eta_1 = y(t_i + \frac{h}{2})$ sein sollte; allerdings ist dieser Wert nicht bekannt. Eine vernünftige Näherung ist jedoch

$$\eta_1 \;=\; y(t_i) + \frac{h}{2} y'(t_i) \underset{Dgl.}{=} y_i + \frac{h}{2} f(t_i, y_i)\,.$$

Dies ist gerade das **Verfahren von Runge** aus dem Jahr 1895. Durch Taylorentwicklung sieht man für hinreichend glattes f

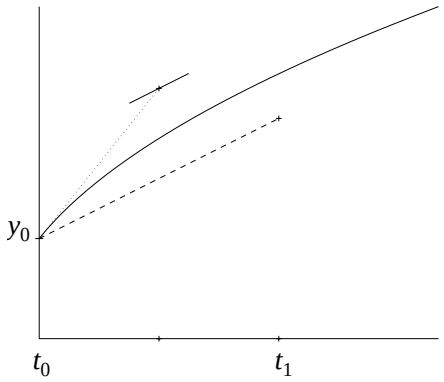


Fig. 5.2. Verfahren von Runge

$$
\begin{aligned}
y_{i+1} &= y_i + h\, f(t_i, y_i) + \frac{h^2}{2} f_t(t_i, y_i) + \frac{h^2}{2} f_y(t_i, y_i) f(t_i, y_i) + O(h^3)\,, \\
y(t_{i+1}) &= y(t_i) + h\, y'(t_i) + \frac{h^2}{2} y''(t_i) + O(h^3) \\
&= y_i + h\, f(t_i, y_i) + \frac{h^2}{2} \Big(f_t(t_i, y_i) + f_y(t_i, y_i) f(t_i, y_i) \Big) + O(h^3)\,,
\end{aligned}
$$

und daher gilt $\|y_{i+1} - y(t_{i+1})\|_2 = O(h^3)$. Das Verfahren von Runge hat also einen kleineren lokalen Fehler als die beiden Euler-Verfahren.

Definition 5.1 *Ein Einschrittverfahren hat die* ***(Konsistenz)-Ordnung*** q*, falls für jede Differentialgleichung* $y' = f(t, y)$ *mit* $f \in C^{q+1}(I \times J)$ *und für jedes* $t_i \in I$ *für den lokalen Fehler gilt:*

$$
y_i = y(t_i) \in J \quad \Longrightarrow \quad y_{i+1} - y(t_{i+1}) = O(h^{q+1})\,,\ h \to 0\,.
$$

Beachte: Die Ordnung ist q (und nicht $q+1$), obwohl die entsprechende h-Potenz $q+1$ ist! Wie wir in Satz 5.7 sehen werden, ist die Konvergenzordnung an einem festen Punkt $t_0 \in (0, T]$ bei einem Verfahren der Ordnung q nämlich lediglich $O(h^q)$.

Beispiel. Die beiden Euler-Verfahren haben die Ordnung $q = 1$ und das Verfahren von Runge hat die Ordnung $q = 2$.

Der Zusammenhang zwischen Quadraturverfahren und dem Ansatz (5.1) wird durch das folgende Resultat untermauert:

Satz 5.2 *Hat ein Einschrittverfahren der Form* (5.1) *die Ordnung q, dann hat die Quadraturformel $Q[g] = \sum_{j=1}^{s} b_j g(c_j) \approx \int_0^1 g(x)\,dx$ den Exaktheitsgrad $q-1$.*

Beweis. Für $0 \le n < q$ betrachten wir das spezielle "Anfangswertproblem"

$$y' = t^n\,, \qquad y(0) = 0\,.$$

Nach dem Satz von Picard-Lindelöf hat dieses Problem die eindeutige Lösung $y(t) = t^{n+1}/(n+1)$. Sei $y_0 = 0$: Dann gilt nach Definition 5.1 für ein Einschrittverfahren der Ordnung q die Abschätzung

$$|y(h) - y_1| \;=\; \Big|\, \frac{1}{n+1}\, h^{n+1} - h \sum_{j=1}^{s} b_j (c_j h)^n \,\Big| \;=\; O(h^{q+1})\,, \qquad h \to 0\,.$$

Dies ist offensichtlich *unabhängig* von der Wahl der η_j! Nach Division durch h^{n+1} erhält man

$$\Big|\, \frac{1}{n+1} - \sum_{j=1}^{s} b_j c_j^n \,\Big| \;=\; O(h^{q-n}) \;=\; o(1)\,, \qquad h \to 0\,,$$

und durch Grenzübergang $h \to 0$ ergibt sich zwangsläufig, daß

$$Q[t^n] = \sum_{j=1}^{s} b_j c_j^n \;=\; \frac{1}{n+1} \;=\; \int_0^1 t^n\,dt\,.$$

Also ist die Quadraturformel $Q[\,\cdot\,]$ für alle Monome t^n, $n = 0, \ldots, q-1$, und damit für den ganzen Unterraum Π_{q-1} exakt. □

Als unmittelbare Folgerung ergibt sich, daß ein s-stufiges Einschrittverfahren maximal die Ordnung $q = 2s$ haben kann, vgl. Proposition **??**.

Dieser Zusammenhang zwischen der Ordnung eines Einschrittverfahrens und dem Exaktheitsgrad einer Quadraturformel läßt sich gezielt weiterverfolgen, um Verfahren höherer Ordnung zu konstruieren. Dies ist die Idee der **Runge-Kutta Verfahren**: Dabei gilt es allerdings, noch eine Regel für die Wahl der $\{\eta_j\}$ anzugeben. Wegen

$$\eta_j \;\approx\; y(t_i + c_j h) \;=\; y(t_i) + \int_{t_i}^{t_i + c_j h} y'(t)\,dt \;=\; y(t_i) + \int_{t_i}^{t_i + c_j h} f(t, y(t))\,dt$$

bietet sich hier wieder eine Quadraturformel an. Um zusätzliche Funktionsauswertungen $f(t,y)$ zu vermeiden, beschränkt man sich dabei auf die *gleichen* Werte $f(t_i + c_j h, \eta_j)$, $j = 1, \ldots, s$, wie für die Berechnung von y_{i+1}. Das ergibt den folgenden Ansatz:

$$\eta_j = y_i + h \sum_{k=1}^{s} a_{jk} f(t_i + c_k h, \eta_k)\,, \qquad \sum_{k=1}^{s} a_{jk} = c_j\,. \tag{5.3}$$

Falls $a_{jk} = 0$ für $j \leq k$ ist diese Rechenvorschrift *explizit* und führt auf ein **explizites Runge-Kutta Verfahren**; ansonsten ergibt sich ein **implizites Runge-Kutta Verfahren**. Die Bedingung $\sum_{k=1}^{s} a_{jk} = c_j$ ist zwar natürlich und uns von Quadraturformeln geläufig, wird aber in der Literatur nicht einheitlich vorausgesetzt; es ist auf alle Fälle eine unwesentliche, aber sehr hilfreiche Einschränkung.

Üblicherweise werden die Koeffizienten $\{a_{jk}, b_j, c_j\}$ in einem quadratischen Tableau zusammengefaßt (das sogenannte **Runge-Kutta abc**),

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array} \quad = \quad \begin{array}{c|cccc} c_1 & a_{11} & & \cdots & a_{1s} \\ c_2 & a_{21} & & & \vdots \\ c_3 & a_{31} & a_{32} & & \\ \vdots & & & & \\ c_s & a_{s1} & \cdots & a_{s,s-1} & \\ \hline & b_1 & b_2 \quad \cdots & b_{s-1} & b_s \end{array}$$

wobei wir kurzerhand $A = [a_{jk}] \in \mathbb{R}^{s \times s}$, $b = [b_1, \ldots, b_s]^T \in \mathbb{R}^s$ und $c = [c_1, \ldots, c_s]^T \in \mathbb{R}^s$ gesetzt haben. Wir sprechen im weiteren kurz von dem Runge-Kutta Verfahren (A, b, c).

Beispiel 5.3 Für das explizite Euler-Verfahren und für das implizite Euler-Verfahren ergeben sich beispielsweise die folgenden Tableaus:

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

Das Verfahren von Runge scheint auf den ersten Blick nicht in das allgemeine Runge-Kutta Schema hineinzupassen, da zur Berechnung von η_1 auf den Funktionswert $f(t_i, y_i)$ zugegriffen wird, der nicht in der Rechenvorschrift (5.2) vorkommt. Daher behilft man sich mit einem Kunstgriff und führt künstlich $c_0 = 0$ mit $\eta_0 = y_i$ als weitere Stufe ein; damit wird das Verfahren von Runge zu einem zweistufigen Runge-Kutta Verfahren mit dem Tableau

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

Wir wollen nun versuchen, ein Verfahren dritter Ordnung zu konstruieren und leiten uns dafür zunächst Bedingungen an die Parameter her.

Satz 5.4 *Runge-Kutta Verfahren haben mindestens die Ordnung eins. Ein Runge-Kutta Verfahren* (5.1), (5.3) *ist von zweiter Ordnung, wenn*

$$\sum_{j=1}^{s} b_j c_j = \frac{1}{2} \,. \tag{5.4}$$

Es ist von dritter Ordnung, wenn darüberhinaus

$$\sum_{j=1}^{s} b_j c_j^2 \;=\; \frac{1}{3}\,, \qquad \textit{und} \qquad \sum_{j=1}^{s} b_j \sum_{k=1}^{s} a_{jk} c_k \;=\; \frac{1}{6}\;. \tag{5.5}$$

Beweis. Laut Definition können wir uns auf Differentialgleichungen mit hinreichend glatter rechter Seite f beschränken. Dann gilt aufgrund der Taylor-Entwicklung und der Gültigkeit der Differentialgleichung

$$\begin{aligned} y(t_i+h) \;&=\; y(t_i) \;+\; hy'(t_i) \;+\; \tfrac{1}{2}\,h^2 y''(t_i) \;+\; \tfrac{1}{6}\,h^3 y'''(t_i) \;+\; O(h^4) \\ &=\; y \;+\; hf \;+\; \tfrac{1}{2}\,h^2(f_t + f_y f) \;+\; \tfrac{1}{6}\,h^3(f_{tt} + 2f_{ty}f + f^* f_{yy} f + f_y f_t + f_y^2 f) \\ &\quad\; +\; O(h^4)\,, \end{aligned} \tag{5.6}$$

wobei bei der Funktion f und ihren partiellen Ableitungen immer das (konstante) Argument $(t_i, y(t_i))$ weggelassen wurde.

Zum Vergleich nun eine Entwicklung von y_{i+1} nach Potenzen von h. Dazu ist zunächst zu beachten, daß wegen (5.3)

$$\eta_j - y_i \;=\; h\sum_{k=1}^{s} a_{jk} f(t_i + c_k h, \eta_k) \;=\; h\sum_{k=1}^{s} a_{jk} f(t_i, y_i) + O(h^2) \;=\; hc_j f(t_i, y_i) + O(h^2)\,,$$

und daher ergibt eine Taylorentwicklung von (5.3) genauer (Argumente (t_i, y_i) bei f und dessen Ableitungen werden der Einfachheit halber wieder weggelassen)

$$\begin{aligned} \eta_j \;&=\; y_i + h\sum_{k=1}^{s} a_{jk} f(t_i + c_k h, \eta_k) \\ &=\; y_i + h\sum_{k=1}^{s} a_{jk}\,\big(\, f + f_t\, c_k h + f_y(\eta_k - y_i) + O(h^2)\,\big) \\ &=\; y_i \;+\; h\, f \sum_{k=1}^{s} a_{jk} \;+\; h^2\big(f_t \sum_{k=1}^{s} a_{jk} c_k \;+\; f_y \sum_{k=1}^{s} a_{jk} c_k f\big) \;+\; O(h^3)\,, \end{aligned}$$

also

$$\eta_j \;=\; y_i \;+\; h\, f c_j \;+\; h^2(f_t + f_y f)\sum_{k=1}^{s} a_{jk} c_k \;+\; O(h^3)\,.$$

Damit ergibt sich schließlich aus (5.1)

$$
\begin{aligned}
y_{i+1} &= y_i + h\sum_{j=1}^{s} b_j f(t_i + c_j h, \eta_j) \\
&= y_i + h\sum_{j=1}^{s} b_j \Big(f + f_t\, c_j h + f_y(\eta_j - y_i) + \tfrac{1}{2}\, f_{tt}\, c_j^2 h^2 + f_{ty} c_j h(\eta_j - y_i) \\
&\qquad + \tfrac{1}{2}\,(\eta_j - y_i)^* f_{yy}(\eta_j - y_i) + \ldots \Big) \\
&= y_i + h\, f \sum_{j=1}^{s} b_j + h^2 f_t \sum_{j=1}^{s} b_j c_j + h^2 f_y f \sum_{j=1}^{s} b_j c_j + h^3 f_y (f_t + f_y f) \sum_{j=1}^{s} b_j \sum_{k=1}^{s} a_{jk} c_k \\
&\quad + h^3 f_{tt} \frac{1}{2} \sum_{j=1}^{s} b_j c_j^2 + h^3 f_{ty} f \sum_{j=1}^{s} b_j c_j^2 + h^3 f^* f_{yy} f \frac{1}{2} \sum_{j=1}^{s} b_j c_j^2 + O(h^4) \\
&= y_i + h\, f + h^2 (f_t + f_y f) \sum_{j=1}^{s} b_j c_j + h^3 (f_{tt} + 2 f_{ty} f + f^* f_{yy} f) \frac{1}{2} \sum_{j=1}^{s} b_j c_j^2 \\
&\quad + h^3 f_y (f_t + f_y f) \sum_{j=1}^{s} b_j \sum_{k=1}^{s} a_{jk} c_k + O(h^4)\,,
\end{aligned}
$$

wobei wir zuletzt verwendet haben, daß $\sum_{j=1}^{s} b_j = 1$, vgl. (5.1).

Nach Definition 5.1 gilt es, dieses Ergebnis mit (5.6) zu vergleichen (unter der Voraussetzung $y_i = y(t_i)$): Demnach ist jedes Runge-Kutta Verfahren ein Verfahren erster Ordnung und hat die Ordnung 2, wenn (5.4) erfüllt ist; es hat die Ordnung 3, wenn darüberhinaus die beiden Bedingungen (5.5) erfüllt sind. □

Man überprüft sofort, daß beim Verfahren von Runge (5.4) erwartungsgemäß erfüllt ist:

$$
b_1 c_1 + b_2 c_2 = 0 \cdot 0 + \frac{1}{2} \cdot 1 = \frac{1}{2}\,.
$$

Die beiden Gleichungen (5.5) sind hingegen beide *nicht* erfüllt.

Beachte: Die ersten beiden Gleichungen von (5.4), (5.5) bedeuten gerade, daß die Quadraturformel

$$
Q[p] = \sum_{k=1}^{s} b_k p(c_k) \approx \int_0^1 p(x)\, dx
$$

für alle Polynome $p \in \Pi_2$ exakt ist (vgl. Satz 5.2).

Beispiel 5.5 Wann immer in der Literatur oder in den Anwendungen von *dem* Runge-Kutta Verfahren gesprochen wird, dann ist das folgende explizite Verfahren von **Kutta** (1901) auf der Basis der Simpson-Formel gemeint. Das besondere an diesem Verfahren ist die

Verdoppelung des mittleren Knotens $c = 1/2$ bei gleichzeitiger Halbierung des zugehörigen Gewichts $b = 2/3$. Dadurch ergeben sich die vier Stufen

$$c_1 = 0, \quad c_2 = 1/2, \quad c_3 = 1/2, \quad c_4 = 1\,.$$

mit den Gewichten

$$b_1 = 1/6, \quad b_2 = 1/3, \quad b_3 = 1/3, \quad b_4 = 1/6\,.$$

Wegen des Exaktheitsgrads $q = 3$ der Simpson-Formel sind die Bedingung (5.4) und die erste, von $\{a_{jk}\}$ unabhängige Bedingung in (5.5) zwangsläufig erfüllt. Berücksichtigt man, daß das Verfahren explizit sein soll, vereinfacht sich die verbleibende Ordnungsbedingung in (5.5) zu

$$\tfrac{1}{6}\,a_{32} + \tfrac{1}{12}\,a_{42} + \tfrac{1}{12}\,a_{43} = \tfrac{1}{6}\,,$$

so daß die Bestimmung der $\{a_{jk}\}$ in dieser Weise unterbestimmt ist. Erweitert man Satz 5.4 noch um die Bedingungen für ein Verfahren vierter Ordnung ($\longrightarrow$ Übungen), dann ergibt sich eine eindeutige Lösung aller Ordnungsbedingungen für ein explizites Verfahren vierter Ordnung, die im folgenden Tableau dargestellt ist:

0				
1/2	1/2			
1/2	0	1/2		
1	0	0	1	
	1/6	1/3	1/3	1/6

Die Abbildung 5.1 zeigt das Verfahren von Kutta für ein Testbeispiel.

Es stellt sich nun zwangsläufig die Frage, welche Ordnung mit einem s-stufigen Verfahren überhaupt erreichbar ist. Wie bereits oben bemerkt wurde, kann diese maximale Ordnung höchstens $2s$ sein. Das Verfahren von Kutta ist trotzdem in der Klasse der vierstufigen Verfahren optimal, wie das folgende Resultat zeigt, das wir allerdings erst im nächsten Abschnitt beweisen können.

Bemerkung 5.6 Ein s-stufiges *explizites* Runge-Kutta Verfahren hat höchstens die Ordnung $q = s$.

Eines der meist verwendeten Verfahren läuft unter dem Namen `dopri5`: Dieses explizite sechsstufige Verfahren fünfter Ordnung ist in fast allen einschlägigen Programmbibliotheken implementiert; das zugehörige Runge-Kutta Tableau findet man etwa in Deuflhard/Bornemann (Numerische Mathematik II).

Wir beweisen nun den bereits angekündigten Satz über die Konvergenz allgemeiner Runge-Kutta Verfahren.

Satz 5.7 *Sei $I = [0, T]$ und $f : I \times \mathbb{R}^d \to \mathbb{R}^d$ sei $q+1$ mal stetig differenzierbar in $I \times \mathbb{R}^d$ und das Runge-Kutta Verfahren (A, b, c) habe die Ordnung q. Dann existiert ein $h_0 > 0$, so daß*

bei Schrittweite $h \in (0, h_0)$ alle Näherungen y_i des Runge-Kutta Verfahrens für $t_i = ih \in I$ lokal eindeutig definiert sind. Ferner gilt

$$\|y(t_i) - y_i\|_2 \leq Ch^q\,, \qquad h < h_0\,,$$

wobei die Konstante C von i und h unabhängig ist solange t_i in $(0, T]$ liegt.

Beweis. Der Beweis geht analog zu den Beweisen für die beiden Euler-Verfahren. Dabei ist der erste Beweisschritt über den lokalen Fehler wegen der Ordnung des Runge-Kutta Verfahrens trivialerweise erledigt. Für den zweiten Beweisschritt betrachten wir zunächst den *expliziten* Fall. In diesem Fall hat die Näherung y_{i+1} des Runge-Kutta Verfahrens gemäß (5.1) die Form

$$y_{i+1} = y_i + h\Phi(t_i, y_i, h)\,, \tag{5.7}$$

wobei die Funktion Φ in elementarer Weise aus der Funktion f zusammengesetzt ist und daher Lipschitz-stetig von y_i abhängt. Ist ferner

$$z_{i+1} = z_i + h\Phi(t_i, z_i, h)\,,$$

so folgt unmittelbar

$$\|y_{i+1} - z_{i+1}\|_2 \leq (1 + hc_1)\ \|y_i - z_i\|_2 \tag{5.8}$$

für ein festes $c_1 > 0$. Dies ist die Kernaussage des zweiten Beweisschritts.

Wir betrachten nun einen Runge-Kutta Schritt eines *impliziten* Verfahrens zum (festen) Zeitpunkt $t = t_i$. Dazu schreiben wir die durch (5.3) definierten Stufen η_j untereinander in einen großen Vektor $\boldsymbol{\eta}$ und übertragen entsprechend s Kopien der Näherung y_i nach i Zeitschritten in einen großen Vektor $\boldsymbol{y}$. Da wir uns für die Abhängigkeit von y_i interessieren, definieren wir ferner einen großen Vektor $\boldsymbol{z}$, bestehend aus s Kopien einer Approximation z_i von y_i. Für dieses z_i ergibt sich nun $\boldsymbol{\eta}$ aus einer impliziten Gleichung der Form

$$F(\boldsymbol{\eta}, h, \boldsymbol{z}) = \boldsymbol{\eta} - h\Psi(\boldsymbol{\eta}, h) - \boldsymbol{z} = \boldsymbol{0}\,. \tag{5.9}$$

Dabei ist (trivialerweise) $F(\boldsymbol{y}, 0, \boldsymbol{y}) = \boldsymbol{0}$ und F an der Stelle $(\boldsymbol{y}, 0, \boldsymbol{y})$ nach der ersten Blockvariablen differenzierbar, wobei $F_{\boldsymbol{\eta}}(\boldsymbol{y}, 0, \boldsymbol{y}) = I$, also invertierbar ist. Nach dem Satz über implizite Funktionen existiert somit eine lokal eindeutig bestimmte Lösung $\boldsymbol{\eta} = \boldsymbol{\eta}(h, \boldsymbol{z})$ der Gleichung (5.9) für alle $h \in [0, h_0)$ und für alle $\boldsymbol{z}$ in der Umgebung von $\boldsymbol{y}$. Ferner ist die Funktion $\boldsymbol{\eta}$ in diesem Gebiet stetig nach h und z_i differenzierbar. Somit ergibt sich mit der Kettenregel, daß

$$z_{i+1} = z_i + h\sum_{j=1}^{s} b_j f\left(t_i + c_j h, \eta_j(h, z_i)\right)$$

Lipschitz-stetig von z_i abhängt und man erhält wie zuvor die gewünschte Aussage (5.8) für jedes $h \in (0, h_0)$.

Der Beweis ist an dieser Stelle nicht ganz lückenlos, da eigentlich noch gezeigt werden muß, daß h_0 und c_1 in (5.8) unabhängig von t_i gewählt werden kann. Das läßt sich jedoch mit einem üblichen Kompaktheitsargument nachweisen.

Mit Hilfe von (5.8) ergibt sich nun wie in den früheren Beweisen im dritten Beweisschritt für den Gesamtfehler $\varepsilon_i = \|y(t_i) - y_i\|_2$ eine Rekursion der Form

$$\varepsilon_{i+1} \leq (1 + c_1 h)\, \varepsilon_i \, + \, c_2 h^{q+1} \,, \qquad c_1, c_2 > 0 \,, \tag{5.10}$$

und durch Induktion erhält man schließlich für alle $i = 0, \ldots, n$ mit $h = T/n$ die Ungleichung

$$\varepsilon_i \, \leq \, \frac{c_2}{c_1} \, (1 + 2c_1 h)^i \, h^q \, \leq \, \frac{c_2}{c_1} \, \big(1 + \tfrac{2c_1 T}{n}\big)^n \, h^q \, \leq \, \frac{c_2}{c_1} \, e^{2c_1 T} \, h^q \,.$$

□

6 Stabilitätstheorie

Zentraler Punkt des vergangenen Abschnitts war die Erhöhung der Konvergenzgeschwindigkeit für $h \to 0$. Nicht berücksichtigt wurde dabei jedoch die Stabilität des Verfahrens, d.h., die Anfälligkeit der Rekursion (5.1) gegenüber Fehlern und Fehlerfortpflanzung. Dies entspricht genau dem zweiten Teilschritt des allgemeinen Beweisprinzips, das in den Sätzen 3.1, 4.2 und 5.7 zum tragen kam. Die Frage ist also, in welcher Weise Approximationsfehler in y_i im $(i+1)$-ten Schritt propagiert werden:

$$y(t_i) - y_i \qquad \overset{?}{\rightsquigarrow} \qquad y(t_{i+1}) - y_{i+1} \,.$$

Wie wir im Zusammenhang mit dem expliziten und impliziten Euler-Verfahren gesehen haben, können Runge-Kutta Verfahren ein sehr unterschiedlich stabiles Verhalten aufweisen. Im folgenden soll das Stabilitätsverhalten eines allgemeinen Einschrittverfahrens untersucht werden. Dazu ist es jedoch zunächst erforderlich, das allgemeine Anfangswertproblem (1.1) auf ein einfaches Modellproblem zu reduzieren. Diese Reduktion kann für Differentialgleichung und Runge-Kutta Verfahren parallel erfolgen:

1. **Linearisierung**
   Wir interessieren uns für den Einfluß einer (kleinen) Störung u_i des exakten Werts $y(t_i)$ unserer Differentialgleichung: Bezeichnen wir mit $y + u$ die zugehörige Lösung, dann gilt – zumindest in einem Zeitintervall, in dem die resultierende Störung u "klein" bleibt –

   $$(y+u)' = f(t, y+u) \, \approx \, f(t,y) + \underbrace{f_y(t,y)}_{\in \, \mathbb{R}^{d \times d}} u \, = \, y' + A(t,y)u \,, \qquad A(t) \in \mathbb{R}^{d \times d} \,.$$

2. **Einfrieren der Zeit**
   Im zweiten Schritt betrachten wir einen kurzen Zeitschritt $\Delta t \equiv h$ und vernachlässigen

dabei den Einfluß der Zeit in der Jacobi-Matrix $A(t,y)$, gehen also davon aus, daß sich lokal die Lösung der linearisierten Differentialgleichung wie die Lösung der stationären Differentialgleichung

$$u' = Au\,, \qquad A = f_y(t_i, y_i) \in \mathbb{R}^{d\times d}\,,$$

mit gleichem Anfangswert $u(t_i) = u_i$ verhält.

3. **Diagonalisierung**
   Dieser letzte Schritt beruht auf der Annahme, daß die Matrix A diagonalisierbar ist: Es existiere also eine Basis $\{x_1, \ldots, x_d\} \subset \mathbb{R}^d$ mit $Ax_n = \lambda_n x_n$, $n = 1, \ldots, d$. Entwickelt man $u(t) = \sum_{n=1}^{d} \eta_n(t)\, x_n$, dann ergibt sich

$$\eta_n' = \lambda_n \eta_n\,, \qquad n = 1, \ldots, d\,.$$

   Die η_n sind nun skalare Funktionen der Zeit; die zugehörigen Differentialgleichungen entsprechen gerade der Modellgleichung aus den vorigen Abschnitten.

Es ist also unter den getroffenen Annahmen ausreichend, die Fortpflanzung kleiner Ausgangsstörungen in diesen entkoppelten Differentialgleichungen zu betrachten. Die Hoffnung ist dann, daß das Resultat

$$\sum_{n=1}^{d} \eta_n(t)\, x_n$$

eine gute Approximation an den fortgepflanzten Fehler $u(t)$ des nichtlinearen zeitabhängigen Problems darstellt.

Im Rest dieses Abschnitts betrachten wir daher nur noch die eindimensionale **Testgleichung**

$$y' = \lambda y, \quad y(0) = 1\,, \qquad \lambda \in \mathbb{C}\,. \tag{6.1}$$

Die Lösung $y(t) = e^{\lambda t}$ verhält sich dabei in Abhängigkeit von λ wie folgt:

$$\begin{aligned}
&\text{Re } \lambda > 0: && |y(t)| \to \infty \quad \text{für } t \to \infty\,;\\
&\text{Re } \lambda < 0: && |y(t)| \to 0 \quad \text{für } t \to \infty\,;\\
&\text{Re } \lambda = 0: && |y(t)| = |y_0| \quad \text{für alle } t \in \mathbb{R}_0^+\,.
\end{aligned}$$

Eine Grundregel der Numerik lautet, daß eine numerische Lösung möglichst viele Eigenschaften der kontinuierlichen Lösung besitzen sollte; demnach sind wir vor allem an solchen numerischen Algorithmen für Anfangswertaufgaben interessiert, die die obigen drei Eigenschaften der einfachsten Modellgleichung realisieren. Dennoch dürften einige der folgenden Stabilitätsbegriffe erst im folgenden Abschnitt klarer werden.

Definition 6.1 *Seien $\{y_n\}$ die Näherungen eines numerischen Verfahrens zur Lösung der Testgleichung* (6.1)*. Dann bezeichnet man das Verfahren als* ***A-stabil****, falls bei beliebigem $\lambda \in \mathbb{C}$ mit Re $\lambda \le 0$ die Näherungen bei jeder Schrittweite h kontraktiv sind, also wenn*

$$|y_{n+1}| \le |y_n| \qquad \textit{für alle } n \textit{ und beliebiges } h\,.$$

Das Verfahren heißt ***Isometrie erhaltend****, wenn für beliebiges* λ *mit* $Re\ \lambda = 0$ *gilt, daß*

$$|y_{n+1}| = |y_n| = |y_0| \qquad \textit{für alle } n \textit{ und beliebiges } h\,.$$

In der Literatur wird der Begriff der A-Stabilität nicht ganz einheitlich verwendet: Manche Autoren fordern anstelle der obigen Definition, daß $|y_{n+1}| < |y_n|$ falls Re $\lambda < 0$ ist. Für Runge-Kutta Verfahren sind die beiden Definitionen allerdings äquivalent (dies folgt aus dem folgenden Resultat, da die dort definierte Stabilitätsfunktion bis auf höchstens endlich viele Punkte stetig und sogar holomorph ist).

Definition und Satz 6.2 *Es bezeichne* $\mathbf{1} = [1, \ldots, 1]^T \in \mathbb{R}^s$*. Zu jedem Runge-Kutta Verfahren* (A, b, c) *existiert eine rationale Funktion*

$$R(\zeta) = 1 + \zeta b^T (I - \zeta A)^{-1} \mathbf{1}\,,$$

so daß sich bei Anwendung auf die Testgleichung (6.1) *mit Schrittweite* $h > 0$ *die Näherungen*

$$y_n = \big(R(h\lambda)\big)^n\,, \qquad n = 0, 1, \ldots\,, \tag{6.2}$$

ergeben. Dabei ist R *eine rationale Funktion mit Zähler- und Nennergrad höchstens* s*; ist das Runge-Kutta Verfahren explizit, dann ist* R *ein Polynom. Die Funktion* R *heißt* ***Stabilitätsfunktion****.*

Beweis. Wir führen die folgenden s-dimensionalen Vektoren ein:

$$u = \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_s \end{bmatrix}, \quad \tau = \begin{bmatrix} t_i + c_1 h \\ \vdots \\ t_i + c_s h \end{bmatrix}, \quad f(\tau, u) = \begin{bmatrix} f(\tau_1, \eta_1) \\ \vdots \\ f(\tau_s, \eta_s) \end{bmatrix} = \lambda u\,.$$

Gemäß (5.1), (5.3), ergibt sich dann die Näherung y_{n+1} aus y_n durch Lösen des Gleichungssystems

$$\begin{aligned} u &= y_n \mathbf{1} + hAf(\tau, u) = y_n \mathbf{1} + h\lambda A u\,, \\ y_{n+1} &= y_n + hb^T f(\tau, u) = y_n + h\lambda b^T u\,. \end{aligned}$$

Demnach ergibt sich $(I - h\lambda A)u = y_n \mathbf{1}$ und

$$y_{n+1} = y_n + h\lambda b^T (I - h\lambda A)^{-1} (y_n \mathbf{1}) = R(h\lambda)\, y_n\,, \tag{6.3}$$

sowie induktiv die gewünschte Darstellung (6.2) von y_n.

Wenn das Runge-Kutta Verfahren explizit ist, dann ist $A \in \mathbb{R}^{s \times s}$ eine strikte untere Dreiecksmatrix und folglich $A^s = 0$; demnach ergibt sich (nachrechnen !)

$$(I - h\lambda A)^{-1} = I + h\lambda A + \ldots + (h\lambda)^{s-1} A^{s-1}\,, \tag{6.4}$$

so daß R ein Polynom vom Grad s ist. Um zu sehen, daß R im allgemeinen eine rationale Funktion ist, greifen wir auf die Cramersche Regel zurück, wonach sich die einzelnen Komponenten η_k von $u = (I - h\lambda A)^{-1}(y_n \mathbf{1})$ in der Form

$$\eta_k = p_k(h\lambda)/\det(I - h\lambda A) \quad \text{mit} \quad p_k \in \Pi_{s-1}\,, \qquad k = 1, \ldots, s\,,$$

schreiben lassen. Die Behauptung folgt dann unmittelbar aus (6.3). □

Man beachte, daß die Stabilitätsfunktion *nicht* von der Wahl der Knoten $\{c_j\}$ abhängt. Dies ist auch durchaus plausibel, da die verwendete Testgleichung autonom ist, also nicht von der Zeit abhängt.

Beispiel 6.3 In (4.3) haben wir bereits gesehen, daß das implizite Euler-Verfahren die Stabilitätsfunktion $R(\zeta) = \frac{1}{1-\zeta}$ besitzt. Zum expliziten Euler-Verfahren gehört hingegen die Stabilitätsfunktion

$$R(\zeta) = 1 + \zeta(1 - 0)^{-1} 1 = 1 + \zeta\,.$$

Für das Runge-Kutta Verfahren aus Beispiel 5.5 ergibt sich mit Hilfe von (6.4) die Stabilitätsfunktion

$$\begin{aligned} R(\zeta) &= 1 + \zeta\, [1/6, 1/3, 1/3, 1/6] \begin{bmatrix} 1 & 0 & 0 & 0 \\ \zeta/2 & 1 & 0 & 0 \\ \zeta^2/4 & \zeta/2 & 1 & 0 \\ \zeta^3/4 & \zeta^2/2 & \zeta & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \\ &= 1 + \zeta\, [1/6, 1/3, 1/3, 1/6] \begin{bmatrix} 1 \\ 1 + \zeta/2 \\ 1 + \zeta/2 + \zeta^2/4 \\ 1 + \zeta + \zeta^2/2 + \zeta^3/4 \end{bmatrix} \\ &= 1 + \zeta\,(1 + \zeta/2 + \zeta^2/6 + \zeta^3/24) \;=\; 1 + \zeta + \tfrac{1}{2}\zeta^2 + \tfrac{1}{6}\zeta^3 + \tfrac{1}{24}\zeta^4\,. \end{aligned}$$

Man erkennt, daß R_4 gerade aus den ersten fünf Summanden der Taylorreihe von e^ζ besteht. Das hat einen tieferen Grund, wie das folgende Resultat zeigt.

Satz 6.4 *Ist R die Stabilitätsfunktion eines Einschrittverfahrens der Ordnung q, dann gilt*

$$R(\zeta) = e^\zeta + O(\zeta^{q+1})\,, \qquad \zeta \to 0\,.$$

Beweis. $R(\zeta)$ und e^ζ können beide in eine lokal konvergente Taylorreihe um $\zeta = 0$ entwickelt werden. Ferner gilt nach Definition 5.1 bei Anwendung des Runge-Kutta Verfahrens auf die Testgleichung (6.1) mit $\lambda = 1$ und Schrittweite $h > 0$:

$$y_1 - e^h = R(h) - e^h = O(h^{q+1}), \qquad h \to 0 .$$

Daher müssen die ersten q Terme der Taylorreihen übereinstimmen und daraus folgt unmittelbar die Behauptung. □

Die Umkehrung dieses Satzes ist offensichtlich falsch, da – wie bereits oben erwähnt – die Stabilitätsfunktion nicht von dem Vektor c abhängt, während c bereits für ein Verfahren zweiter Ordnung speziellen Bedingungen genügen muß, vgl. Satz 5.4.

Als einfache Anwendung von Satz 6.4 beweisen wir nun die im vorigen Abschnitt formulierte Bemerkung 5.6.

Beweis von Bemerkung 5.6. Für ein s-stufiges explizites Runge-Kutta Verfahren ist R ein Polynom vom Grad s, und daher kann in Satz 6.4 q maximal s werden, nämlich dann, wenn R das s-te Taylorpolynom der Exponentialfunktion ist. □

Mit dem folgenden Satz kommen wir auf die in Definition 6.1 eingeführten Stabilitätseigenschaften eines Runge-Kutta Verfahrens zurück. Diese Eigenschaften können nämlich unmittelbar an der Stabilitätsfunktion abgelesen werden. Der Beweis ist so offensichtlich, daß wir ihn weglassen können.

Satz 6.5 *Gegeben sei ein Runge-Kutta Verfahren mit Stabilitätsfunktion R. Dann gilt:*

(a) Das Verfahren ist genau dann A-stabil, wenn $|R(\zeta)| \le 1$ für alle ζ mit $Re\ \zeta < 0$;

(b) Das Verfahren ist genau dann Isometrie erhaltend, wenn $|R(\zeta)| = 1$ für alle ζ mit $Re\ \zeta = 0$.

Bemerkungen. Wir schließen unmittelbar aus diesem Resultat, daß alle expliziten Runge-Kutta Verfahren *keine* der beiden Eigenschaften haben können, da für Polynome grundsätzlich $R(\infty) = \infty$ gilt.

Das implizite Euler-Verfahren mit Stabilitätsfunktion $R(\zeta) = (1-\zeta)^{-1}$ hingegen ist A-stabil, denn

$$|1 - \zeta|^2 \;=\; (1 - \mathrm{Re}\ \zeta)^2 + (\mathrm{Im}\ \zeta)^2 \;=\; 1 - 2\,\mathrm{Re}\ \zeta + |\zeta|^2 \;\ge\; 1 \qquad \text{für Re } \zeta \le 0 .$$

Leider ist die Ordnung des impliziten Euler-Verfahrens zu schlecht für ein praktikables Verfahren. Wir werden uns daher in Abschnitt 8 verstärkt impliziten Runge-Kutta Verfahren zuwenden.

Zuvor aber noch einige Bemerkungen zu expliziten Verfahren:

Definition 6.6 *Mit* $\mathcal{S} := \{\zeta \in \mathbb{C} : |R(\zeta)| \leq 1\}$ *wird das* ***Stabilitätsgebiet*** *eines Runge-Kutta Verfahrens bezeichnet.*

Für A-stabile Runge-Kutta Verfahren ist die abgeschlossene linke Halbebene $\mathbb{C}^-$ von $\mathbb{C}$ in $\mathcal{S}$ enthalten. Im folgenden sammeln wir einige allgemeine Aussagen zum Stabilitätsgebiet eines beliebigen Runge-Kutta Verfahrens.

Lemma 6.7 *Für jedes Runge-Kutta Verfahren ist* $0 \in \partial\mathcal{S}$.

Beweis. Da jedes Runge-Kutta Verfahren (mindestens) die Ordnung 1 besitzt, gilt nach Satz 6.4, daß

$$R(\zeta) = 1 + \zeta + O(\zeta^2)\,, \qquad \zeta \to 0\,.$$

Demnach ist $R(0) = 1$, also $0 \in \mathcal{S}$, aber es gibt ein ganzes Teilintervall $(0, \varepsilon) \notin \mathcal{S}$, denn

$$|R(\zeta)| > 1 + \zeta/2 > 1 \qquad \text{für } \zeta \in (0, \varepsilon)$$

mit $\varepsilon > 0$ hinreichend klein. □

Satz 6.8 *Das Stabilitätsgebiet eines expliziten Runge-Kutta Verfahrens ist immer beschränkt.*

Beweis. Dies folgt sofort daraus, daß die Stabilitätsfunktion eines expliziten Verfahrens ein Polynom ist, welches notwendigerweise für $|\zeta| \to \infty$ gegen unendlich strebt. □

Satz 6.9 *Das Stabilitätsgebiet* $\mathcal{S}$ *eines Runge-Kutta Verfahrens enthalte den Halbkreis*

$$\mathcal{B}_\tau^- := \{\zeta \in \mathbb{C}^- : |\zeta| \leq \tau\} \subset \mathcal{S}\,.$$

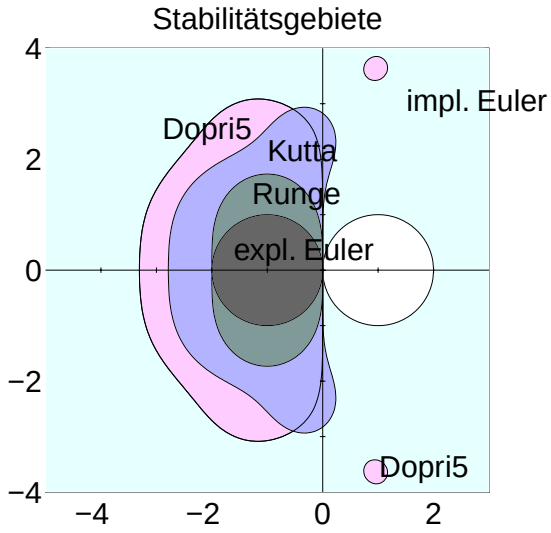
Dann gilt für die Näherungen y_n *des Runge-Kutta Verfahrens, angewendet auf die Testgleichung* (6.1) *mit* $\lambda \in \mathbb{C}^-$*:*

$$|y_{n+1}| \leq |y_n|\,, \qquad \textit{sobald } h < \tau/|\lambda|\,.$$

Beweis. Für Re $\lambda \leq 0$ und $h < \tau/|\lambda|$ liegt $\zeta := h\lambda$ in $\mathcal{B}_\tau^-$, und daher folgt die Aussage aus der Rekursion $y_{n+1} = R(h\lambda)y_n$. □

Satz 6.9 rettet also die Stabilität eines (evtl. expliziten) Runge-Kutta Verfahrens auf Kosten einer gegebenenfalls sehr kleinen Schrittweite h, abhängig von der Größe von $|\lambda|$ aber bemerkenswerterweise *unabhängig* von der Ordnung des Verfahrens.

Die folgende Abbildung zeigt die Stabilitätsgebiete der uns bislang bekannten Runge-Kutta Verfahren. Man kann erkennen, daß unter den expliziten Verfahren lediglich das Verfahren von Kutta (Beispiel 5.5) die Voraussetzung des Satzes 6.9 erfüllt. Der entsprechende Wert von τ ist $\tau = 2.615\ldots$.



7 Steife Differentialgleichungen

Beispiel 7.1 (Wärmeleitungsgleichung)
Wir greifen das Beispiel

$$u_t(t,x) = f(t,u) = u_{xx}(t,x)\,, \qquad u(0,x) = g(x)\,, \tag{7.1}$$

der Wärmeleitungsgleichung aus Abschnitt 1.3 mit Randbedingungen $u(t,-1) = u(t,1) = 0$ auf.

Da die rechte Seite $f(t,u) = u_{xx}$ ohnehin linear in u ist, entfällt der erste Reduktionsschritt des vorigen Abschnitts. Zudem ist die Funktion f auch noch unabhängig von der Zeit, so daß auch der zweite Reduktionsschritt entfällt.

Bleibt der dritte Reduktionsschritt: die Diagonalisierung des linearen Operators $Av = v_{xx}$, wobei $v \in C^2[-1,1]$ die Randbedingungen $v(-1) = v(1) = 0$ erfüllen muß. Die Eigenfunktionen dieses Operators können explizit angegeben werden, denn die Differentialgleichung

$$v_{xx} = \lambda v\,, \qquad v(-1) = v(1) = 0\,,$$

hat lediglich für $\lambda_n = -(\frac{\pi}{2}\,n)^2$, $n \in \mathbb{N}$, von Null verschiedene Lösungen, nämlich

$$v_n(x) = \begin{cases} \cos(\frac{\pi}{2}\,nx)\,, & n \text{ ungerade}\,, \\ \sin(\frac{\pi}{2}\,nx)\,, & n \text{ gerade}\,. \end{cases}$$

Damit sind die Eigenfunktionen und Eigenwerte von A bekannt, und wie im vorigen Abschnitt gesehen, kann die Lösung der Ausgangsgleichung (7.1) in diese Eigenfunktionen entwickelt werden,

$$u(t,x) = \sum_{n=1}^{\infty} \eta_{2n-1}(t)\,\cos\big(\frac{\pi}{2}\,(2n-1)x\big) \;+\; \sum_{n=1}^{\infty} \eta_{2n}(t)\,\sin\big(\pi nx\big)\,,$$

wobei die Koeffizientenfunktionen η_n Lösungen der gewöhnlichen Differentialgleichungen

$$\eta_n' = -\big(\frac{\pi}{2}\,n\big)^2\,\eta_n\,, \qquad n \in \mathbb{N}\,, \tag{7.2}$$

sind.

Zur Lösung dieser Differentialgleichungen benötigen wir Anfangswerte. Dazu betrachten wir die spezielle Situation, daß die Anfangstemperatur durch

$$u(0,x) = g(x) = 1 - x^2\,, \qquad -1 \le x \le 1\,,$$

gegeben sei. Aus der Theorie der Fourier-Reihen ist bekannt, daß g in eine gleichmäßig konvergente Fourier-Reihe entwickelt werden kann, nämlich

$$g(x) = \sum_{n=1}^{\infty} (-1)^{n+1}\,\frac{32}{\pi^3(2n-1)^3}\,\cos\big(\frac{\pi}{2}\,(2n-1)x\big)\,.$$

Demnach ergeben sich für die Koeffizientenfunktionen η_n die Anfangswerte

$$\eta_{2n}(0) = 0\,, \quad \eta_{2n-1}(0) = (-1)^{n+1}\,\frac{32}{\pi^3(2n-1)^3}\,, \qquad n > 0\,,$$

und durch Lösen der skalaren Differentialgleichungen (7.2) erhält man die analytische Darstellung

$$u(t,x) = \sum_{n=1}^{\infty} (-1)^{n+1} e^{-\pi^2(n-1/2)^2 t}\,\frac{32}{\pi^3(2n-1)^3}\,\cos\big(\frac{\pi}{2}\,(2n-1)x\big) \tag{7.3}$$

der Lösung von (7.1).

Stellen wir uns nun vor, wir wenden ein Runge-Kutta Verfahren zur Lösung der Differentialgleichung (7.1) an und ignorieren dabei den zusätzlichen Fehler, der bei der Diskretisierung der Ortsvariablen entsteht. Da das Runge-Kutta Verfahren nicht die exakte Lösung, sondern nur eine Approximation davon berechnen kann, haben wir bereits nach einem einzigen Zeitschritt in jeden Term der Reihe (7.3) einen Fehler der Größe ε_n eingeschleppt, und nach i weiteren Zeitschritten hat sich dieser Fehler zu

$$R(h\lambda_{2n-1})^i \varepsilon_n \;=\; R(-\pi^2 h(n-1/2)^2)^i \varepsilon_n$$

fortgepflanzt. Falls nun $-\pi^2 h(n-1/2)^2$ nicht im Stabilitätsbereich $\mathcal{S}$ des Verfahrens liegt, dann ist $|R(h\lambda_{2n-1})| > 1$ und der fortgepflanzte Fehler explodiert; wegen des entsprechenden Kosinusterms in (7.3) führt dies zu starken hochoszillierenden Artefakten in der Näherungslösung.

Diese Situation ist besonders unangenehm, da in der exakten Lösung (7.3) die hochfrequenten Kosinusanteile bereits nach kurzer Zeit $t > 0$ überhaupt keine Rolle mehr spielen, da die Vorfaktoren $e^{-\pi^2(n-1/2)^2 t}$ sehr schnell sehr klein werden. Trotzdem steuern gerade diese Anteile das Stabilitätsverhalten des Runge-Kutta Verfahrens. Da die zugehörigen Eigenwerte λ_{2n-1} für $n \to \infty$ gegen $-\infty$ streben, kann nur ein A-stabiles Runge-Kutta Verfahren vernünftige Näherungslösungen berechnen.

Beispiel 7.2 (Chemische Reaktionskinetik)
Für ein zweites, diesmal numerisches Beispiel, betrachten wir das folgende chemische Reaktionsmodell dreier Gase A, B und C:

$$\begin{aligned} A &\xrightarrow{0.04} B \\ B+B &\xrightarrow{3\cdot 10^7} C+B \\ B+C &\xrightarrow{10^4} A+C\,. \end{aligned}$$

Die Zahlen geben die entsprechenden Reaktionskoeffizienten k_i an. Demnach ist die erste Reaktion sehr langsam und die zweite extrem schnell. Das zugehörige Differentialgleichungssystem lautet

$$\begin{aligned} y_1' &= -0.04\, y_1 + 10^4\, y_2 y_3\,, \\ y_2' &= 0.04\, y_1 - 10^4 y_2 y_3 - 3\cdot 10^7 y_2^2\,, \\ y_3' &= 3\cdot 10^7 y_2^2\,. \end{aligned}$$

Zu Beginn der Reaktion sei lediglich Substanz A vorhanden, d.h., die Startwerte sind

$$y_1(0) = 1\,, \qquad y_2(0) = y_3(0) = 0\,.$$

Wir untersuchen die Lösung dieses Differentialgleichungssystems im Zeitintervall $[0, 0.3]$. Aufgrund der Reaktionsformeln erwarten wir einen langsamen Abbau der Substanz A und eine entsprechende Zunahme von Substanz C. Tatsächlich beobachtet man im wesentlichen lineare Funktionen y_1 und y_3, wobei zum Zeitpunkt $T = 0.3$ schließlich $y_1(T) = 0.9887$ und $y_3(T) = 0.0113$ ist. Das entscheidende "Zünglein an der Waage" ist die Substanz B, gleichwohl ihr Anteil im Gasgemisch verschwindend gering ist. Tatsächlich sieht man anhand der Differentialgleichung, daß Substanz B zunächst zunimmt bis die negativen Terme in y_2' überwiegen und dann nur noch abnimmt. Der Umschlagpunkt wird etwa dann erreicht, wenn

$$\max\{10^4 y_3, 3\cdot 10^7 y_2\}\, y_2 \approx 0.04\, y_1 \approx 0.04\,;$$

für den angegebenen Wert von y_3 ergibt dies $y_2^2 \approx 1.3 \cdot 10^{-9}$, d.h. y_2 ist an dieser Maximalstelle ungefähr von der Größenordnung $3.6 \cdot 10^{-5}$.

Die folgende Abbildung zeigt Näherungen für y_2, berechnet mit zwei verschiedenen Differentialgleichungslösern: `dopri5` und `ode23s`. Beide Verfahren steuern ihre Schrittweite selber, ein Thema, auf daß wir am Schluß der Vorlesung in Abschnitt 10 noch einmal zurückkommen werden.

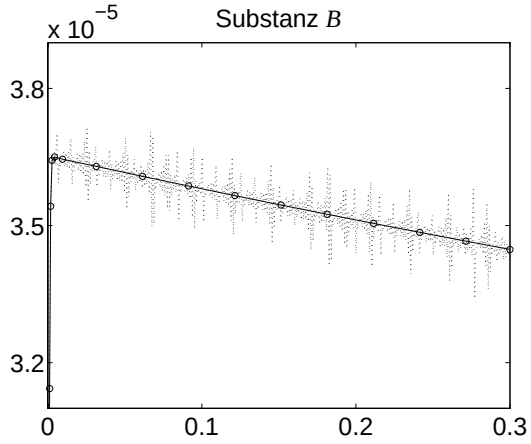


Fig. 7.1. Ergebnisse der beiden Verfahren

Das Verfahren `dopri5` haben wir bereits am Ende von Abschnitt 5 kennengelernt; es ist ein explizites Verfahren und damit wenig geeignet für steife Probleme. Dies ist sehr gut an der gepunktelten Lösungskurve in der Abbildung zu erkennen. Die Approximation oszilliert hin und her; für das Durchlaufen des gesamten Zeitintervalls werden 825 Zeitschritte benötigt.

Das Verfahren `ode23s`, auf der anderen Seite, ist hingegen ein A-stabiles Rosenbrock-Verfahren zweiter Ordnung, das wir in Abschnitt 9 noch genauer untersuchen werden. Wie man sieht, ergibt sich hier eine glatte Lösungskurve, wobei bis zum Zeitpunkt $T = 0.3$ lediglich 18 (!) Zeitschritte nötig sind – die Zeitschritte sind durch kleine Kreise auf der Kurve markiert. Wie man erwartet, werden kurze Zeitschritte lediglich benötigt, bis das Maximum von y_2 durchlaufen wird; danach sind relativ große Zeitschritte möglich – und dies, obwohl das Verfahren eine viel kleinere Ordnung hat als `dopri5`.

Das unterschiedliche Verhalten der beiden Lösungsverfahren nach Durchlaufen des Maximums von y_2 läßt sich wie folgt erklären. Die Jacobi-Matrix

$$f_y = \begin{bmatrix} -0.04 & 10^4 y_3 & 10^4 y_2 \\ 0.04 & -6 \cdot 10^7 y_2 - 10^4 y_3 & -10^4 y_2 \\ 0 & 6 \cdot 10^7 y_2 & 0 \end{bmatrix}$$

hat den Eigenwert $\lambda_1 = 0$ mit linkem Eigenvektor $[1, 1, 1]$. Die (rechten) Eigenvektoren zu den verbleibenden beiden Eigenwerten sind also senkrecht zu $[1, 1, 1]^T$, d.h., sie sind von der Form $[-\alpha - 1, \alpha, 1]^T$. Eine recht einfache Rechnung ergibt dann die Eigenwerte $\lambda_2 \approx -0.3$

und $\lambda_3 \approx -1900$, solange $y_1 \approx 1$, $y_2 \approx 3 \cdot 10^{-5}$ und $y_3 \approx 10^{-2}$ ist. Aufgrund des einen stark negativen Eigenwerts ist auch diese Differentialgleichung sehr steif.

8 Implizite Runge-Kutta Verfahren

Die vorigen Abschnitte haben den Bedarf an guten impliziten Runge-Kutta Verfahren offensichtlich gemacht. Das implizite Euler-Verfahren kommt dabei wegen seiner geringen Konvergenzordnung nicht ernsthaft in Betracht. Im folgenden sollen daher implizite Verfahren mit möglichst hoher Konvergenzordnung hergeleitet werden.

War nach Satz 5.6 für explizite Runge-Kutta Verfahren noch $q = s$ die maximal mögliche Ordnung, so können wir für implizite Verfahren lediglich mit Hilfe von Satz 5.2 auf die Oberschranke $q = 2s$ schließen. Für diese Ordnung muß aufgrund der Abschnitte **??** und **??** die Quadraturformel

$$Q[g] = \sum_{j=1}^{s} b_j g(c_j) \tag{8.1}$$

für das Integral $\int_0^1 g(t)\,dt$ den maximal möglichen Exaktheitsgrad $2s-1$ haben, also die s-stufige Gauß-(Legendre) Quadraturformel sein.

Demnach wählen wir im weiteren für $\{c_j\}$ die Nullstellen des s-ten Legendre-Polynoms (umskaliert auf das Intervall $[0,1]$) und für $\{b_j\}$ die zugehörigen Gewichte der Gauß-Quadraturformel. Die verbleibenden Koeffizienten $\{a_{jk}\}$ werden wie in (5.3) so gewählt, daß

$$h \sum_{k=1}^{s} a_{jk} f(t_i + c_k h, \eta_k) \;\approx\; \int_{t_i}^{t_i + c_j h} f(t, y(t))\,dt \;=\; h \int_0^{c_j} f\big(t_i + \tau h, y(t_i + \tau h)\big)\,d\tau\,.$$

Da die η_k noch nicht bekannt sind, bietet es sich wiederum an, maximale Exaktheit dieser Quadraturformel als Konstruktionsprinzip zu verwenden:

$$\sum_{k=1}^{s} a_{jk} p(c_k) \;\approx\; \int_0^{c_j} p(t)\,dt\,. \tag{8.2}$$

Man beachte, daß die Knoten $\{c_k\}$ bereits bestimmt sind. Daher ergibt sich die optimale Wahl der Gewichte $\{a_{jk}\}$ wie in Abschnitt **??** mit den Lagrange-Grundpolynomen,

$$a_{jk} = \int_0^{c_j} l_k(t)\,dt\,, \qquad l_k(t) = \prod_{i=1,\ i\neq k}^{s} \frac{t - c_i}{c_k - c_i}\,, \tag{8.3}$$

und die resultierende Quadraturformel (8.2) hat Exaktheitsgrad $s-1$.

Das aus (8.1), (8.3) resultierende implizite Runge-Kutta Verfahren heißt s-stufiges **(Runge-Kutta-)Gauß-Verfahren**.

Beispiel. Das einfachste Gauß-Verfahren ($s = 1$) führt auf die sogenannte **implizite Mittelpunktsregel**. Das erste Legendre-Polynom hat eine Nullstelle genau in der Mitte des Intervalls $c_1 = 1/2$ und das zugehörige Gewicht, mit dem die Quadraturformel (8.1) auf den Exaktheitsgrad $q = 1$ kommt, ist gerade $b_1 = 1$. a_{11} ergibt sich nach (8.3) zu $a_{11} = \int_0^{1/2} 1\,dt = 1/2$. Das zugehörige Tableau ist somit

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

und das Einschrittverfahren hat die Form

$$y_{i+1} = y_i + hf(t_i + h/2, \eta_1)\,, \qquad \eta_1 = y_i + \frac{h}{2} f(t_i + h/2, \eta_1)\,, \tag{8.4}$$

bzw. durch Kombination dieser beiden Gleichungen,

$$y_{i+1} = y_i + hf\big(t_i + h/2, (y_i + y_{i+1})/2\big)\,.$$

Man kann die implizite Mittelpunktsregel auch als eine Kombination des impliziten und des expliziten Euler-Verfahrens interpretieren: Demnach wird in (8.4) zunächst in einem ersten Halbschritt mit dem impliziten Euler-Verfahren eine erste Näherung η_1 für $y(t_i+h/2)$ berechnet; diese Näherung dient dann einem zweiten Halbschritt zur Berechnung von y_{i+1} mit dem expliziten Euler-Verfahren als Ausgangspunkt.

Die implizite Mittelpunktsregel wird insbesondere bei partiellen Differentialgleichungen wie der Wärmeleitungsgleichung gerne eingesetzt und läuft in diesem Kontext unter dem Namen **Crank-Nicolson** Verfahren.

Für die späteren Resultate wird es sich als nützlich erweisen, die Fehlerabschätzung aus Satz **??** für den Quadraturfehler auf die hier vorliegende Situation anzupassen.

Lemma 8.1 *Sei $g \in C^{2s}[0,T]$ eine reellwertige Funktion und $[\tau, \tau+h] \subset [0,T]$. Dann gilt für die Gauß-Legendre Quadraturformel* (8.1) *die folgende Fehlerabschätzung:*

$$\Big| h \sum_{j=1}^{s} b_j g(\tau + c_j h) \;-\; \int_\tau^{\tau+h} g(t)\,dt \Big| \;\le\; \kappa_s \, \|g^{(2s)}\|_{[0,T]}\, h^{2s+1}\,.$$

Dabei ist $\kappa_s = \frac{1}{2s+1}\, s!^4/(2s)!^3$.

Beweis. Sei $G(t) := g(\tau + th)$, $0 \le t \le 1$. Dann ist nach Satz **??**

$$\begin{aligned} &\Big| h \sum_{j=1}^{s} b_j g(\tau + c_j h) \;-\; \int_\tau^{\tau+h} g(t)\,dt \Big| \\ &\qquad = \; h \Big| \sum_{j=1}^{s} b_j G(c_j) \;-\; \int_0^1 G(t)\,dt \Big| \;\le\; \frac{\gamma_s^{-2}}{(2s)!}\, \|G^{(2s)}\|_{[0,1]}\, h\,. \end{aligned}$$

Hierbei ist γ_s der Höchstkoeffizient des s-ten Orthonormalpolynoms zur Gewichtsfunktion $w \equiv 1$ über $[0,1]$. Man überlegt sich leicht, daß dieses Orthonormalpolynom durch $\sqrt{2}u_s(2t-1)$ gegeben ist, wenn u_s das orthonormierte Legendre-Polynom bezeichnet. Nach Bemerkung **??** ist daher $\gamma_s = 2^s\sqrt{2s+1}\,(2s)!/(2^s s!^2)$. Daraus folgt nun aber unmittelbar die Behauptung, denn nach der Kettenregel ist $G^{(k)}(t) = h^k g^{(k)}(\tau + th)$. □

Wir benötigen außerdem noch den folgenden Hilfssatz, in dem Π_s^d den Raum aller Funktionen mit Werten im $\mathbb{R}^d$ bezeichnet, deren einzelne Komponenten Polynome vom Grad s sind.

Lemma 8.2 *Sei $f \in C^{2s}(I \times J)$ und h so klein, daß die Näherungen y_i des s-stufigen Gauß-Verfahrens wohldefiniert sind (vgl. Satz 5.7). Dann existiert zu jedem $t_i \in I$ ein Polynom $p \in \Pi_s^d$ mit*

$$\begin{aligned} p(t_i) = y_i\,, \qquad & p(t_i + h) = y_{i+1}\,, \\ p'(t_i + c_j h) = f\big(t_i + c_j h, p(t_i + c_j h)\big)\,, \qquad & j = 1, \ldots, s\,. \end{aligned}$$

Das Polynom p erfüllt also die Differentialgleichung $p'(t) = f(t, p(t))$ punktweise an den *isolierten* Knoten $t_i + c_j h$, $j = 1, \ldots, s$. Man sagt, das Polynom p **kollokiert** die Differentialgleichung an den vorgegebenen Knoten ($\rightsquigarrow$ **Kollokationsverfahren**).

Beweis. Wir wählen für p' das (komponentenweise definierte) Interpolationspolynom in Π_{s-1}^d mit $p'(t_i + c_j h) = f(t_i + c_j h, \eta_j)$, $j = 1, \ldots, s$. Wegen des Exaktheitsgrads $q = s - 1$ der Quadraturformeln (8.2) ergibt sich

$$\eta_j \;=\; y_i + h\sum_{k=1}^{s} a_{jk} f(t_i + c_k h, \eta_k) \;=\; y_i + h\sum_{k=1}^{s} a_{jk} p'(t_i + c_k h) \;=\; y_i + \int_{t_i}^{t_i + c_j h} p'(t)\,dt\,.$$

Dies ist gerade der Wert derjenigen Stammfunktion p von p' an der Stelle $t_i + c_j h$, die den Anfangswert $p(t_i) = y_i$ durchläuft. Demnach ist also $p(t_i + c_j h) = \eta_j$, $j = 1, \ldots, s$, und folglich

$$p'(t_i + c_j h) \;=\; f(t_i + c_j h, \eta_j) \;=\; f\big(t_i + c_j h, p(t_i + c_j h)\big)\,.$$

Damit ist der zweite Teil der Behauptung nachgewiesen. Der verbleibende Teil $p(t_{i+1}) = y_{i+1}$ folgt, da der Exaktheitsgrad der Quadraturformel (8.1) größer als $s - 1$ ist:

$$\begin{aligned} y_{i+1} \;&=\; y_i + h\sum_{j=1}^{s} b_j f(t_i + c_j h, \eta_j) \;=\; p(t_i) + h\sum_{j=1}^{s} b_j p'(t_i + c_j h) \\ &=\; p(t_i) + \int_{t_i}^{t_{i+1}} p'(t)\,dt \;=\; p(t_{i+1})\,. \end{aligned}$$

□

Bemerkung. In der Formulierung des Lemmas wurde das Resultat aus Satz 5.7 angesprochen, wonach für jedes implizite Runge-Kutta Verfahren bei hinreichend glatter rechten Seite f alle Näherungen y_i für hinreichend kleines $h > 0$ wohldefiniert sind. Für dissipative Differentialgleichungen, also Differentialgleichungen, bei denen f einer Lipschitz Bedingung (2.2) mit einem negativen l genügt, läßt sich zeigen, daß die Näherungen der Gauß-Verfahren für *alle* $h > 0$ wohldefiniert sind. Ein Resultat dieser Art haben wir bereits in Satz 4.1 für das implizite Euler-Verfahren bewiesen (dies ist allerdings *kein* Gauß-Verfahren).

Mit Hilfe von Lemma 8.2 läßt sich nun weiter zeigen, daß die Gauß-Verfahren die maximal mögliche Konvergenzordnung eines s-stufigen Verfahrens realisieren.

Zuvor erinnern wir uns aber kurz an die Methode der **Variation der Konstanten**: Ist $y' = A(t)y$ eine lineare Differentialgleichung, dann hängt die Lösung y offensichtlich linear vom Anfangswert y_0 zur Zeit $t = t_0$ ab:

$$y(t) = Y(t, t_0)\, y_0\,, \quad t \geq t_0\,; \qquad Y(t,t) = I\,, \quad t \geq 0\,.$$

Für ein konstantes A ergibt sich beispielsweise $Y(t, t_0) = e^{A(t-t_0)}$. Durch Differenzieren erkennt man, daß

$$A(t)\, y \;=\; y' \;=\; \frac{\partial}{\partial t} Y(t, t_0)\, y_0 \qquad \rightsquigarrow \qquad A(t) Y(t, t_0) = \frac{\partial}{\partial t} Y(t, t_0)\,. \tag{8.5}$$

Betrachten wir nun die *inhomogene* Differentialgleichung

$$y' = A(t)\, y + g(t)\,, \qquad y(0) = y_0\,,$$

dann ergibt sich die Lösungsformel

$$y(t) = Y(t,0)\, y_0 + \int_0^t Y(t,\tau) g(\tau)\, d\tau\,. \tag{8.6}$$

Davon überzeugt man sich durch nachrechnen:

$$\begin{aligned} y' \quad &= \quad \frac{\partial}{\partial t} Y(t,0)\, y_0 + Y(t,t)\, g(t) + \int_0^t \frac{\partial}{\partial t} Y(t,\tau) g(\tau)\, d\tau \\ &\overset{(8.5)}{=} \quad A(t) Y(t,0)\, y_0 + g(t) + \int_0^t A(t) Y(t,\tau) g(\tau)\, d\tau \\ &= \quad g(t) + A(t) \Big(Y(t,0)\, y_0 + \int_0^t Y(t,\tau) g(\tau)\, d\tau \Big) \;=\; g(t) + A(t)\, y(t)\,. \end{aligned}$$

Man beachte, daß das Integral in (8.6) die Änderung der Lösung aufgrund der Inhomogenität beschreibt.

Satz 8.3 *Die Ordnung des s-stufigen Gauß-Verfahrens ist $q = 2s$.*

Beweisskizze. OBdA sei $h > 0$ so klein gewählt, daß alle Zeitschritte des Gauß-Verfahrens wohldefiniert sind. Bezeichnet dann p das Kollokationspolynom aus Lemma 8.2, so gilt offensichtlich

$$p' = f(t,p) + \varepsilon(t)\,, \qquad t_i \le t \le t_i + h\,, \tag{8.7}$$

mit $\varepsilon(t_i + c_j h) = 0$, $j = 1, \dots, s$. Wir gehen wie üblich davon aus, daß $p(t_i) = y(t_i)$ auf der Lösungskurve liegt und fragen nach dem Fehler $\|p(t_{i+1}) - y(t_{i+1})\|_2 = \|y_{i+1} - y(t_{i+1})\|_2$.

Dazu fassen wir (8.7) als *inhomogene* Differentialgleichung auf ($y' = f(t,y)$ ist die zugehörige homogene Differentialgleichung) und suchen eine Verallgemeinerung der Formel der Variation der Konstanten (8.6) auf die vorliegende nichtlineare Situation. Dazu betrachten wir für festes $\theta \in [0,1]$ die Differentialgleichung

$$u' = f(t,u) + \theta\varepsilon(t)\,, \qquad u(t_i) = y_i\,. \tag{8.8}$$

Wir bezeichnen die Lösung mit $u(t,\theta)$, $t_i \le t \le t_i + h$, $0 \le \theta \le 1$. Offensichtlich ist $u(t,0) = y(t)$ und $u(t,1) = p(t)$. Dies ergibt

$$p(t) - y(t) \;=\; u(t,1) - u(t,0) \;=\; \int_0^1 u_\theta(t,\theta)\,d\theta\,. \tag{8.9}$$

Differentiation von (8.8) nach θ ergibt

$$u'_\theta = \underbrace{f_u(t,u)}_{= A(t;\theta) \in \mathbb{R}^{d\times d}} u_\theta + \varepsilon(t)\,, \qquad u_\theta(t_i) = 0\,.$$

Hierauf kann die Formel (8.6) der Variation der Konstanten angewendet werden (die "homogene Lösung" ist identisch Null wegen des Anfangswerts) und es ergibt sich

$$u_\theta(t,\theta) = \int_{t_i}^{t} Y(t,\tau;\theta)\varepsilon(\tau)\,d\tau$$

mit einem geeigneten Lösungsoperator $Y(\,\cdot\,,\,\cdot\,;\theta)$. Einsetzen in die Fehlerdarstellung (8.9) ergibt

$$y_{i+1} - y(t_i+h) \;=\; \int_0^1 \int_{t_i}^{t_i+h} Y(t_i+h,\tau;\theta)\varepsilon(\tau)\,d\tau\,d\theta \;=\; \int_{t_i}^{t_i+h} \underbrace{\int_0^1 Y(t_i+h,\tau;\theta)\,d\theta}_{=:\,g(\tau)}\,\varepsilon(\tau)\,d\tau\,,$$

und dieses Integral kann gemäß Lemma 8.1 durch die Gauß-Quadraturformel abgeschätzt werden,

$$y_{i+1} - y(t_i+h) \;=\; h\sum_{j=1}^{s} b_j g(t_i+c_jh)\varepsilon(t_i+c_jh) \;+\; O(h^{2s+1}) \;=\; O(h^{2s+1})\,,$$

wobei im letzten Schritt verwendet wurde, daß $\varepsilon(t_i + c_j h) = 0$ für $j = 1, \dots, s$.

Leider ist der Beweis so nicht vollständig, da die Konstante in dem O-Term nach Lemma 8.1 von der $2s$-ten Ableitung von $g\varepsilon$ und damit implizit von dem unbekannten Polynom p selbst abhängt. Allerdings läßt sich mit erheblich mehr Aufwand zeigen, daß die "O-Konstante" tatsächlich beschränkt bleibt. □

Insbesondere hat also die implizite Mittelpunktsregel bzw. das Crank-Nicolson Verfahren die Ordung $q = 2$ und ist damit dem impliziten Euler Verfahren (zumindest in dieser Hinsicht) überlegen.

Die Stabilitätsfunktion der impliziten Mittelpunktsregel ist auch schnell ausgerechnet: Nach Satz 6.2 ist

$$R(\zeta) \;=\; 1+\zeta(1-\frac{1}{2}\,\zeta)^{-1} \;=\; \frac{1+\zeta/2}{1-\zeta/2} \;=\; 1+\zeta+\tfrac{1}{2}\,\zeta^2+\tfrac{1}{4}\,\zeta^3+\ldots\,.$$

Man sieht nun unmittelbar daß die Stabilitätsfunktion R eine Möbiustransformation ist, die die imaginäre Achse auf den Einheitskreisrand und die linke Halbebene von $\mathbb{C}$ auf das Innere des Einheitskreises abbildet. Also ist die implizite Mittelpunktsregel nach Satz 6.5 A-stabil und zudem Isometrie erhaltend.

Das zweite zentrale Ergebnis dieses Abschnitts ist nun, daß beide genannten Eigenschaften auf *alle* Gauß-Verfahren zutreffen.

Satz 8.4 *Alle Gauß-Verfahren sind A-stabil und Isometrie erhaltend.*

Beweis. Betrachten wir die übliche Testgleichung $y' = \lambda y$ mit $y_0 = 1$ und $\lambda \in \mathbb{C}$ und wählen als Schrittweite für das Runge-Kutta Verfahren $h = 1$. Wie im Beweis von Satz 6.2 gesehen, ergibt sich $y_1 = R(\lambda)$ mit

$$R(\lambda) \;=\; 1+\lambda b^T(I-\lambda A)^{-1}\mathbf{1}\,,$$

sofern λ kein Pol von R ist. Das ist bis auf höchstens s Ausnahmewerte auch der Fall, nämlich bis auf die Kehrwerte der Eigenwerte von A.

Sei nun $\lambda \in \mathbb{C}$ keiner dieser Ausnahmewerte. Dann existieren neben der Lösung $y_1 = R(\lambda)$ natürlich auch die Zwischenpunkte η_j, $j = 1, \ldots, s$, des Runge-Kutta Verfahrens. Wir greifen nun auf das in Lemma 8.2 konstruierte Kollokationspolynom p zu diesem Runge-Kutta Schritt zurück und setzen $q := |p|^2 = \overline{p}p$. Demnach ist

$$|R(\lambda)|^2 \;=\; |y_1|^2 \;=\; |p(1)|^2 \;=\; q(1) \;=\; q(0)+\int_0^1 q'(\tau)\,d\tau \;=\; 1+2\,\mathrm{Re}\int_0^1 \overline{p(\tau)}p'(\tau)\,d\tau\,.$$

Da $\overline{p}p' \in \Pi_{2s-1}$, kann das Integral mit der Gauß-Quadraturformel exakt ausgewertet werden, d.h.,

$$|R(\lambda)|^2 \;=\; 1+2\,\mathrm{Re}\sum_{j=1}^{s} b_j\overline{p(c_j)}p'(c_j)\,.$$

Die Gewichte b_j der Gaußformeln sind immer positiv (vgl. Satz **??**); außerdem folgt aus der Kollokationseigenschaft von p, daß $p'(c_j) = f(c_j, p(c_j)) = \lambda p(c_j)$. Folglich ist

$$|R(\lambda)|^2 \;=\; 1+2\,\mathrm{Re}\;\lambda\sum_{j=1}^{s} b_j|p(c_j)|^2\,,$$

und $|R(\lambda)|$ ist genau dann kleiner, größer oder gleich eins, wenn Re λ kleiner, größer oder gleich Null ist.

Bisher haben wir die Polstellen von R außer acht gelassen. Es ist aber aus Stetigkeitsgründen offensichtlich, daß diese nur in der rechten Halbebene liegen können und damit ist der Satz vollständig bewiesen. □

Zum Abschluß seien noch die Runge-Kutta Tableaus der Gauß-Verfahren der Ordnung $q = 4$ und $q = 6$ angeführt:

$$\begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \qquad \begin{array}{c|ccc} \frac{1}{2} - \frac{\sqrt{15}}{10} & \frac{5}{36} & \frac{2}{9} - \frac{\sqrt{15}}{15} & \frac{5}{36} - \frac{\sqrt{15}}{30} \\ \frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\ \frac{1}{2} + \frac{\sqrt{15}}{10} & \frac{5}{36} + \frac{\sqrt{15}}{30} & \frac{2}{9} + \frac{\sqrt{15}}{15} & \frac{5}{36} \\ \hline & \frac{5}{18} & \frac{4}{9} & \frac{5}{18} \end{array}$$

Zum Abschluß dieses Paragraphen skizzieren wir eine effiziente *Implementierung* der Gauß-Verfahren (bzw. allgemeiner impliziter Runge-Kutta Verfahren). Der Hauptaufwand steckt dabei in der Lösung des nichtlinearen Gleichungssystems

$$\eta_j = y_i + h \sum_{\nu=1}^{s} a_{j\nu} f(t_i + c_\nu h, \eta_\nu) \,, \qquad j = 1, \ldots, s \,.$$

Dieses Gleichungssystem wird in der Regel iterativ mit einem Newton-artigen Verfahren gelöst. Auf diese Weise ergeben sich Näherungen für η_ν, aus denen dann die Steigungen $f(t_i + c_\nu h, \eta_\nu)$ für den Schlußschritt

$$y_{i+1} = y_i + h \sum_{\nu=1}^{s} b_\nu f(t_i + c_\nu h, \eta_\nu) \tag{8.10}$$

berechnet werden.

Mit einem Trick kann man die zusätzlichen Funktionsauswertungen $f(t_i + c_\nu h, \eta_\nu)$ für den Schlußschritt (8.10) vermeiden: Dazu führen wir Hilfsvariablen (Steigungen) k_j durch

$$k_j = f(t_i + c_j h, \eta_j)$$

ein und erhalten auf diese Weise das äquivalente Gleichungssystem

$$\begin{aligned} k_j &= f\Big(t_i + c_j h, y_i + h \sum_{\nu=1}^{s} a_{j\nu} k_\nu\Big) \,, \qquad j = 1, \ldots, s \,, \\ y_{i+1} &= y_i + h \sum_{\nu=1}^{s} b_\nu k_\nu \,. \end{aligned} \tag{8.11}$$

Die Lösung des $sd \times sd$-dimensionalen Gleichungssystems (8.11) für die $\{k_j\}$ geschieht, wie bereits angesprochen, mit einem Newton-artigen Verfahren, und zwar verwendet man meistens ein Quasi-Newton Verfahren wie in Algorithmus 4.4, bei dem anstelle der exakten Ableitungen $f_y(t_i + c_j h, y_i + h \sum_{\nu=1}^{s} a_{j\nu} k_\nu)$ von f die "eingefrorene" Näherung $J = f_y(t_i, y_i)$ verwendet wird. Mit der abkürzenden Schreibweise

$$\mathbf{k} = \Phi(\mathbf{k}h)\,, \qquad \mathbf{k} = \begin{bmatrix} k_1 \\ \vdots \\ k_s \end{bmatrix} \in \mathbb{R}^{sd}\,,$$

für das nichtlineare Gleichungssystem (8.11) ergibt dies die folgende (eingefrorene) Jacobi-Matrix $\mathrm{D}\Phi$ von Φ:

$$\mathrm{D}\Phi = \begin{bmatrix} a_{11}J & a_{12}J & \dots & a_{1s}J \\ a_{21}J & a_{22}J & & a_{2s}J \\ \vdots & & & \vdots \\ a_{s1}J & a_{s2}J & \dots & a_{ss}J \end{bmatrix}\,. \tag{8.12}$$

Als Startnäherung für die Iteration bietet sich $k_j = f(t_i, y_i)$, $j = 1, \dots, s$, oder der Wert von k_j aus dem vorhergegangenen Zeitschritt an. Insgesamt ergibt sich somit der folgende Algorithmus:

Algorithmus 8.5

- Berechne $J = f_y(t_i, y_i)$ und $\mathrm{D}\Phi$ gemäß (8.12)
- Setze $k_j^{(0)} = f(t_i, y_i),\ \ j = 1, \dots, s$
- `for` $n = 0, 1, \dots$ `do`

  Löse $(I - h\mathrm{D}\Phi)\Delta\mathbf{k}^{(n)} = \Phi(\mathbf{k}^{(n)}h) - \mathbf{k}^{(n)}$

  Setze $\mathbf{k}^{(n+1)} = \mathbf{k}^{(n)} + \Delta\mathbf{k}^{(n)}$

  `until stop.`
- $y_{i+1} = y_i + h \sum_{j=1}^{s} b_j k_j$

Die Abbruchbedingung für die Quasi-Newton Iteration kann dabei anhand ähnlicher Heuristiken wie in Abschnitt 4 erfolgen.

9 Rosenbrock Verfahren

In diesem Abschnitt führen wir eine Klasse von Einschrittverfahren ein, die zwar nicht mehr zu den Runge-Kutta Verfahren gehören, aber aus ihnen abgeleitet werden können. Wir beschränken uns dabei ausschließlich auf *autonome Differentialgleichungen*,

$$y' = f(y)\,, \qquad y(0) = y_0\,.$$

Erinnerung: Ein Runge-Kutta Verfahren ist *explizit*, falls die Koeffizientenmatrix A eine strikte untere Dreiecksgestalt hat; andernfalls ist das Verfahren implizit. Die nichtlinearen Gleichungssysteme, die im impliziten Fall gelöst werden müssen – etwa (8.11) für die Stufen k_j – haben im allgemeinen die Dimension $sd \times sd$. Nicht ganz so schlimm ist die Situation, wenn die Matrix A eine untere Dreiecksmatrix mit nicht-verschwindender Diagonalen ist. In diesem Fall *entkoppeln* die s Gleichungen für die Stufen,

$$k_j = f\Big(y_i + h\sum_{\nu=1}^{j-1} a_{j\nu}k_\nu + a_{jj}hk_j\Big)\,, \qquad j = 1,\ldots,s\,, \tag{9.1}$$

und können daher sequentiell gelöst werden.

Damit reduziert sich der Aufwand auf die Lösung von s lediglich d-dimensionalen nichtlinearen Gleichungssystemen. Durch Linearisierung der j-ten Gleichung ergibt sich das Newton-Verfahren

$$(I - a_{jj}hJ)\,\big(k_j^{(n+1)} - k_j^{(n)}\big) \;=\; f\Big(y_i + h\sum_{\nu=1}^{j-1} a_{j\nu}k_\nu + a_{jj}hk_j^{(n)}\Big) - k_j^{(n)}\,,$$

bzw.

$$(I - a_{jj}hJ)k_j^{(n+1)} \;=\; f\Big(y_i + h\sum_{\nu=1}^{j-1} a_{j\nu}k_\nu + a_{jj}hk_j^{(n)}\Big) - a_{jj}hJk_j^{(n)}\,,$$

wobei in der Regel wieder die Näherungsableitung $J = f_y(y_i)$ anstelle der Ableitung an der exakten y-Koordinate verwendet wird. Für hinreichend kleine h sind diese linearen Gleichungssysteme eindeutig lösbar.

Häufig ist es für die Genauigkeit völlig ausreichend, lediglich einen einzigen Iterationsschritt durchzuführen – zumindest bei hinreichend genauer Startnäherung $k_j^{(0)}$. Denkbar wäre etwa eine Startnäherung der Form $k_j^{(0)} = \sum_{\nu=1}^{j-1} d_{j\nu}/a_{jj}k_\nu$. Dies führt auf das folgende Verfahren

$$(I - a_{jj}hJ)k_j \;=\; f\Big(y_i + h\sum_{\nu=1}^{j-1}(a_{j\nu} + d_{j\nu})k_\nu\Big) - hJ\sum_{\nu=1}^{j-1} d_{j\nu}k_\nu\,, \quad j = 1,\ldots,s\,. \tag{9.2}$$

Verfahren dieser Struktur werden gelegentlich auch **linear implizite** Runge-Kutta Verfahren genannt: Sie sind implizit, da zur Bestimmung jeder einzelnen Stufe k_j ein Gleichungssystem gelöst werden muß; diese Gleichungssysteme sind jedoch lediglich linear. Daher ist der Arbeitsaufwand um ein Vielfaches niedriger als bei (echt) impliziten Runge-Kutta Verfahren. In gewisser Weise stellen linear implizite Verfahren also einen Kompromiß aus expliziten und impliziten Runge-Kutta Verfahren dar. Dabei erben sie von den expliziten Verfahren den moderaten Arbeitsaufwand. Wie wir nun zeigen wollen, erben sie gleichzeitig den Hauptvorteil der impliziten Verfahren: die Stabilität.

Proposition 9.1 *Sei (A, b, c) ein Runge-Kutta Verfahren mit nichtsingulärer linker unterer Dreiecksmatrix $A \in \mathbb{R}^{s\times s}$. Dann hat das daraus abgeleitete linear implizite Verfahren*

$$y_{i+1} = y_i + h \sum_{j=1}^{s} b_j k_j \,,$$

mit Stufen k_j gemäß (9.2) die gleiche Stabilitätsfunktion wie das Ausgangsverfahren – unabhängig von der Wahl der $d_{j\nu}$.

Beweis. Wir müssen das linear implizite Verfahren auf die eindimensionale Testgleichung $y' = \lambda y$, $y_0 = 1$, anwenden. Wegen $f(y) = \lambda y$ ergibt sich in diesem Fall $J = \lambda$ und die j-te Gleichung der Rekursion (9.2) nimmt die Form

$$(1 - a_{jj}h\lambda)k_j \;=\; \lambda + h\lambda \sum_{\nu=1}^{j-1} (a_{j\nu} + d_{j\nu})k_\nu - h\lambda \sum_{\nu=1}^{j-1} d_{j\nu}k_\nu \;=\; \lambda + h\lambda \sum_{\nu=1}^{j-1} a_{j\nu}k_\nu$$

an. Multiplikation mit h und Substitution von $\zeta = h\lambda$ ergibt folglich

$$(1 - a_{jj}\zeta)k_j h \;=\; \zeta + \zeta \sum_{\nu=1}^{j-1} a_{j\nu}k_\nu h \,, \qquad j = 1, \ldots, s \,.$$

Wir bringen alle $k_\nu h$ auf die linke Seite; mit $\mathbf{1} = [1, \ldots, 1]^T \in \mathbb{R}^s$ erhält man daraufhin

$$\underbrace{k_j h - \zeta \sum_{\nu=1}^{j} a_{j\nu}k_\nu h}_{k_j h - \zeta(A\mathbf{k}h)_j} \;=\; \zeta \;=\; (\zeta \mathbf{1})_j \,, \qquad j = 1, \ldots, s \,,$$

bzw. $(I - \zeta A)\mathbf{k}h = \zeta\mathbf{1}$. Eingesetzt in die Definition von y_1 folgt daraus

$$y_1 \;=\; 1 + b^T \mathbf{k}h \;=\; 1 + \zeta b^T (I - \zeta A)^{-1}\mathbf{1} \,.$$

Dies ist aber gerade die Stabilitätsfunktion des Runge-Kutta Ausgangsverfahrens, vgl. Definition 6.2. □

Da einige der wichtigsten Stabilitätseigenschaften gemäß Satz 6.5 allein von der Stabilitätsfunktion abhängen, übertragen sich entsprechende Eigenschaften eines impliziten Runge-Kutta Verfahrens auf das dazugehörige linear implizite Verfahren.

Beispiel. Zur Herleitung des **linear-impliziten Euler-Verfahrens** verwenden wir die Koeffizienten $A = 1$, $b = 1$ und $c = 1$, vgl. Beispiel 5.3. Demnach ergibt sich aus (9.2) das Verfahren

$$y_{i+1} = y_i + hk \,, \qquad (I - hf_y(y_i))\, k = f(y_i) \,. \tag{9.3}$$

Wie das implizite Euler-Verfahren ist auch das linear-implizite Euler-Verfahren A-stabil. Im Gegensatz zum impliziten Euler-Verfahren muß aber bei der linear-impliziten Variante in jedem Zeitschritt lediglich ein lineares Gleichungssystem gelöst werden.

Für die Praxis ergibt sich noch eine erhebliche Vereinfachung, wenn man für alle Stufen den gleichen Koeffizienten $a_{jj} = a$ verwendet. In diesem Fall haben alle linearen Gleichungssysteme (9.2) die gleiche Koeffizientenmatrix $I - ahJ$ und daher wird lediglich die LR-Zerlegung einer einzigen Matrix benötigt. Meist ist es sogar möglich, die *selbe* Jacobi-Matrix J über mehrere Zeitschritte hinweg zu verwenden. Auf diese Weise spart man sich zusätzlichen Aufwand.

Löst man sich von der bisherigen Bedeutung der Koeffizienten $a_{j\nu}$ und vereinfacht die Definition (9.2) ein wenig, dann erhält man schließlich das allgemeine Schema eines **Rosenbrock-Verfahrens**:

Algorithmus 9.2 (Rosenbrock-Verfahren)

- Berechne $J = f_y(y_i)$ (nicht notwendig in jedem Zeitschritt)
- Zerlege $I - ahJ = LR$ mit dem Gauß-Algorithmus
- `for` $j = 1, \ldots, s$ `do`

  $$\text{Löse } (I - ahJ)k_j \;=\; f\Big(y_i + h\sum_{\nu=1}^{j-1} a_{j\nu}k_\nu\Big) \;-\; hJ\sum_{\nu=1}^{j-1} d_{j\nu}k_\nu$$

  `end for`
- $y_{i+1} = y_i + h\sum_{j=1}^{s} b_j k_j$.

Dabei sind a, $a_{j\nu}$, $d_{j\nu}$ und b_j geeignet zu wählende Parameter, die allein im Hinblick auf Ordnung und Stabilität des Verfahrens optimiert werden können.

Beispiel 9.3 Das Programmpaket MATLAB bietet das Rosenbrock-Verfahren `ode23s` (das wir bereits in Beispiel 7.2 kennengelernt haben) zur Lösung steifer Differentialgleichungen an. In seiner Grundform ist das Verfahren folgendermaßen definiert:

$$\begin{aligned}(I - ahJ)k_1 &= f(y_i)\,,\\ (I - ahJ)k_2 &= f(y_i + h\,\tfrac{1}{2}k_1) - ahJk_1\,,\\ y_{i+1} &= y_i + hk_2\,.\end{aligned}$$

Dabei ist $a = 1/(2+\sqrt{2})$ und $J = f_y(y_i)$. Offensichtlich ist dieses Verfahren ein Spezialfall von Algorithmus 9.2 mit den Parametern $a_{2,1} = 1/2$ und $d_{2,1} = a$, sowie $b_1 = 0$ und $b_2 = 1$.

Wir wollen im weiteren exemplarisch die Ordnung und die Stabilitätseigenschaften des Verfahrens `ode23s` untersuchen.

Satz 9.4 *Das Rosenbrock-Verfahren* `ode23s` *ist ein Verfahren zweiter Ordnung.*

Beweis. Wir beginnen mit der folgenden Beobachtung: Für $0 < h < 1/(2a\|J\|_2)$ ist die $s \times s$-Matrix $I - ahJ$ invertierbar, und für eine allgemeine Gleichung $(I - ahJ)k = f$ ergibt sich wegen $\|(I - ahJ)k\|_2 \geq \|k\|_2 - ah\|J\|_2\|k\|_2 \geq \frac{1}{2}\|k\|_2$ unmittelbar die Abschätzung

$$\|k\|_2 \leq 2\|f\|_2 .$$

Aus der Definition von k_1 folgt daher durch rekursives Einsetzen

$$k_1 = f + ahJk_1 = f + ahJ\big(f + ahJk_1\big) = f + ahJf + O(h^2) , \tag{9.4}$$

wobei wir wie schon früher das Argument y_i von f wieder weggelassen haben; dies werden wir im weiteren auch bei Ableitungen von f so halten. Mittels Taylorentwicklung ergibt sich nun in entsprechender Weise

$$\begin{aligned} k_2 &= f(y_i + h\,\tfrac{1}{2}k_1) - ahJk_1 + ahJk_2 \\ &= f + f_y h\,\tfrac{1}{2}k_1 - ahJk_1 + ahJk_2 + O(h^2) \\ &= f + h(\tfrac{1}{2}f_y - aJ)k_1 + ahJ\big(f + O(h)\big) + O(h^2) \\ &= f + h\big(aJf + (\tfrac{1}{2}f_y - aJ)k_1\big) + O(h^2) . \end{aligned}$$

Wieder unter Verwendung von (9.4) ergibt das schließlich

$$\begin{aligned} y_{i+1} &= y_i + hk_2 = y_i + hf + h^2\big(aJf + (\tfrac{1}{2}f_y - aJ)k_1\big) + O(h^3) \\ &= y_i + hf + h^2\big(aJf + (\tfrac{1}{2}f_y - aJ)f\big) + O(h^3) \\ &= y_i + hf + h^2\,\tfrac{1}{2}f_y f + O(h^3) . \end{aligned}$$

Da die Differentialgleichung autonom ist, also $f_t = 0$ gilt, ergibt ein Vergleich der obigen Entwicklung mit der Taylorentwicklung (5.6) der exakten Lösung eine Übereinstimmung bis auf den Term $O(h^3)$, d.h., das Rosenbrock-Verfahren `ode23s` ist ein Einschrittverfahren zweiter Ordnung. □

Man beachte, daß in diesem Beweis an keiner Stelle verwendet wurde, daß J die *exakte Ableitung* $f_y(y_i)$ ist. Mit anderen Worten: Das Verfahren `ode23s` ist selbst dann noch ein Verfahren zweiter Ordnung, wenn J nur eine Näherung an $f_y(y_i)$ darstellt (genau genommen ist noch nicht einmal das notwendig). Es besteht also kein unmittelbarer Zwang, die exakte Jacobi-Matrix *in jedem Zeitschritt* auszuwerten. Es reicht, J ab und an neu auszurechnen.

Das folgende Resultat zeigt, daß `ode23s` selbst bei exakter Jacobi-Matrix J kein Verfahren dritter Ordnung ist.

Lemma 9.5 *Für $J = f_y(y_i)$ hat die Stabilitätsfunktion $R(\zeta)$ des Rosenbrock-Verfahrens* `ode23s` *die Form*

$$R(\zeta) = \frac{1+(1-2a)\zeta}{(1-a\zeta)^2} = 1 + \zeta + \tfrac{1}{2}\zeta^2 + (\tfrac{1}{2}-a)\zeta^3 + O(\zeta^4)\,.$$

Dabei ist weiterhin $a = 1/(2+\sqrt{2})$.

Beweis. Für die Testgleichung $y' = \lambda y$ ist $f(y_i) = \lambda y_i$ und $J = f_y(y_i) = \lambda$. Für hinreichend kleines $h > 0$ und $y_i = 1$ ergibt sich somit

$$k_1 = \lambda/(1-ah\lambda)\,, \qquad k_2 = \big(\lambda(1+h\tfrac{1}{2}k_1) - ah\lambda k_1\big)/(1-ah\lambda)\,.$$

Durch Einsetzen in $y_{i+1} = y_i + hk_2 = 1 + hk_2$ ergibt das mit $\zeta = h\lambda$

$$\begin{aligned} y_{i+1} &= 1 + \frac{h\lambda + h^2\lambda k_1(\frac{1}{2}-a)}{1-ah\lambda} = 1 + \frac{\zeta(1-a\zeta)+\zeta^2(\frac{1}{2}-a)}{(1-a\zeta)^2} \\ &= 1 + \frac{\zeta + (\frac{1}{2}-2a)\zeta^2}{(1-a\zeta)^2} \equiv R(\zeta)\,. \end{aligned}$$

Somit folgt

$$R(\zeta) = \frac{(1-a\zeta)^2 + \zeta + (\frac{1}{2}-2a)\zeta^2}{(1-a\zeta)^2} = \frac{1+(1-2a)\zeta + (a^2-2a+\frac{1}{2})\zeta^2}{(1-a\zeta)^2}\,.$$

Da $a^2 - 2a + 1/2 = 0$ für das gegebene $a = 1/(2+\sqrt{2}) = 1 - 1/\sqrt{2}$ folgt unmittelbar die erste Behauptung.

Durch Einsetzen der geometrischen Reihe ergibt sich die zweite Darstellung von $R(\zeta)$:

$$\begin{aligned} R(\zeta) &= \big(1 + a\zeta + a^2\zeta^2 + a^3\zeta^3 + O(\zeta^4)\big)^2\big(1+(1-2a)\zeta\big) \\ &= \big(1 + 2a\zeta + 3a^2\zeta^2 + 4a^3\zeta^3 + O(\zeta^4)\big)\big(1+(1-2a)\zeta\big) \\ &= 1 + (2a+1-2a)\zeta + (3a^2+2a-4a^2)\zeta^2 + (4a^3+3a^2-6a^3)\zeta^3 + O(\zeta^4) \\ &= 1 + \zeta + (2a-a^2)\zeta^2 + (3a^2-2a^3)\zeta^3 + O(\zeta^4)\,. \end{aligned}$$

Wie bereits oben festgestellt wurde, ist $a^2 - 2a + 1/2 = 0$, also $2a - a^2 = 1/2$ und $3a^2 - 2a^3 = -2a(a^2-2a+1/2) - a^2 + a = 1/2 - a$. Folglich ist auch die zweite Darstellung nachgewiesen. □

Da $1/2 - a \neq 1/6$ stimmen also offensichtlich nur die ersten drei Terme der Entwicklung von $R(\zeta)$ mit der entsprechenden Taylorentwicklung der Exponentialfunktion überein. Nach Satz 6.4 kann `ode23s` also kein Verfahren dritter Ordnung sein.

Nun wenden wir uns der Stabilität dieses Rosenbrock-Verfahrens zu.

Satz 9.6 *Das Verfahren* `ode23s` *ist A-stabil, falls in jedem Zeitschritt die exakte Jacobi-Matrix $J = f_y(y_i)$ verwendet wird.*

Beweis. Wir müssen zeigen, daß $|R(\zeta)| \leq 1$ für alle $\zeta \in \mathbb{C}$ mit $\mathrm{Re}\,\zeta \leq 0$. Dazu betrachten wir die Stabilitätsfunktion auf der imaginären Achse, also $R(it)$ mit $t \in \mathbb{R}$. Dort gilt

$$|R(it)|^2 = \frac{\left|1 + i(1-2a)t\right|^2}{\left|1 - iat\right|^4} = \frac{1+(1-2a)^2t^2}{(1+a^2t^2)^2} = \frac{1+(1-4a+4a^2)t^2}{1+2a^2t^2+a^4t^4}\,.$$

Wegen $a = 1/(2+\sqrt{2})$ ergibt sich $1 - 4a + 4a^2 = 2a^2$ und folglich ist

$$|R(it)|^2 = \frac{1+2a^2t^2}{1+2a^2t^2+a^4t^4} \leq 1\,, \qquad t \in \mathbb{R}\,.$$

Da R lediglich eine Polstelle für $\zeta = 1/a = 2+\sqrt{2}$ besitzt und diese in der rechten Halbebene liegt, ist R eine analytische Funktion über der linken Halbebene, die zudem für $|\zeta| \to \infty$ gegen 0 strebt. Da R auf dem Rand dieser Halbebene (der imaginären Achse) durch 1 beschränkt ist, ist R nach dem Maximumprinzip ($\longrightarrow$ Funktionentheorie) in der ganzen linken Halbebene durch 1 beschränkt. Also ist `ode23s` nach Satz 6.5 A-stabil. □

Man beachte, daß dieses Resultat unter der Voraussetzung bewiesen wurde, daß die Jacobi-Matrix J in jedem Schritt exakt ausgerechnet wird. Mit anderen Worten: Eine inexakte Jacobi-Matrix J beeinflußt nicht die Konvergenzordnung aber möglicherweise das Stabilitätsverhalten von `ode23s`.

10 Schrittweitensteuerung

Für eine effiziente Implementierung der behandelten Einschrittverfahren ist es unerläßlich, sich über eine optimale Wahl der Schrittweite h Gedanken zu machen. Aus Kostengründen sollte h natürlich so groß wie möglich sein: Je größer h ist, desto weniger Zeitschritte sind nötig, um ein Zeitintervall $[0,T]$ vollständig zu durchlaufen. Andererseits ist der Fehler $y_i - y(t_i)$ im Intervall $[0,T]$ bei einem Verfahren der Ordnung q nach Satz 5.7 von der Größe $O(h^q)$, also an die Größe von h gekoppelt. Allerdings ist dies nicht die gesamte Wahrheit: Beispielsweise haben wir für das implizite Euler-Verfahren in Korollar 4.3 gesehen, daß die Konstante in dieser O-Abschätzung sehr klein sein kann, und zwar in Abhängigkeit von dem lokalen Verhalten der rechten Seite f. In solchen Situationen kann dann h natürlich größer gewählt werden, ohne eine vorgegebene Genauigkeit der Näherungslösung zu verletzen. Entsprechende Beobachtungen haben wir auch bei dem numerischen Beispiel 7.2 mit dem Code `ode23s` gemacht.

Damit entsteht für einen praktischen Code die Notwendigkeit, den Fehler der aktuellen Approximation zu schätzen, um ggf. Korrekturen an der Schrittweite vornehmen zu können oder gar um eine Näherung wegen mangelnder Genauigkeit zu verwerfen.

Solche Fehlerschätzer beruhen zumeist auf dem Ergebnis eines zweiten Verfahrens (Kontrollverfahrens). Das Kontrollverfahren ist in der Regel ein Verfahren *höherer Ordnung* mit

vergleichbaren (oder besseren) Stabilitätseigenschaften. Wir bezeichnen mit $\hat{y}_i$ die Näherungen des Kontrollverfahrens, während y_i weiterhin die Näherungen des Ausgangsverfahrens sind. Aufgrund dieser Konstruktion erwarten wir $\|\hat{y}_i - y(t_i)\|_2 \ll \|y_i - y(t_i)\|_2$, und mit der Dreiecksungleichung folgt daraus, daß

$$\delta_i \coloneqq \|y_i - \hat{y}_i\|_2 \leqslant \|y_i - y(t_i)\|_2 \pm \|\hat{y}_i - y(t_i)\|_2 \approx \|y_i - y(t_i)\|_2 \tag{10.1}$$

ein plausibler Fehlerschätzer ist.

Natürlich sollte das Kontrollverfahren nicht die Kosten des Codes in die Höhe treiben. Allerdings kann ein Code auch nicht billiger sein als die Implementierung des Kontrollverfahrens alleine. Daher verwendet man meist Verfahren, die die *selben* Stufen $f(x_i + c_j h, \eta_j)$ (bzw. k_j im Rosenbrock-Fall) verwenden. Auf diese Weise erspart man sich zusätzliche Funktionsauswertungen. Für Runge-Kutta Verfahren ergibt sich also das folgende Schema:

$$\begin{aligned} y_{i+1} &= y_i + h \sum_{j=1}^{s} b_j f(t_i + c_j h, \eta_j) \,, \\ \hat{y}_{i+1} &= y_i + h \sum_{j=1}^{s} \hat{b}_j f(t_i + c_j h, \eta_j) \,, \end{aligned} \tag{10.2}$$

wobei die Stufen $f(t_i + c_j h, \eta_j)$ wie in (5.3) berechnet werden, also

$$\eta_j = y_i + h \sum_{k=1}^{s} a_{jk} f(t_i + c_k h, \eta_k) \,.$$

Man spricht bei einer solchen Konstruktion auch von **eingebetteten Runge-Kutta Verfahren**.

Für die Schrittweitensteuerung muß der Benutzer eine Fehlertoleranz $\epsilon > 0$ vorgeben, und die Schrittweite wird dann so eingestellt, daß in jedem Schritt die Ungleichung $\delta_i \leq \epsilon$ erfüllt ist. Dabei ist wie zuvor $\delta_i = \|y_i - \hat{y}_i\|_2$ der jeweilige Fehlerschätzwert (10.1). Ist die Testungleichung $\delta_{i+1} \leq \epsilon$ im $(i+1)$-ten Schritt erfüllt, dann wird die Näherung y_{i+1} akzeptiert (manche Implementierungen bevorzugen die vermeintlich bessere Approximation $\hat{y}_{i+1}$) und ggf. die Schrittweite h für den nächsten Schritt modifiziert; ist die Testungleichung hingegen verletzt, dann muß der $(i+1)$-te Zeitschritt mit einer entsprechend kleineren Schrittweite wiederholt werden.

Die Modifikation der Schrittweite folgt in beiden Fällen nach dem gleichen Muster. Nehmen wir an, das Ausgangsverfahren habe die Ordnung q und das Kontrollverfahren die Ordnung $q+1$. Dann hat der lokale Fehler der beiden Verfahren (also der Fehler, der im $(i+1)$-ten Schritt entsteht, falls (t_i, y_i) auf der exakten Lösungskurve liegt) asymptotisch die Form

$$y_{i+1} - y(t_{i+1}) = h^{q+1} w_{i+1} + O(h^{q+2}) \,, \qquad \hat{y}_{i+1} - y(t_{i+1}) = O(h^{q+2}) \,,$$

mit einem (i.A. unbekannten) $w_{i+1} \in \mathbb{R}^d$, so daß

$$\delta_{i+1} = \|y_{i+1} - \hat{y}_{i+1}\|_2 = \|h^{q+1} w_{i+1} + O(h^{q+2})\|_2 \approx h^{q+1} \|w_{i+1}\|_2 \,.$$

Mit Schrittweite $\tilde{h}$ anstelle von h würde sich entsprechend

$$\tilde{h}^{q+1}\,\|w_{i+1}\|_2 \;=\; \big(\tilde{h}/h\big)^{q+1} h^{q+1}\,\|w_{i+1}\|_2 \;\approx\; \big(\tilde{h}/h\big)^{q+1}\delta_{i+1}$$

näherungsweise als Fehler ergeben. Folglich ist

$$\tilde{h} = \tau\,\big(\frac{\epsilon}{\delta_{i+1}}\big)^{1/(q+1)}\,h \tag{10.3}$$

mit $\tau = 1$ die größte Schrittweite, für die noch $\|y_{i+1} - \hat{y}_{i+1}\|_2 \lesssim \epsilon$ gilt. In diesem Sinne liefert $\tilde{h}$ einen optimalen Kompromiß aus Genauigkeit und (zukünftigem) Rechenaufwand. Daher ersetzt man die alte Schrittweite h durch $\tilde{h}$, wobei der Toleranzparameter τ in (10.3) in der Praxis meist ein wenig kleiner als 1 gewählt wird, etwa $\tau = 0.8$ oder 0.9. Ein praktischer Code könnte also wie folgt aussehen:

Algorithmus 10.1 (Schrittweitensteuerung)
q und $q+1$ seien die Ordnungen der beiden Verfahren; ϵ sei die Fehlertoleranz; $\tau \in [0.8, 0.9]$

- Wähle eine Startschrittweite $h > 0$ und setze $\delta := \epsilon$
- `for` $i = 0, \ldots$ `do`
- `repeat`
  setze $h := \tau\big(\epsilon/\delta\big)^{1/(q+1)} h$
  berechne y_{i+1} und $\hat{y}_{i+1}$ aus (10.2)
  `until` $\delta := \|y_{i+1} - \hat{y}_{i+1}\| \le \epsilon$
- `until` $t_{i+1} > T$

Bemerkungen. Da der obige Algorithmus auf einer Schätzung des *lokalen Fehlers* beruht, wird die Fehlerfortpflanzung bei dieser Vorgehensweise nicht berücksichtigt. Dies bedeutet, daß das Einschrittverfahren durchaus am Intervallende T einen Fehler oberhalb der vorgegebenen Fehlertoleranz haben kann. Eine gute Faustregel ist etwa der Schätzwert $T\epsilon$ für den maximalen Fehler in dem gesamten Integrationsintervall. Dies sollte vom Anwender immer berücksichtigt werden.

Im Zusammenhang mit Rosenbrock-Verfahren bedeutet eine Schrittweitenänderung auch immer, daß sich die Matrix $I - ahJ$ ändert, selbst wenn die alte Näherung J von $f_y(t_i, y_i)$ weiter benutzt werden soll. In dem Fall muß im zweiten Schritt von Algorithmus 9.2 eine neue LR-Zerlegung berechnet werden. Um diese Berechnung zu sparen, wird man bei Rosenbrock-Verfahren die Schrittweite nur dann modifizieren, wenn J ohnehin neu berechnet wird oder wenn die Fehlertoleranz ϵ deutlich unter- oder überschritten wird.

Die meisten Runge-Kutta Verfahren sind derart konstruiert, daß ihre Ordnung angesichts der Anzahl an Stufen $f(t_i + c_j h, \eta_j)$ größtmöglich ist. Dies bedeutet, daß das Kontrollverfahren –

welches ja die gleichen Stufen verwenden soll, vgl. (10.2) – in der Regel nur dann eine größere Ordnung haben kann, wenn es zusätzliche Stufen verwendet. Zusätzliche Stufen bedeuten aber zusätzliche Funktionsauswertungen, also zusätzlichen Aufwand.

Einen Ausweg aus diesem Problem bietet der sogenannte **Fehlberg-Trick**: Fehlberg hat vorgeschlagen, als zusätzliche Stufe die erste Stufe des folgenden Zeitschritts zu verwenden (meist ist das die Stufe $f(t_{i+1}, y_{i+1}) = f(t_i + h, y_{i+1})$, wobei h noch die alte Schrittweite ist). Man beachte, daß diese Stufe bei einem erfolgreichen Zeitschritt ohnehin berechnet wird; lediglich in dem Fall, daß die Berechnung von y_{i+1} mit kleinerer Schrittweite wiederholt werden muß (d.h., in einem gescheiterten Zeitschritt) führt der Fehlberg-Trick zu einem zusätzlichen Aufwand. Üblicherweise ist allerdings die überwiegende Mehrheit der Zeitschritte erfolgreich, so daß mit dieser Technik das Kontrollverfahren lediglich einen vernachlässigbaren zusätzlichen Aufwand mit sich bringt.

Beispiel 10.2 (klassisches Runge-Kutta Verfahren) Das klassische Runge-Kutta Verfahren der Ordnung 4 aus Beispiel 5.5 ist durch das Tableau

$$\begin{array}{c|c} c & A \\ \hline & \hat{b}^T \end{array} \quad = \quad \begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ 1 & 0 & 0 & 1 & \\ \hline & 1/6 & 1/3 & 1/3 & 1/6 \end{array}$$

gegeben; um den Aufwand für dieses Beispiel nicht zu hoch zu treiben, wählen wir dieses Verfahren als Kontrollverfahren und suchen nun ein eingebettetes Verfahren *dritter Ordnung*, das mit den selben Stufen auskommt. Nach Satz 5.4 müssen die Gewichte b_j, $j = 1, \ldots, 4$, neben der grundlegenden Gleichung $\sum_{j=1}^4 b_j = 1$ aus (5.1) noch die drei Gleichungen

$$\sum_{j=1}^4 b_j c_j = \frac{1}{2}\,, \qquad \sum_{j=1}^4 b_j c_j^2 = \frac{1}{3}\,, \qquad \sum_{j=1}^4 b_j \sum_{k=1}^4 a_{jk} c_k = \frac{1}{6}$$

erfüllen, also das folgende Gleichungssystem lösen:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1/2 & 1/2 & 1 \\ 0 & 1/4 & 1/4 & 1 \\ 0 & 0 & 1/4 & 1/2 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1/2 \\ 1/3 \\ 1/6 \end{bmatrix}.$$

Man sieht sofort, daß die Matrix nicht singulär ist und das Gleichungssystem daher nur eine Lösung hat, nämlich gerade die Gewichte des klassischen Runge-Kutta Verfahrens. Mit anderen Worten: Es gibt *kein* eingebettetes Einschrittverfahren dritter Ordnung – außer dem Kontrollverfahren selber.

Auch hier liefert der Fehlberg-Trick einen Ausweg: Wir fügen künstlich die fünfte Stufe $1 \;\big|\; \frac{1}{6}\ \frac{1}{3}\ \frac{1}{3}\ \frac{1}{6}\ 0$ in das Runge-Kutta Tableau ein,

$$\eta_5 = y_{i+1} = y_i + h \sum_{j=1}^{4} \hat{b}_j f(t_i + c_j h, \eta_j) \, .$$

und die Ordnungsbedingungen führen dann in entsprechender Weise auf das 4×5 Gleichungssystem

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1/2 & 1/2 & 1 & 1 \\ 0 & 1/4 & 1/4 & 1 & 1 \\ 0 & 0 & 1/4 & 1/2 & 1/2 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \begin{bmatrix} 1 \\ 1/2 \\ 1/3 \\ 1/6 \end{bmatrix}$$

für die Gewichte b_j, $j = 1, \ldots, 5$. Der Rang dieser Matrix bleibt natürlich 4 (es wurde ja nur eine neue Spalte hinten angefügt) und die Koeffizienten $\hat{b}_1 = 1/6$, $\hat{b}_2 = 1/3$, $\hat{b}_3 = 1/3$, $\hat{b}_4 = 1/6$ und $\hat{b}_5 = 0$ des klassischen Runge-Kutta Verfahrens bilden eine spezielle Lösung des Gleichungssystems. Alle anderen Lösungen ergeben sich durch Addition eines nichttrivialen Vertreters aus dem Nullraum dieser Matrix, wobei man sofort erkennt, daß dieser Nullraum durch den Vektor $[0, 0, 0, 1, -1]^T$ aufgespannt wird.

Ein möglicher Ansatz ist also

$$b = [1/6, 1/3, 1/3, 0, 1/6]^T \, , \qquad \hat{b} = [1/6, 1/3, 1/3, 1/6, 0]^T \, .$$

Der Fehlerschätzer (10.1) macht natürlich nur dann Sinn, wenn das eingebettete Verfahren selber *kein* Verfahren vierter Ordnung ist! Dies ergibt sich mit einer etwas aufwendigeren Rechnung analog zu Beispiel 6.3 ($\longrightarrow$ Übung), indem man zeigt, daß die Stabilitätsfunktion durch

$$\begin{aligned} R(\zeta) &= 1 + \zeta b^T (I - \zeta A)^{-1} \mathbf{1} \\ &= 1 + \zeta \, [1/6, 1/3, 1/3, 0, 1/6] \begin{bmatrix} 1 \\ 1 + \zeta/2 \\ 1 + \zeta/2 + \zeta^2/4 \\ 1 + \zeta + \zeta^2/2 + \zeta^3/4 \\ 1 + \zeta + \zeta^2/2 + \zeta^3/6 + \zeta^4/24 \end{bmatrix} \\ &= 1 + \zeta + \tfrac{1}{2}\zeta^2 + \tfrac{1}{6}\zeta^3 + \tfrac{1}{36}\zeta^4 + \tfrac{1}{144}\zeta^5 \end{aligned}$$

gegeben ist. Da der Koeffizient vor ζ^4 nicht der Koeffizient $1/24$ der Exponentialreihe ist, hat das eingebettete Runge-Kutta Verfahren nach Satz 6.4 bestenfalls dritte Ordnung. Aus (10.1) ergibt sich schließlich der folgende Fehlerschätzer:

$$\delta_{i+1} \, = \, \| y_{i+1} - \hat{y}_{i+1} \|_2 \, = \, \frac{h}{6} \, \| f(t_i + h, \hat{y}_{i+1}) - f(t_i + h, \eta_4) \|_2 \, .$$

Als zweites Beispiel betrachten wir den Rosenbrock-Code `ode23s` aus MATLAB.

Beispiel 10.3 (`ode23s`) In `ode23s` ist als Kontrollverfahren ein Verfahren dritter Ordnung implementiert, das ebenfalls den Fehlberg-Trick verwendet, also eine künstliche dritte Stufe einführt. Im autonomen Fall genügen diese drei Stufen von `ode23s` der Rekursion

$$\begin{aligned}(I-ahJ)k_1 &= f(y_i)\,,\\ (I-ahJ)k_2 &= f(y_i+h\tfrac{1}{2}k_1)-ahJk_1\,,\\ (I-ahJ)k_3 &= f(y_i+hk_2)-d_{3,1}hJk_1-d_{3,2}hJk_2\end{aligned}$$

mit $a=1/(2+\sqrt{2})$, $d_{3,1}=-(4+\sqrt{2})/(2+\sqrt{2})$ und $d_{3,2}=(6+\sqrt{2})/(2+\sqrt{2})$. Wie zuvor ist dann $y_{i+1}=y_i+hk_2$ die berechnete Näherung für $y(t_{i+1})$. Das zugehörige Kontrollverfahren ist durch

$$\hat{y}_{i+1}=y_i+\frac{h}{6}\,(k_1+4k_2+k_3)$$

gegeben. Für unsere Analyse nehmen wir im weiteren vereinfachend an, daß J die exakte Jacobi-Matrix $f_y(y_i)$ ist. Selbst wenn J nicht exakt ist, bleibt das Kontrollverfahren jedoch ein Verfahren dritter Ordnung.

Wie im Beweis von Satz 9.4 erhalten wir aus (9.4)

$$k_1=f+ahf_yk_1=f+ahf_yf+a^2h^2f_y^2f+O(h^3)$$

und entsprechend (unter Berücksichtigung, daß f von t unabhängig sein soll)

$$\begin{aligned}k_2 &= f+h\tfrac{1}{2}f_yk_1+h^2\tfrac{1}{8}k_1^*f_{yy}k_1-ahf_yk_1+ahf_yk_2+O(h^3)\\ &= f+h\tfrac{1}{2}f_y(f+ahf_yf)+h^2\tfrac{1}{8}f^*f_{yy}f-ahf_y(f+ahf_yf)+ahf_yk_2+O(h^3)\\ &= f+h\tfrac{1}{2}f_y(f+ahf_yf)+h^2\tfrac{1}{8}f^*f_{yy}f-ahf_y(f+ahf_yf)\\ &\quad +ahf_y\big(f+h\tfrac{1}{2}f_yf\underbrace{-ahf_yf+ahf_yk_2}_{=\,O(h^2)}\big)+O(h^3)\\ &= f+h(\tfrac{1}{2}-a+a)f_yf+h^2(\tfrac{1}{2}af_y^2f+\tfrac{1}{8}f^*f_{yy}f-a^2f_y^2f+\tfrac{1}{2}af_y^2f)+O(h^3)\\ &= f+h\tfrac{1}{2}f_yf+h^2\big((a-a^2)f_y^2f+\tfrac{1}{8}f^*f_{yy}f\big)+O(h^3)\,.\end{aligned}$$

Schließlich ist

$$\begin{aligned}k_3 &= f+hf_yk_2+h^2\tfrac{1}{2}k_2^*f_{yy}k_2-d_{3,1}hf_yk_1-d_{3,2}hf_yk_2+ahf_yk_3+O(h^3)\\ &= f+hf_y(f+h\tfrac{1}{2}f_yf)+h^2\tfrac{1}{2}f^*f_{yy}f-d_{3,1}hf_y(f+ahf_yf)-d_{3,2}hf_y(f+h\tfrac{1}{2}f_yf)\\ &\quad +ahf_yk_3+O(h^3)\\ &= f+h(1-d_{3,1}-d_{3,2})f_yf+h^2\big((\tfrac{1}{2}-d_{3,1}a-\tfrac{1}{2}d_{3,2})f_y^2f+\tfrac{1}{2}f^*f_{yy}f\big)\\ &\quad +ahf_yk_3+O(h^3)\,.\end{aligned}$$

Damit ergibt sich durch rekursives Einsetzen

$$
\begin{aligned}
k_3 &= f + h(1-d_{3,1}-d_{3,2})f_y f + h^2\big((\tfrac{1}{2}-d_{3,1}a-\tfrac{1}{2}d_{3,2})f_y^2 f + \tfrac{1}{2}f^* f_{yy} f\big) \\
&\quad + ahf_y\big(f + h(1-d_{3,1}-d_{3,2})f_y f + ahf_y k_3\big) + O(h^3) \\
&= f + h(1-d_{3,1}-d_{3,2}+a)f_y f \\
&\quad + h^2\big((\tfrac{1}{2}-2d_{3,1}a-d_{3,2}a-\tfrac{1}{2}d_{3,2}+a)f_y^2 f + \tfrac{1}{2}f^* f_{yy} f\big) + h^2 a^2 f_y^2 k_3 + O(h^3) \\
&= f + h(1-d_{3,1}-d_{3,2}+a)f_y f \\
&\quad + h^2\big((\tfrac{1}{2}-2d_{3,1}a-d_{3,2}a-\tfrac{1}{2}d_{3,2}+a+a^2)f_y^2 f + \tfrac{1}{2}f^* f_{yy} f\big) + O(h^3)\,.
\end{aligned}
$$

Da $d_{3,1}+d_{3,2}=2a$ und $\frac{1}{2}-d_{3,1}a-\frac{1}{2}d_{3,2}+a=2a^2$ (nachrechnen !), vereinfacht sich dies zu

$$
\begin{aligned}
k_3 &= f + h(1-a)f_y f + h^2\big((\tfrac{1}{2}-d_{3,1}a-\tfrac{1}{2}d_{3,2}+a-a^2)f_y^2 f + \tfrac{1}{2}f^* f_{yy} f\big) + O(h^3) \\
&= f + h(1-a)f_y f + h^2\big(a^2 f_y^2 f + \tfrac{1}{2}f^* f_{yy} f\big) + O(h^3)\,.
\end{aligned}
$$

Somit ergibt sich

$$
\begin{aligned}
\hat{y}_{i+1} &= y_i + \frac{h}{6}(k_1 + 4k_2 + k_3) \\
&= y_i + \frac{h}{6}\Big(6f + h(a+2+1-a)f_y f + h^2(a^2+4a-4a^2+a^2)f_y^2 f + h^2 f^* f_{yy} f\Big) \\
&\quad + O(h^3)\,.
\end{aligned}
$$

Da $4a-2a^2=1$ folgt schließlich

$$
\hat{y}_{i+1} = y_i + hf + \tfrac{1}{2}h^2 f_y f + \tfrac{1}{6}h^3(f_y^2 f + f^* f_{yy} f) + O(h^3)\,,
$$

und dies ist in der Tat die notwendige Potenzreihenentwicklung (5.6) für ein Verfahren dritter Ordnung (bei autonomer rechter Seite f).

Rosenbrock-Verfahren sollten neben einem Fehlerschätzer, der über die nächste Schrittweite entscheidet auch noch über einen Test verfügen, wann die Jacobi-Matrix J neu berechnet werden soll. In der Regel ist dies nicht nach jedem Zeitschritt erforderlich.

Nehmen wir an, das Rosenbrock-Verfahren hat die Ordnung q und das Kontrollverfahren für den Fehlerschätzer hat die Ordnung $q+1$, unabhängig von der Wahl von J – wie bei `ode23s`. Die Kontrolle über die Jacobi-Matrix könnte nun anhand eines zweiten eingebetteten Verfahrens mit Näherungen $\breve{y}_i$ erfolgen, das nur mit exakter Jacobi-Matrix die Ordnung $q+1$ hat. Häufig ist es dann so, daß ein solches Verfahren zumindest noch die Ordnung q hat, wenn näherungsweise $J = f_y(t_i, y_i) + O(h)$ gilt, also etwa, wenn J die exakte Jacobi-Matrix des vorigen Zeitschritts ist. Ist auch diese Eigenschaft verletzt, ist die Ordnung in der Regel kleiner als q.

Bei exakter Jacobi-Matrix erwartet man daher $\|\breve{y}_i - \hat{y}_i\| \ll \|y_i - \hat{y}_i\|$. Bei darauffolgenden Zeitschritten gilt wenigstens noch $J = f_y + O(h)$ und es ergibt sich $\|\breve{y}_i - \hat{y}_i\| \approx \|y_i - \hat{y}_i\|$ bis irgendwann $\|\breve{y}_i - \hat{y}_i\| \gg \|y_i - \hat{y}_i\|$. Daher liegt es nahe, die Jacobi-Matrix in dem Moment neu zu berechnen, in dem $\|\breve{y}_i - \hat{y}_i\| > \|y_i - \hat{y}_i\|$ wird.

II. Fouriertransformation

11 Innenprodukträume und Orthogonalbasen

In den nächsten beiden Kapiteln dieser Vorlesung interessieren wir uns für Funktionenräume (Vektorräume) und geeignete Basen in diesen Räumen, bezüglich der wir eine gegebene Funktion f effizient darstellen können.

Wir erinnern zunächst an den Begriff der Orthogonalität.

Definition 11.1 *Eine Abbildung $\langle \cdot, \cdot \rangle : X \times X \to \mathbb{C}$ in einem Vektorraum X über $\mathbb{C}$ heißt* ***Innenprodukt*** *oder auch* ***Skalarprodukt****, falls*

$$
\begin{array}{lll}
(i) & \langle f, g \rangle = \overline{\langle g, f \rangle} & \forall\, f, g \in X, \\
(ii) & \langle \alpha f + \beta g, h \rangle = \alpha \langle f, h \rangle + \beta \langle g, h \rangle & \forall\, f, g, h \in X, \quad \forall\, \alpha, \beta \in \mathbb{C}\,, \\
(iii) & \langle f, f \rangle > 0 & \forall\, f \in X \setminus \{0\}\,.
\end{array}
$$

Wegen (i) gilt eine entsprechende Linearitätsbedingung (ii) auch für das zweite Argument; hier treten dann hingegen die Koeffizienten komplex konjugiert auf. Man nennt eine Abbildung $\langle \cdot, \cdot \rangle$, die (i) und (ii) erfüllt, daher auch **hermitesche Bilinearform**. (iii) bezeichnet man als **Definitheit**. Wegen (ii) (mit $\beta = 0$, $h = \alpha f$) ist $\langle \alpha f, \alpha f \rangle = |\alpha|^2 \langle f, f \rangle$; speziell für $\alpha = 0$ ergibt sich $\langle 0, 0 \rangle = 0$.

Proposition 11.2 *Es gilt die* ***Cauchy-Schwarzsche Ungleichung***

$$
|\langle f, g \rangle|^2 \leq \langle f, f \rangle \cdot \langle g, g \rangle\,.
$$

Beweis. Übung. □

Definition und Satz 11.3 *Ein Innenprodukt $\langle \cdot, \cdot \rangle : X \times X \to \mathbb{R}$ induziert eine Norm in X, die wegen (iii) in Definition 11.1 wohldefiniert ist:*

$$
\|f\| := \langle f, f \rangle^{1/2}.
$$

Beweis. Übung. □

Beispiele.

- $\mathbb{C}^{m\times n}$ $\quad (m,n\in\mathbb{N})$:

  Im Raum X der komplexwertigen $m\times n$-Matrizen ist das bekannteste Skalarprodukt mit der Frobeniusnorm assoziiert:

$$\langle\langle A,B\rangle\rangle := \operatorname{Spur}(B^*A) = \sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}\overline{b_{ij}}\,.$$

- $\mathcal{L}^2(I)$ $\quad (I$ ein reelles Intervall$)$:

  Sei X der Raum aller komplexwertigen Funktionen $f: I\to\mathbb{C}$, deren Betrag quadratisch integrierbar ist. In X ist dann das folgende Skalarprodukt mit der entsprechenden Norm wohldefiniert:

$$\langle f,g\rangle_{\mathcal{L}^2(I)} := \int_I f(t)\overline{g(t)}\,dt\,, \qquad \|f\|_{\mathcal{L}^2(I)} = \Big(\int_I |f(t)|^2\,dt\Big)^{1/2}.$$

  Der Raum X, versehen mit diesem Innenprodukt, heißt $\mathcal{L}^2(I)$.

- $H^1(I)$ $\quad (I=[a,b]\subset\mathbb{R})$:

  X bezeichne den Raum aller komplexwertigen Funktionen $F\in\mathcal{L}^2(I)$ mit der folgenden Eigenschaft:

$$\exists\, f\in\mathcal{L}^2(I)\,:\quad F(x)=F(a)+\int_a^x f(t)\,dt\,.$$

  Ist $f\in\mathcal{L}^2(I)$ in dieser Weise mit $F\in X$ assoziiert und $g\in\mathcal{L}^2(I)$ in entsprechender Weise mit $G\in X$, dann ist

$$\begin{aligned}\langle F,G\rangle_{H^1(I)} &:= \int_a^b F(t)\overline{G(t)}\,dt + \int_a^b f(t)\overline{g(t)}\,dt\,,\\ \|F\|_{H^1(I)} &= \Big(\int_a^b |F(t)|^2\,dt + \int_a^b |f(t)|^2\,dt\Big)^{1/2},\end{aligned}$$

  ein Innenprodukt in X mit zugehöriger Norm. Dieser Raum ist der $H^1([a,b])$. f wird **schwache Ableitung** von F genannt und mit F' bezeichnet. Für differenzierbare Funktionen stimmt die schwache Ableitung nach dem Hauptsatz der Differential- und Integralrechnung mit der klassischen Ableitung überein.

  $H^1([a,b])$ enthält insbesondere alle linearen Splines über $[a,b]$. Zum Beweis dieser Behauptung reicht es, für eine Hutfunktion $F=B$ mit

$$B(x) = \begin{cases} (x-x_0)/(x_1-x_0) & x_0 \le x < x_1\,, \\ (x-x_2)/(x_1-x_2) & x_1 \le x < x_2\,, \\ 0 & \text{sonst} \end{cases}$$

$(x_0 < x_1 < x_2)$ die schwache Ableitung anzugeben: sie lautet

$$f(x) = \begin{cases} 1/(x_1-x_0) & x_0 < x < x_1\,, \\ -1/(x_2-x_1) & x_1 < x < x_2\,, \\ 0 & x < x_0 \quad \text{oder} \quad x > x_2\,. \end{cases}$$

Die Werte $f(x_i)$, $i = 0, 1, 2$, können beliebig festgelegt werden.

- Beachte: $\langle f, g\rangle := \int_a^b f'(t)\overline{g'(t)}\,dt$ ist *kein* Innenprodukt in $C^1([a,b])$. Dagegen ist es ein Innenprodukt auf den Teilräumen

$$\begin{aligned} X_1 &= \{f \in C^1([a,b]) : f(a) = f(b) = 0\} \subset C^1([a,b])\,, \\ X_2 &= \{f \in C^1([a,b]) : \int_a^b f(x)\,dx = 0\} \subset C^1([a,b])\,. \end{aligned}$$

- Seien x_i paarweise verschiedene reelle und ω_i beliebige positive Zahlen, $i = 0, \ldots, n$. Durch

$$\langle\, f, g\,\rangle := \sum_{i=0}^{n} \omega_i f(x_i)\overline{g(x_i)}$$

wird ein **diskretes Innenprodukt** im Raum Π_n aller (komplexen) Polynome vom Grad kleiner oder gleich n definiert.

Definition 11.4 *Eine Basis $\{\phi_i\}_{i=1}^n$ des endlichdimensionalen Teilraums $X_n \subset X$ heißt* ***Orthogonalbasis****, falls*

$$\langle \phi_i, \phi_j\rangle = 0\,, \qquad i \neq j\,.$$

Ist zudem $\|\phi_i\|^2 = \langle \phi_i, \phi_i\rangle = 1$ $(i = 1, \ldots, n)$, dann heißt $\{\phi_i\}$ ***Orthonormalbasis****.*

Beispiel 11.5 Die Legendre-Polynome

$$\tilde{P}_k(x) = \frac{\sqrt{2k+1}}{k!}\,\frac{d^k}{dx^k}\left\{x^k(1-x)^k\right\}, \qquad 0 \le k \le n\,,$$

bilden eine Orthonormalbasis bezüglich $\mathcal{L}^2(0,1)$ im Raum Π_n aller Polynome vom Grad kleiner oder gleich n.

Hat man erst einmal eine Orthonormalbasis eines endlichdimensionalen Teilraums $X_n \subset X$ zur Verfügung, dann gelten folgende Resultate.

Satz 11.6 *Sei $\{\phi_i\}_{i=1}^n$ eine Orthonormalbasis von $X_n \subset X$. Dann gilt*

(a) $f \in X_n \;\Rightarrow\; f = \sum_{i=1}^{n} \langle f, \phi_i \rangle \phi_i \,,$

(b) $f \in X_n \;\Rightarrow\; \|f\|^2 = \sum_{i=1}^{n} |\langle f, \phi_i \rangle|^2$ ***(Pythagoras)***,

(c) $f \notin X_n \;\Rightarrow\; f_n = \sum_{i=1}^{n} \langle f, \phi_i \rangle \phi_i$ *ist die Bestapproximation an* f *aus* X_n, *d. h.*

$$\|f - f_n\| \;<\; \|f - g\| \qquad \textit{für alle } g \in X_n \setminus \{f_n\} \,,$$

(d) $\sum_{i=1}^{n} |\langle f, \phi_i \rangle|^2 \;\leq\; \|f\|^2$ *für alle* $f \in X$ ***(Besselsche Ungleichung)***.

Beweis. (a) Nach Voraussetzung gilt $f = \sum_{i=1}^{n} \alpha_i \phi_i$ für gewisse $\alpha_i \in \mathbb{C}$. Also folgt

$$\langle f, \phi_j \rangle \;=\; \langle \sum_{i=1}^{n} \alpha_i \phi_i, \phi_j \rangle \;=\; \sum_{i=1}^{n} \alpha_i \langle \phi_i, \phi_j \rangle \;=\; \alpha_j \cdot 1 \,.$$

(b) Aus (a) folgt

$$\begin{aligned} \|f\|^2 \;&=\; \langle f, f \rangle \;=\; \langle \sum_{i=1}^{n} \alpha_i \phi_i, \sum_{j=1}^{n} \alpha_j \phi_j \rangle \;=\; \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \overline{\alpha_j} \langle \phi_i, \phi_j \rangle \\ &=\; \sum_{i=1}^{n} |\alpha_i|^2 \;=\; \sum_{i=1}^{n} |\langle f, \phi_i \rangle|^2 \,. \end{aligned}$$

(c)

$$\begin{aligned} \|f - \sum_{i=1}^{n} \tilde{\alpha}_i \phi_i\|^2 \;&=\; \langle f - \sum_{i=1}^{n} \tilde{\alpha}_i \phi_i, f - \sum_{j=1}^{n} \tilde{\alpha}_j \phi_j \rangle \\ &=\; \|f\|^2 - \sum_{i=1}^{n} 2 \operatorname{Re} \big(\tilde{\alpha}_i \underbrace{\langle \phi_i, f \rangle}_{=:\overline{\alpha_i}} \big) + \sum_{i=1}^{n} \sum_{j=1}^{n} \tilde{\alpha}_i \overline{\tilde{\alpha}_j} \langle \phi_i, \phi_j \rangle \\ &=\; \|f\|^2 - 2 \sum_{i=1}^{n} \operatorname{Re} \big(\tilde{\alpha}_i \overline{\alpha_i} \big) + \sum_{i=1}^{n} |\tilde{\alpha}_i|^2 \,. \end{aligned}$$

Durch quadratische Ergänzung ergibt sich

$$\|f - \sum_{i=1}^{n} \tilde{\alpha}_i \phi_i\|^2 \;=\; \|f\|^2 - \sum_{i=1}^{n} |\alpha_i|^2 + \sum_{i=1}^{n} |\alpha_i - \tilde{\alpha}_i|^2 \,, \tag{11.1}$$

und hieraus folgt sofort die Behauptung.

(d) Für $f \in X_n$ klar wegen (b). Für $f \in X \setminus X_n$ gilt nach (11.1) mit $\alpha_i = \langle f, \phi_i \rangle$

$$0 < \|f - \sum_{i=1}^{n} \alpha_i \phi_i\|^2 = \|f\|^2 - \sum_{i=1}^{n} |\alpha_i|^2 = \|f\|^2 - \sum_{i=1}^{n} |\langle f, \phi_i \rangle|^2 .$$

□

Dieser Satz macht klar, warum Orthogonalität in der Numerik so wichtig ist: Man hat einerseits einfache Basisdarstellungen und kann andererseits unmittelbar die Bestapproximation an eine vorgegebene Funktion berechnen.

12 Trigonometrische Polynome

Definition 12.1 *Sei* $\mathcal{T}_n := span\{e^{ik\theta} : -n \le k \le n\}$. *Ein Element* $t \in \mathcal{T}_n$,

$$t(\theta) = \sum_{k=-n}^{n} \alpha_k e^{ik\theta} , \qquad \alpha_k \in \mathbb{C} , \tag{12.1}$$

heißt ***trigonometrisches Polynom*** *vom Grad* n.

Wegen ihrer Periodizität verwendet man trigonometrische Polynome vor allem zur Approximation **periodischer** Funktionen. Periodische Funktionen treten in Anwendungen beispielsweise im Zusammenhang mit Funktionen über geschlossenen Kurven auf.

Bemerkung. Falls $\alpha_k = \overline{\alpha_{-k}}$ für alle $k = 0, \ldots, n$, dann hat t über $\mathbb{R}$ nur reelle Werte und kann zudem rein reell dargestellt werden,

$$t(\theta) = \frac{a_0}{2} + \sum_{k=1}^{n} \big(a_k \cos k\theta + b_k \sin k\theta \big) ,$$

mit $a_0 = 2\alpha_0$ und

$$a_k = 2 \operatorname{Re} \alpha_k , \quad b_k = -2 \operatorname{Im} \alpha_k , \qquad k = 1, \ldots, n .$$

Man spricht in diesem Fall von einem **reellen trigonometrischen Polynom** vom Grad n.

Wir wollen nun eine gegebene Funktion f über $[0, 2\pi]$ durch trigonometrische Polynome approximieren. Aufgrund der Vorüberlegungen aus Paragraph 11 bietet sich dabei die $\mathcal{L}^2$-Norm als "Gütemaß" an:

Satz 12.2 *Die Funktionen*

$$\frac{1}{\sqrt{2\pi}}\, e^{ik\theta}\,, \qquad k = -n, \ldots, n\,, \tag{12.2}$$

bilden eine Orthonormalbasis bezüglich $\mathcal{L}^2(0,2\pi)$ *von* $\mathcal{T}_n$*. Die* $\mathcal{L}^2(0,2\pi)$*-Bestapproximation an* f *aus* $\mathcal{T}_n$ *hat daher die Form* (12.1) *mit*

$$\alpha_k = \frac{1}{2\pi}\int_0^{2\pi} f(\theta) e^{-ik\theta}\, d\theta\,, \qquad k = -n, \ldots, n\,. \tag{12.3}$$

Beweis. Für $-n \le j, k \le n$ gilt

$$\begin{aligned}
\langle \frac{1}{\sqrt{2\pi}} e^{ik\theta}, \frac{1}{\sqrt{2\pi}} e^{ij\theta} \rangle &= \frac{1}{2\pi}\int_0^{2\pi} e^{ik\theta} e^{-ij\theta}\, d\theta \\
&= \begin{cases} \dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} d\theta \;=\; 1\,, & k = j\,, \\[2ex] \dfrac{1}{2\pi}\,\dfrac{1}{i(k-j)}\, e^{i(k-j)\theta}\Big|_0^{2\pi} \;=\; 0\,, & k \neq j\,. \end{cases}
\end{aligned}$$

Die Form der Bestapproximation ergibt sich daher durch Anwendung von Satz 11.6 (c). □

Man beachte, daß die Koeffizenten α_k *nicht* von n abhängen. Es bietet sich daher an, den Grenzübergang $n \to \infty$ zu betrachten und zu fragen, in welchem Sinn

$$f(\theta) \sim \sum_{k=0}^{\infty} \alpha_k e^{ik\theta} \tag{12.4}$$

Gültigkeit hat. Die rechte Seite von (12.4) bezeichnet man als **formale Fourierreihe** von f; die Untersuchung ihrer Konvergenzeigenschaften ist Gegenstand der **Fourieranalyse**. Hier wollen wir zunächst nur drei wichtige Resultate zitieren:

- Ist $f \in \mathcal{L}^2(0,2\pi)$ in einer Umgebung von $x \in (0,2\pi)$ von beschränkter Variation, dann konvergiert die Fourierreihe gegen $\frac{1}{2}\big(f(x+) + f(x-)\big)$ an der Stelle x; für $x = 0$ und $x = 2\pi$ gilt ein entsprechendes Resultat mit Grenzwert $\frac{1}{2}\big(f(0+) + f(2\pi-)\big)$ (Heuser II, Satz 136.1).

- Ist f stetig und 2π-periodisch, sowie $t_n \in \mathcal{T}_n$ die n-te Partialsumme der Fourierreihe von f, dann gilt

$$\lim_{N\to\infty} \frac{1}{N}\sum_{n=0}^{N} t_n \;=\; f\,,$$

  gleichmäßig in $[0,2\pi]$ (Satz von Fejér, vgl. Heuser II, Satz 139.5).

- Ist f zudem stückweise stetig differenzierbar, dann konvergiert auch die Fourierreihe selber gleichmäßig gegen f (Heuser II, Satz 137.2).

Wie beweisen folgenden Zusammenhang zwischen der $\mathcal{L}^2$-Norm von f und seinen Fourierkoeffizenten:

Proposition 12.3 *Sei $f \in \mathcal{L}^2(0,2\pi)$ mit der formalen Fourierreihe* (12.4).

(a) Dann gilt immer

$$\sum_{k=-\infty}^{\infty} |\alpha_k|^2 \;\leq\; \frac{1}{2\pi} \|f\|^2_{\mathcal{L}^2(0,2\pi)} \,. \tag{12.5}$$

(b) Konvergiert zudem die Fourierreihe von f gleichmäßig in $[0,2\pi]$, dann gilt Gleichheit in (12.5).

Beweis. (a) Die Entwicklungskoeffizenten von f bezüglich der Orthonormalbasis (12.2) lauten $\sqrt{2\pi}\,\alpha_k$ für $-n \leq k \leq n$. Nach der Besselschen Ungleichung, Satz 11.6 (d), gilt daher

$$2\pi \sum_{k=-n}^{n} |\alpha_k|^2 \;\leq\; \|f\|^2_{\mathcal{L}^2(0,2\pi)} \,.$$

Da die Fourierkoeffizenten nicht von n abhängen, folgt (12.5).

(b) Im Falle gleichmäßiger Konvergenz der Fourierreihe gilt

$$\int_0^{2\pi} |f(x)|^2 \, dx \;=\; \lim_{n\to\infty} \int_0^{2\pi} \Big| \sum_{k=-n}^{n} \alpha_k e^{ik\theta} \Big|^2 \, dx \,.$$

Wegen der Orthogonalität der Basisfunktionen von $\mathcal{T}_n$ ergibt das Integral auf der rechten Seite $\sum_{k=-n}^{n} 2\pi\,|\alpha_k|^2$, und wegen (12.5) konvergiert dies für $n \to \infty$ gegen die unendliche Reihe, also gilt

$$\|f\|^2_{\mathcal{L}^2(0,2\pi)} \;=\; 2\pi \sum_{k=-\infty}^{\infty} |\alpha_k|^2 \,.$$

□

Bemerkung 12.4 Es sei an dieser Stelle festgehalten, daß für *jede* Funktion $f \in \mathcal{L}^2(0,2\pi)$ Gleichheit in (12.5) gilt, und $\mathcal{L}^2$-Funktionen gerade dadurch charakterisiert sind, daß die Reihe auf der linken Seite von (12.5) konvergiert. Allerdings können wir das mit unseren derzeitigen Hilfsmitteln nicht beweisen (und zudem ist das ohnehin eine Aufgabe der Vorlesung Funktionalanalysis). Wir werden dennoch im weiteren $\mathcal{L}^2$-Funktionen mit ihren formalen Potenzreihen identifizieren und umgekehrt.

Beispiel 12.5 Als Beispiel betrachten wir die charakteristische Funktion $\chi_{[a,b]}$ eines Intervalls $(a,b) \subset (0,2\pi)$. Wir setzen $c = (a+b)/2$ und $d = (b-a)/2 < \pi$. Gemäß (12.3) gilt dann für die Fourierkoeffizienten die Formel

$$\alpha_k \;=\; \frac{1}{2\pi}\int_a^b e^{-ik\theta}\,d\theta\,, \qquad k\in\mathbb{Z}\,.$$

Also ist $\alpha_0 = (b-a)/(2\pi) = d/\pi$ und für $k\neq 0$ ergibt sich

$$\begin{aligned}
\alpha_k &= \frac{1}{2\pi}\,\frac{1}{-ik}\,e^{-ik\theta}\Big|_a^b \;=\; -\frac{1}{2k\pi i}\,e^{-ikc}\left(e^{-ikd}-e^{ikd}\right)\\
&= -\frac{1}{2k\pi i}\,e^{-ikc}\left(\cos kd - i\sin kd - \cos kd - i\sin kd\right)\\
&= \frac{1}{\pi}\,e^{-ikc}\,\frac{\sin kd}{k}\,.
\end{aligned}$$

Somit hat die formale Fourierreihe von $\chi_{[a,b]}$ die Gestalt

$$\begin{aligned}
\chi_{[a,b]}(\theta) &\sim \frac{d}{\pi} + \frac{1}{\pi}\sum_{|k|=1}^{\infty} e^{-ikc}\,\frac{\sin kd}{k}\,e^{ik\theta}\\
&= \frac{d}{\pi} + \frac{1}{\pi}\sum_{k=1}^{\infty}\frac{\sin kd}{k}\left(e^{ik(\theta-c)}+e^{-ik(\theta-c)}\right)\\
&= \frac{d}{\pi} + \frac{2}{\pi}\sum_{k=1}^{\infty}\frac{1}{k}\,\sin kd\,\cos k(\theta-c)\,.
\end{aligned}$$

Bezeichnen wir mit t_n die Bestapproximation aus $\mathcal{T}_n$ an $\chi_{[a,b]}$, dann gilt für den Fehler $\chi_{[a,b]} - t_n$ gemäß Proposition 12.3

$$\|\chi_{[a,b]} - t_n\|^2_{\mathcal{L}^2(0,2\pi)} \;\geq\; 2\pi \sum_{|k|=n+1}^{\infty} \alpha_k^2 \;=\; \frac{2}{\pi}\sum_{|k|=n+1}^{\infty}\frac{\sin^2 kd}{k^2}\,.$$

Da $d<\pi$ ist, verhält sich die Summe auf der rechten Seite wie

$$\sum_{|k|=n+1}^{\infty} k^{-2} \;\sim\; \int_n^{\infty} t^{-2}\,dt \;\sim\; n^{-1}\,,$$

also ist

$$\|\chi_{[a,b]} - t_n\|_{\mathcal{L}^2(0,2\pi)} \;\gtrsim\; n^{-1/2}\,. \tag{12.6}$$

Im Hinblick auf Bemerkung 12.4 kann dabei $\gtrsim$ in (12.6) auch durch $\sim$ ersetzt werden.

Abbildung 12.1 erläutert exemplarisch das Konvergenzverhalten dieser Fourierreihe. Dargestellt sind die drei Partialsummen für $|k|\leq n$ mit $n = 8, 16$ und 128.

13 Sobolevräume

In Abschnitt 11 haben wir den Raum $H^1([0,2\pi])$ eingeführt. Wir wollen uns im folgenden auf die Teilmenge $H^1_\pi([0,2\pi])$ der periodischen Funktionen in $H^1([0,2\pi])$ beschränken.

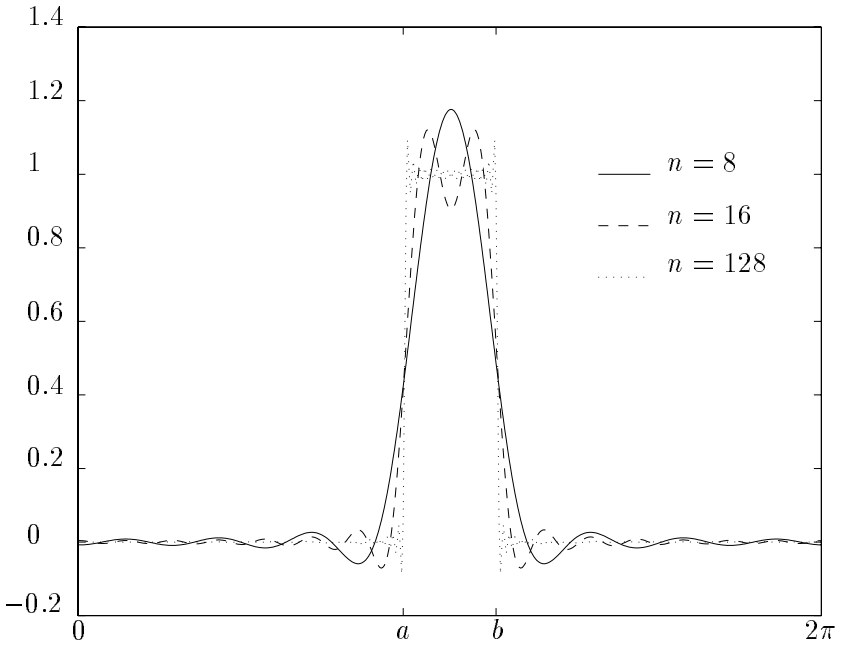


Fig. 12.1. Konvergenz der Fourierreihe der charakteristischen Funktion

Zu einer Funktion $f \in H^1_\pi([0,2\pi])$ gibt es nach Voraussetzung eine Funktion

$$f'(\theta) \sim \sum_{k=-\infty}^{\infty} \beta_k e^{ik\theta} \qquad \text{mit} \qquad f(\theta) = f(0) + \int_0^\theta f'(t)\,dt\,. \tag{13.1}$$

Wegen der 2π-Periodizität von f kommen dabei nicht alle möglichen $\mathcal{L}^2(0,2\pi)$-Funktionen f' in Frage, denn es muß

$$f(0) = f(2\pi) = f(0) + \int_0^{2\pi} f'(t)\,dt$$

gelten; folglich ist

$$\beta_0 = \frac{1}{2\pi}\int_0^{2\pi} f'(\theta)\,d\theta = 0\,. \tag{13.2}$$

Ausgehend von (13.1) bietet es sich an, die Fourierreihe von f durch gliedweise Integration der Fourierreihe von f' zu bestimmen. Zunächst ist diese Vorgehensweise jedoch in keinster Weise gerechtfertigt, da die Konvergenzeigenschaften der beiden Fourierreihen nicht geklärt sind. Dennoch führt dieser Ansatz zum Ziel, wie das folgende Resultat bestätigt.

Lemma 13.1 *Sei $f \in H^1_\pi([0,2\pi])$ wie oben definiert. Dann sind für $k \neq 0$ die Fourierkoeffizienten von f durch $\alpha_k = \beta_k/(ik)$ gegeben.*

Beweis. Nach Satz 12.2 berechnen sich die Fourierkoeffizienten von f nach der Formel

$$\alpha_k = \frac{1}{2\pi}\int_0^{2\pi} f(\theta)e^{-ik\theta}\,d\theta\,.$$

Durch Einsetzen von (13.1) ergibt sich (für $k \neq 0$)

$$\begin{aligned}
\alpha_k &= \frac{1}{2\pi}\int_0^{2\pi}\Big(f(0)+\int_0^{\theta} f'(t)\,dt\Big)e^{-ik\theta}\,d\theta \\
&= \frac{f(0)}{2\pi}\int_0^{2\pi} e^{-ik\theta}\,d\theta \;+\; \frac{1}{2\pi}\int_0^{2\pi} f'(t)\int_t^{2\pi} e^{-ik\theta}\,d\theta\,dt \\
&= \frac{1}{2\pi}\int_0^{2\pi} f'(t)\,\frac{1}{-ik}e^{-ik\theta}\Big|_t^{2\pi}\,dt \\
&= -\frac{1}{ik}\,\frac{1}{2\pi}\Big(\int_0^{2\pi} f'(t)\,dt \;-\; \int_0^{2\pi} f'(t)e^{-ikt}\,dt\Big) \\
&= -\frac{1}{ik}\,(\beta_0-\beta_k) \underset{(13.2)}{=} \frac{1}{ik}\,\beta_k\,.
\end{aligned}$$

□

Als Konsequenz aus diesem Hilfsatz erhalten wir

Satz 13.2 *Eine Funktion $f \in \mathcal{L}^2(0,2\pi)$ mit $f(\theta) \sim \sum \alpha_k e^{ik\theta}$ gehört genau dann zu $H^1_\pi([0,2\pi])$, falls*

$$\sum_{k=-\infty}^{\infty} k^2|\alpha_k|^2 \;<\; \infty\,.$$

In diesem Fall ist

$$\sum_{k=-\infty}^{\infty}(k^2+1)|\alpha_k|^2 \;=\; \frac{1}{2\pi}\,\|f\|^2_{H^1_\pi([0,2\pi])}\,.$$

Beweis. Die eine Beweisrichtung folgt unmittelbar aus dem vorangegangenen Lemma 13.1: Für eine Funktion $f \in H^1_\pi([0,2\pi])$ sind $ik\alpha_k$ die Fourierkoeffizienten der schwachen Ableitung $f' \in \mathcal{L}^2(0,2\pi)$; also konvergiert nach Proposition 12.3 die Reihe $\sum_{k=-\infty}^{\infty} k^2|\alpha_k|^2$ und wegen Bemerkung 12.4 ist

$$\sum_{k=-\infty}^{\infty}(k^2+1)|\alpha_k|^2 \;=\; \sum_{k=-\infty}^{\infty} k^2|\alpha_k|^2 + \sum_{k=-\infty}^{\infty}|\alpha_k|^2 \;=\; \frac{1}{2\pi}\Big(\|f'\|^2_{\mathcal{L}^2(0,2\pi)} + \|f\|^2_{\mathcal{L}^2(0,2\pi)}\Big)\,.$$

Konvergiert umgekehrt die Reihe $\sum_{k=-\infty}^{\infty} k^2|\alpha_k|^2$, dann definiert nach Bemerkung 12.4 die formale Fourierreihe $\sum_{k=-\infty}^{\infty} ik\alpha_k e^{ik\theta}$ eine Funktion $f' \in \mathcal{L}^2(0,2\pi)$, die nach Lemma 13.1 mit der schwachen Ableitung von f übereinstimmt, da beide die gleiche Fourierreihe haben. □

Beispiel 13.3 In Beispiel 12.5 haben wir die formale Fourierreihe der charakteristischen Funktion eines Intervalls $(a,b) \subset [0,2\pi]$ bestimmt. Demnach ist für $0 \le x_0 < x_1 < x_2 \le 2\pi$

$$\begin{aligned} f(\theta) &:= \frac{1}{x_1 - x_0}\,\chi|_{[x_0,x_1]} - \frac{1}{x_2 - x_1}\,\chi|_{[x_1,x_2]} \\ &\sim \sum_{|k|=1}^{\infty} \Big(\frac{1}{2d_1\pi}\, e^{-ikc_1}\, \frac{\sin kd_1}{k} - \frac{1}{2d_2\pi}\, e^{-ikc_2}\, \frac{\sin kd_2}{k} \Big)\, e^{ik\theta}\,, \end{aligned}$$

wobei $c_1 = (x_0 + x_1)/2$, $c_2 = (x_1 + x_2)/2$, $d_1 = (x_1 - x_0)/2$ und $d_2 = (x_2 - x_1)/2$ die jeweiligen Intervallmittelpunkte, bzw. -radien sind. Nach Abschnitt 11 ist f gerade die schwache Ableitung einer Hutfunktion $B \in H^1_\pi([0,2\pi])$, gegeben durch

$$B(\theta) = \begin{cases} (\theta - x_0)/(x_1 - x_0) & x_0 \le \theta < x_1\,, \\ (\theta - x_2)/(x_1 - x_2) & x_1 \le \theta < x_2\,, \\ 0 & \text{sonst}\,. \end{cases}$$

Folglich hat B nach Satz 13.2 die Fourierreihe

$$B(\theta) \sim \alpha_0 + \sum_{|k|=1}^{\infty} \Big(\frac{1}{2d_1\pi}\, e^{-ikc_1} \sin kd_1 - \frac{1}{2d_2\pi}\, e^{-ikc_2} \sin kd_2 \Big) \frac{1}{ik^2}\, e^{ik\theta}\,. \tag{13.3}$$

Dabei ist $\alpha_0 = \frac{1}{2\pi}\int_0^{2\pi} B(\theta)\,d\theta$ ebenfalls schnell ausgerechnet: $\alpha_0 = (d_1 + d_2)/(2\pi)$.

Ausgehend von Satz 13.2 können wir nun eine ganze Reihe weiterer Funktionenräume einführen.

Definition und Satz 13.4 *Sei $s > 0$: Der **Sobolevraum** $H^s_\pi([0,2\pi])$ ist definiert als*

$$H^s_\pi([0,2\pi]) := \{\, f \in \mathcal{L}^2(0,2\pi),\, f(\theta) \sim \sum_{k=-\infty}^{\infty} \alpha_k e^{ik\theta} \;:\; \sum_{k=-\infty}^{\infty} |k|^{2s}|\alpha_k|^2 < \infty \,\}\,.$$

Haben $f, g \in H^s_\pi([0,2\pi])$ die formalen Fourierreihen $\sum \alpha_k e^{ik\theta}$ und $\sum \beta_k e^{ik\theta}$, dann definiert

$$\langle f, g \rangle_{H^s_\pi([0,2\pi])} := 2\pi \sum_{k=-\infty}^{\infty} (|k|^{2s} + 1)\, \alpha_k \overline{\beta_k}$$

ein Innenprodukt in $H^s_\pi([0,2\pi])$.

Beweis. Nachrechnen. □

Eigenschaften:

- Für $r > s > 0$ gilt $H^r_\pi([0,2\pi]) \subset H^s_\pi([0,2\pi])$. Dies folgt unmittelbar aus der Definition 13.4 mit dem Majorantenkriterium.
- Für $s > 1$ ist insbesondere $H^s_\pi([0,2\pi]) \subset H^1_\pi([0,2\pi])$, und daher hat für $s > 1$ jede Funktion $f \in H^s_\pi([0,2\pi])$ eine schwache Ableitung f'. Aus Lemma 13.1 folgt zudem unmittelbar, daß $f' \in H^{s-1}_\pi([0,2\pi])$ liegt. Insbesondere hat also eine Funktion $f \in H^s_\pi([0,2\pi])$, $s \in \mathbb{N}$, s schwache Ableitungen f', f'', f''', ..., $f^{(s)}$, mit $f^{(s)} \in \mathcal{L}^2(0,2\pi)$.

Sobolevräume sind aus der modernen angewandten und numerischen Mathematik kaum mehr wegzudenken und haben inzwischen die ehemals dominierende Rolle der Funktionenräume C^s (mit $s \in \mathbb{N}$) übernommen. In beiden Fällen charakterisiert der Index s eine gewisse "Glattheit" entsprechender Funktionen $f \in H^s_\pi$ bzw. $f \in C^s$, die allerdings für die beiden Funktionenräume nicht übereinstimmt. So haben wir bereits festgehalten, daß die 2π-periodischen Funktionen $f \in C^1(\mathbb{R})$ immer auch zu $H^1_\pi([0,2\pi])$ gehören, und entsprechend gehören die 2π-periodischen Funktionen $f \in C^s(\mathbb{R})$ mit $s \in \mathbb{N}$ immer auch zu $H^s_\pi([0,2\pi])$. Die Umkehrung ist allerdings falsch, wie das Beispiel der Hutfunktion belegt: Die Hutfunktion B aus Beispiel 13.3 ist nicht stetig differenzierbar, gehört aber zu $H^1_\pi([0,2\pi])$; sie gehört sogar zu $H^s_\pi([0,2\pi])$ für jedes $s < 3/2$, denn mit den Fourierkoeffizienten aus (13.3) gilt

$$\sum_{k=-\infty}^{\infty} |k|^{2s} |\alpha_k|^2 \;\le\; \frac{d_1 + d_2}{d_1 d_2 \pi} \sum_{k=1}^{\infty} |k|^{2s-4} \;<\; \infty$$

für $s < 3/2$.

Um ein Gefühl für die "Glattheit" einer Funktion $f \in H^s_\pi([0,2\pi])$ zu bekommen, beweisen wir abschließend noch die folgenden beiden Aussagen.

Satz 13.5 *Für $s > 1/2$ konvergiert die Fourierreihe einer Funktion $f \in H^s_\pi([0,2\pi])$ gleichmäßig in $[0,2\pi]$, d.h., alle Funktionen $f \in H^s_\pi([0,2\pi])$ sind stetig und 2π-periodisch. Speziell für $s = 1$ sind die Funktionen $f \in H^1_\pi([0,2\pi])$ sogar Hölder-stetig mit Hölder-Exponenten $\alpha = 1/2$; genauer gilt*

$$|f(\theta) - f(t)| \;\le\; \|f\|_{H^1_\pi([0,2\pi])} \, |\theta - t|^{1/2}\,, \qquad \theta, t \in [0,2\pi]\,, \quad f \in H^1_\pi([0,2\pi])\,.$$

Beweis. Die gleichmäßige Konvergenz der Fourierreihe für eine Funktion $f \in H^s_\pi([0,2\pi])$ mit $s > 1/2$ folgt unmittelbar aus der Cauchy-Schwarzschen Ungleichung: Es gilt nämlich für $m > n$

$$\begin{aligned}
\sum_{|k|=n}^{m} |\alpha_k e^{ik\theta}| \;&=\; \sum_{|k|=n}^{m} |k|^{-s} \big(|k|^s |\alpha_k|\big) \;\le\; \Big(\sum_{|k|=n}^{m} |k|^{-2s}\Big)^{1/2} \Big(\sum_{|k|=n}^{m} |k|^{2s} |\alpha_k|^2\Big)^{1/2} \\
&\le\; \frac{1}{\sqrt{\pi}} \, \|f\|_{H^s_\pi([0,2\pi])} \Big(\sum_{k=n}^{\infty} k^{-2s}\Big)^{1/2}\,,
\end{aligned}$$

und letzteres ist eine konvergente Majorante, gleichmäßig für alle $\theta \in [0, 2\pi]$. Daraus folgt unmittelbar die erste Behauptung.

Für $s = 1$ ergibt sich (wiederum mit der Cauchy-Schwarzschen Ungleichung, diesmal aber in $\mathcal{L}^2$) aus der ursprünglichen Definition einer Funktion $f \in H^1_\pi([0, 2\pi])$, daß

$$\begin{aligned} |f(\theta) - f(t)| &= \left| \int_t^\theta f'(\tau)\, d\tau \right| \leq \left(\int_t^\theta d\tau \right)^{1/2} \left(\int_t^\theta |f'(\tau)|^2\, d\tau \right)^{1/2} \\ &\leq |\theta - t|^{1/2}\, \|f\|_{H^1_\pi([0,2\pi])}\,. \end{aligned}$$

□

Beispiele. Die Schärfe dieses Satzes macht man sich unmittelbar an den folgenden beiden Beispielen klar:

- Die charakteristische Funktion χ eines Intervalls $(a, b) \subset [0, 2\pi]$ ist nicht stetig, gehört aber nach Beispiel 12.5 und Definition 13.4 zu allen $H^s_\pi([0, 2\pi])$-Räumen mit $s < 1/2$.
- Die $\mathcal{L}^2$-Funktionen $f_\alpha(\theta) = \big(\theta(2\pi - \theta)\big)^\alpha$ mit $0 < \alpha < 1$ haben in $(0, 2\pi)$ die klassische Ableitung

  $$f'_\alpha(\theta) = 2\alpha\big(\theta(2\pi - \theta)\big)^{\alpha-1}(\pi - \theta)\,;$$

  Daher gilt

  $$\int_0^1 |f'_\alpha(\theta)|^2\, d\theta = 4\alpha^2 \int_0^1 \theta^{2\alpha-2}(2\pi - \theta)^{2\alpha-2}(\pi - \theta)^2\, d\theta\,,$$

  und dieses uneigentliche Integral existiert lediglich für $\alpha > 1/2$. Mit anderen Worten: f_α gehört (nur) für $\alpha > 1/2$ zu $H^1_\pi([0, 2\pi])$. Man beachte, daß die Funktion f_α wegen ihres Verhaltens an der Stelle $\theta = 0$ lediglich Hölder-stetig mit Hölder-Exponenten α ist. Für eine $H^1_\pi([0, 2\pi])$-Funktion kann also kein größerer Hölder-Exponent als $\alpha = 1/2$ garantiert werden.

Entsprechende Resultate gelten dann natürlich auch für die schwachen Ableitungen von Funktionen aus $H^s_\pi([0, 2\pi])$ mit größeren s. Beispielsweise ist eine Funktion $f \in H^2_\pi([0, 2\pi])$ grundsätzlich stetig differenzierbar, und die (klassische) Ableitung f' ist zudem Hölder-stetig mit Exponenten $\alpha = 1/2$.

14 Trigonometrische Interpolation

Wenden wir uns nun der numerischen Approximation einer Funktion f mittels trigonometrischer Polynome zu. Am naheliegendsten ist dabei der Zugang über die Bestapproximation

bezüglich $\mathcal{L}^2$, also die Berechnung des Polynoms aus Satz 12.2. Leider können die Fourierkoeffizienten (12.3) in den seltensten Fällen analytisch berechnet werden. Zur numerischen Approximation bietet sich die **zusammengesetzte Trapezregel** an: Dazu verwenden wir die Werte $f(\theta_j)$ an $N \geq 2n$ äquidistanten Stützstellen $\theta_j = jh = 2\pi j/N$ (mit $h = 2\pi/N$ und $j = 0, \ldots, N-1$); wegen der Periodizität von f ergibt sich daraus die folgende Approximation:

$$\alpha_k \approx \hat{\alpha}_k = \frac{1}{N} \sum_{j=0}^{N-1} f(\theta_j) e^{-ik\theta_j}, \qquad k = -n, \ldots, n. \tag{14.1}$$

Für die Trapezregel gibt es eine besondere Fehlerabschätzung für *periodische* Funktionen, vgl. (14.2), die wesentlich besser ist als die übliche Abschätzung. Ihren Beweis werden wir am Ende dieses Abschnitts nachreichen.

Satz 14.1 *Für jede Funktion $g \in H^s_\pi([0, 2\pi])$ mit $s > 1/2$ gilt*

$$\Big| \int_0^{2\pi} g(\theta)\, d\theta - \frac{2\pi}{N} \sum_{j=0}^{N-1} g(\theta_j) \Big| \leq c_s \|g\|_{H^s_\pi([0,2\pi])} h^s \tag{14.2}$$

mit $h = 2\pi/N$ und einer positiven Konstanten c_s, die nur von s abhängt.

Besonders für glatte periodische Funktionen bilden die Koeffizenten $\hat{\alpha}_k$ aus (14.1) also sehr gute Näherungen für α_k. Wir studieren daher im weiteren die Näherungspolynome $t_n \in \mathcal{T}_n$ mit $n \leq N/2$ gegeben durch

$$t_n(\theta) = \sum_{k=-n}^{n} \hat{\alpha}_k e^{ik\theta}, \qquad n < N/2, \tag{14.3a}$$

$$\text{bzw.} \quad t_n(\theta) = \sum_{k=1-n}^{n} \hat{\alpha}_k e^{ik\theta}, \qquad n = N/2,\ N \text{ gerade}, \tag{14.3b}$$

Lemma 14.2 *Wir betrachten das diskrete Innenprodukt*

$$\langle\langle \phi, \psi \rangle\rangle := \sum_{\nu=0}^{N-1} \phi(\theta_\nu) \overline{\psi(\theta_\nu)}, \qquad \theta_\nu = \frac{2\pi\nu}{N}. \tag{14.4}$$

Für $j, k \in \mathbb{Z}$ ergibt sich

$$\langle\langle e^{ij\theta}, e^{ik\theta} \rangle\rangle = \begin{cases} N, & j - k = lN \ \text{ für ein } l \in \mathbb{Z}, \\ 0, & \text{sonst}; \end{cases}$$

insbesondere bilden also die Funktionen $\{e^{ik\theta}/\sqrt{N} : -N/2 < k \leq N/2\}$ ein Orthonormalsystem bezüglich des diskreten Innenprodukts (14.4).

Beweis. Seien $j,k \in \mathbb{Z}$ beliebig gewählt. Dann ist

$$\langle\langle e^{ij\theta}, e^{ik\theta}\rangle\rangle \;=\; \sum_{\nu=0}^{N-1} e^{ij\theta_\nu} e^{-ik\theta_\nu} \;=\; \sum_{\nu=0}^{N-1} e^{i(j-k)\nu 2\pi/N} \;=\; \sum_{\nu=0}^{N-1} \left(e^{i(j-k)2\pi/N}\right)^\nu$$

$$= \begin{cases} \displaystyle\sum_{\nu=0}^{N-1} \left(e^{i2l\pi}\right)^\nu \;=\; N\,, & j-k = lN \;\text{ für ein } l \in \mathbb{Z}\,, \\[2ex] \dfrac{1-e^{i(j-k)2\pi}}{1-e^{i(j-k)2\pi/N}} \;=\; 0\,, & \text{sonst}\,. \end{cases}$$

□

Es ist zu beachten, daß das in Lemma 14.2 genannte Orthonormalsystem immer genau N Basisfunktionen enthält, unabhängig davon, ob N gerade oder ungerade ist. Der ungewöhnliche Indexbereich für k in (14.3b) erklärt sich dadurch, daß die Funktionen $e^{in\theta}$ und $e^{-in\theta}$ an den diskreten Punkten θ_j übereinstimmen und daher bezüglich des Innenprodukts (14.4) nicht unterscheidbar sind. Mit anderen Worten: Für gerades N und $n = N/2$ ist (14.4) kein Innenprodukt mehr über $\mathcal{T}_n$.

Aus Lemma 14.2 erhalten wir nun sofort

Satz 14.3 *Sei $n < N/2$ und t_n wie in* (14.3a) *definiert. Dann gilt für jedes $t \in \mathcal{T}_n \setminus \{t_n\}$:*

$$\sum_{\nu=0}^{N-1} \left|t_n(\theta_\nu) - f(\theta_\nu)\right|^2 \;<\; \sum_{\nu=0}^{N-1} \left|t(\theta_\nu) - f(\theta_\nu)\right|^2 . \tag{14.5}$$

Beweis. Da $n < N/2$ ist, bilden also die Funktionen $e^{ik\theta}/\sqrt{N}$ eine Orthonormalbasis von $\mathcal{T}_n$ bezüglich (14.4). Wegen $\sum_{\nu=0}^{N-1} |t(\theta_\nu) - f(\theta_\nu)|^2 = \langle\langle t-f, t-f\rangle\rangle$ folgt die Aussage daher aus Satz 11.6 (c) und der speziellen Definition (14.1) der $\hat{\alpha}_k$ in (14.3). □

Das Polynom t_n ist also das trigonometrische Polynom vom Grad n, das die Funktionswerte $f(\theta_j)$ an den Punkten θ_j, $j = 0,\ldots,N-1$, am besten (im quadratischen Mittel) approximiert. Besonders interessant ist der Fall $N = 2n$:

Korollar 14.4 *Für $N = 2n$ interpoliert t_n aus* (14.3b) *die Funktion f in allen Knoten $\theta = \theta_j$, $j = 0,\ldots,N-1$, d. h. $\langle\langle f - t_n, f - t_n\rangle\rangle = 0$.*

Beweis. Nach Lemma 14.2 bilden die Funktionen

$$\tau_k(\theta) \;=\; \frac{1}{\sqrt{N}}\, e^{ik\theta}\,, \qquad 1-n \le k \le n\,,$$

ein Orthonormalsystem bezüglich $\langle\langle\cdot,\cdot\rangle\rangle$. Mit anderen Worten, die Vektoren $\mathbf{t}_k = \big[\tau_k(\theta_j)\big]_{j=0}^{N-1} \in \mathbb{C}^N$, $1-n \le k \le n$, bilden eine Orthonormalbasis des $\mathbb{C}^N$. Folglich gibt es eine Linearkombination

$$\hat{\mathbf{t}} = \sum_{k=1-n}^{n} \tilde{\alpha}_k \mathbf{t}_k = \big[f(\theta_j)\big]_{j=0}^{N-1},$$

also ein Element $\hat{t} \in \mathcal{T}_n' = \mathrm{span}\{e^{ik\theta} : 1-n \le k \le n\}$ mit $\langle\langle \hat{t}-f, \hat{t}-f\rangle\rangle = 0$. Wie in Satz 14.3 sieht man, daß t_n aus (14.3b) die Bestapproximation von f aus $\mathcal{T}_n'$ bezüglich (14.5) ist. Daher muß $\hat{t} = t_n$ sein. □

Das Polynom aus Korollar 14.4 ist das **trigonometrischen Interpolationspolynom**, für das wir im weiteren eine Fehlerdarstellung herleiten wollen. Wir beschränken uns dabei durchweg auf den wichtigsten Fall, N gerade, so daß t_n durch die Definition (14.3b) gegeben ist.

Zunächst beweisen wir das folgende Hilfsresultat (bei dem N allerdings noch ungerade sein darf).

Lemma 14.5 *Sei $f \in H_\pi^s([0,2\pi])$ für ein $s > 1/2$. Dann gilt*

$$\hat{\alpha}_k = \sum_{l=-\infty}^{\infty} \alpha_{k+lN}, \qquad -N/2 < k \le N/2.$$

Beweis. Aufgrund der Voraussetzung konvergiert die Fourierreihe von f nach Satz 13.5 gleichmäßig gegen f. Daher gilt nach (14.1) und Lemma 14.2:

$$\hat{\alpha}_k = \frac{1}{N}\sum_{j=0}^{N-1}\Big(\sum_{\nu=-\infty}^{\infty}\alpha_\nu e^{i\nu\theta_j}\Big)e^{-ik\theta_j} = \frac{1}{N}\sum_{\nu=-\infty}^{\infty}\alpha_\nu\langle\langle e^{i\nu\theta}, e^{ik\theta}\rangle\rangle = \sum_{l=-\infty}^{\infty}\alpha_{k+lN}.$$

□

Lemma 14.5 beschreibt ein Phänomen, das als **Aliasing** bekannt ist, und aus dem "Alltag" vertraut ist: Beobachtet man in einem Western-Film das Wagenrad einer anfahrenden Kutsche, dann scheinen bei einer gewissen Geschwindigkeit der Kutsche die Speichen des Rads zunächst stillzustehen, bevor sie sich langsam (scheinbar) rückwärts zu drehen beginnen – obwohl die Kutsche weiter an Geschwindigkeit zulegt. Der Film zeigt uns Bilder mit einem gewissen zeitlichen Abstand, die vom Gehirn zu einer kontinuierlichen Bildsequenz zusammengesetzt werden ("Interpolationsaufgabe"). Das Auge nimmt lediglich wahr, daß sich das Wagenrad von einem Bild zum nächsten um den Winkel θ etwa gegen die Fahrtrichtung gedreht hat. Für das Auge ist es nicht unterscheidbar, ob sich das Wagenrad zwischen den beiden Bildern wirklich um den Winkel θ gegen die Fahrtrichtung, oder vielmehr um einen

Winkel $2\pi - \theta$, $4\pi - \theta$, etc. *in Fahrtrichtung* bewegt hat. Die zugehörigen Frequenzen fallen bei der interpolierten Funktion übereinander, wie es in Lemma 14.5 bewiesen wurde.

Man kann Lemma 14.5 auch folgendermaßen interpretieren: Ein trigonometrisches Polynom vom Grad n kann nur eine bestimmte Bandbreite an Frequenzen auflösen, nämlich Frequenzen bis hin zu $2\pi/n$. Falls die zugrundeliegende Funktion wesentliche Frequenzen oberhalb dieser Schranke besitzt, muß der Grad der trigonometrischen Approximation erhöht werden, falls diese Frequenzen als solche erkannt werden sollen.

Satz 14.6 *Sei $f \in H^s_\pi([0,2\pi])$ für ein $s > 1/2$. Ist t_n das trigonometrische Interpolationspolynom* (14.3b) *an f, dann gilt*

$$\|f - t_n\|_{[0,2\pi]} \le 2\left(\frac{2s}{\pi(2s-1)}\right)^{1/2} n^{1/2-s}\,\|f\|_{H^s([0,2\pi])}\,.$$

Beweis. Wiederum haben wir gleichmäßige Konvergenz der Fourierreihe von f, so daß

$$f(\theta) \;=\; \sum_{j=-\infty}^{\infty} \alpha_j e^{ij\theta} \;=\; \sum_{l=-\infty}^{\infty} \sum_{k=1-n}^{n} \alpha_{k+lN} e^{i(k+lN)\theta}\,.$$

Daher folgt aus Lemma 14.5, daß

$$f(\theta) - t_n(\theta) \;=\; \sum_{k=1-n}^{n} \sum_{l=-\infty}^{\infty} \alpha_{k+lN} \underbrace{\left(e^{i(k+lN)\theta} - e^{ik\theta}\right)}_{=0 \text{ für } l=0}$$

und damit ist

$$|f(\theta) - t_n(\theta)| \;\le\; 2\sum_{|\nu|\ge n} |\alpha_\nu| \;=\; 2\sum_{|\nu|\ge n} |\nu|^{-s}\left(|\nu|^s|\alpha_\nu|\right).$$

Unter Verwendung der Cauchy-Schwarzschen Ungleichung ergibt sich somit

$$\begin{aligned}
|f(\theta) - t_n(\theta)|^2 &\le 4\Big(\sum_{|\nu|\ge n} |\nu|^{-2s}\Big)\Big(\sum_{|\nu|\ge n} |\nu|^{2s}|\alpha_\nu|^2\Big) \\
&\le 8\left(n^{-2s} + \int_n^\infty t^{-2s}dt\right) \sum_{\nu=-\infty}^{\infty} |\nu|^{2s}|\alpha_\nu|^2 \\
&\le \frac{4}{\pi}\left(n^{-2s} + \frac{1}{2s-1}\,n^{1-2s}\right)\|f\|^2_{H^s([0,2\pi])} \\
&\le \frac{4}{\pi}\,\frac{2s}{2s-1}\,n^{1-2s}\;\|f\|^2_{H^s([0,2\pi])}\,.
\end{aligned}$$

□

Bemerkungen. Man sieht sehr leicht ein, daß eine bessere Konvergenzordnung in der Regel nicht erwartet werden kann. So liegen in $H^s_\pi([0,2\pi])$ für $s < 1/2$ bekanntlich unstetige Funktionen; folglich kann das trigonometrische Interpolationspolynom (das ja stetig ist) nicht mehr gleichmäßig gegen eine entsprechende Funktion $f \in H^s_\pi([0,2\pi])$ konvergieren. Die Aussage von Satz 14.6 reflektiert diesen Sachverhalt dadurch, daß für $s \to 1/2$ die Konvergenzschranke schlechter wird.

Geht man zu anderen anderen Normen von $f^{(s)}$ über, lassen sich zum Teil bessere Abschätzungen beweisen (vergleiche F. A. Willers, "Methoden der praktischen Analysis", de Gruyter, 1971, §25):

$$\left\| f - t_n \right\|_{[0,2\pi]} \leq C n^{-s} \log(n) \left\| f^{(s)} \right\|_{[0,2\pi]},$$

falls $f \in C^s(\mathbb{R})$, $s \in \mathbb{N}$, und 2π-periodisch ist.

Ganz analog zu dem Beweis von Satz 14.6 kann auch die eingangs erwähnte Fehlerabschätzung für die zusammengesetzte Trapezregel bewiesen werden:

Beweis von Satz 14.1. Wir betrachten die Koeffizienten $\hat{\alpha}_k$ der trigonometrischen Polynome t_n aus (14.3) zu einer Funktion $g \in H^s_\pi([0,2\pi])$ mit der (gleichmäßig konvergenten) Fourierreihe $\sum \alpha_k e^{ik\theta}$ (N braucht hierbei nicht unbedingt gerade zu sein). Nach Lemma 14.5 gilt dann

$$\frac{1}{N} \sum_{j=0}^{N-1} g(\theta_j) \;=\; \hat{\alpha}_0 \;=\; \alpha_0 + \sum_{|l|=1}^{\infty} \alpha_{lN}\,.$$

Dabei ist $2\pi\alpha_0 = \int_0^{2\pi} g(\theta)\,d\theta$ gerade der gesuchte Integralwert von g. Daher gilt

$$\begin{aligned}
\Big| \int_0^{2\pi} g(\theta)\,d\theta \;-\; \frac{2\pi}{N} \sum_{j=0}^{N-1} g(\theta_j) \Big| \;&=\; 2\pi\,|\alpha_0 - \hat{\alpha}_0| \;\leq\; 2\pi \sum_{|l|=1}^{\infty} |\alpha_{lN}| \\
&\leq\; 2\pi \sum_{|l|=1}^{\infty} |lN|^{-s} \big| |lN|^s \alpha_{lN} \big| \\
&\leq\; 2\pi \Big(\sum_{|l|=1}^{\infty} |l|^{-2s} N^{-2s} \Big)^{1/2} \Big(\sum_{|l|=1}^{\infty} |lN|^{2s} |\alpha_{lN}|^2 \Big)^{1/2} \\
&\leq\; \frac{2\sqrt{\pi}}{N^s} \left\| f \right\|_{H^s_\pi([0,2\pi])} \Big(\sum_{k=1}^{\infty} k^{-2s} \Big)^{1/2}.
\end{aligned}$$

Da die letzte Reihe für $s > 1/2$ konvergiert, folgt hieraus die Behauptung. □

Beispiel 14.7 Abbildung 14.1 zeigt ein EKG-Signal*, bestehend aus $N = 2048$ Meßwerten.

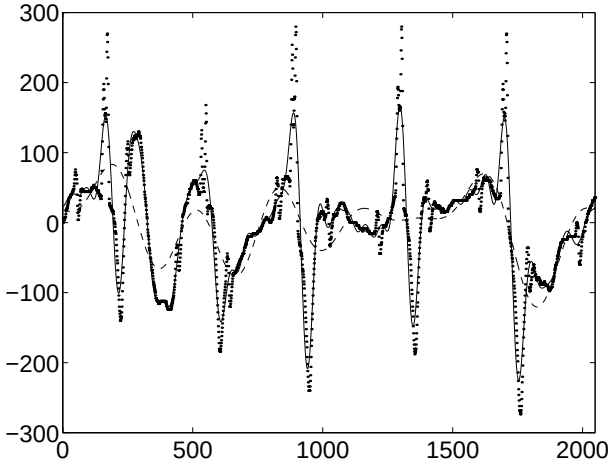


Fig. 14.1. EKG-Signal mit trigonometrischen Bestapproximationen

Nach Satz 14.3 ist die beste Approximation aus $\mathcal{T}_n$ durch (14.3) gegeben, wobei die Koeffizienten $\hat{\alpha}_k$ die gleichen sind wie für das trigonometrische Interpolationspolynom (14.3b). Abbildung 14.1 zeigt die entsprechenden Bestapproximationen aus $\mathcal{T}_8$ (gestrichelt) und $\mathcal{T}_{32}$ (durchgezogene Linie), zusammen mit den Meßwerten. Man sieht sehr deutlich, wie die scharfen Peaks in den Meßwerten mit abnehmendem Polynomgrad ausgeglättet werden. Das Polynom t_8 erfaßt allerdings nicht mehr alle wesentlichen Details der Daten: so wird zum Beispiel das Minimum am ca. 200. Datenpunkt von t_8 schlichtweg "übersehen", nicht aber von t_{32}; für t_8 ist das Signal an dieser Stelle zu hochfrequent.

15 Schnelle Fouriertransformation

Die Abbildung, die den Funktionswerten $f(\theta_j)$ einer Funktion f an den Punkten $\theta_j = 2j\pi/N$, $j = 0, \ldots, N-1$, die Koeffizienten $N\hat{\alpha}_k$ gemäß (14.1) zuordnet, wird **diskrete Fouriertransformation** genannt. Sie ist einerseits von Bedeutung, um eine geeignete Darstellung des trigonometrischen Interpolationspolynoms zu berechnen, aber auch im Hinblick auf Satz 14.3, um Approximationen mit niedrigeren Frequenzen zu bestimmen ("Glätten der Daten").

Bezeichnen wir die Daten mit $y_j = f(\theta_j)$, $j = 0, \ldots, N-1$, und führen noch die N-te Einheitswurzel $\omega = e^{-i2\pi/N}$ ein, dann kann die diskrete Fouriertransformation durch die folgende Matrixvektormultiplikation dargestellt werden:

*Die Daten wurden freundlicherweise von Prof. Dr. P. Maaß (Universität Potsdam) zur Verfügung gestellt

$$\begin{bmatrix} c_0 \\ \vdots \\ c_n \\ c_{n+1} \\ \vdots \\ c_{N-1} \end{bmatrix} := N \begin{bmatrix} \hat{\alpha}_0 \\ \vdots \\ \hat{\alpha}_n \\ \hat{\alpha}_{1-n} \\ \vdots \\ \hat{\alpha}_{-1} \end{bmatrix} = \begin{bmatrix} \omega^0 & \omega^0 & \cdots & \omega^0 \\ \omega^0 & \omega^1 & \cdots & \omega^{N-1} \\ \omega^0 & \omega^2 & \cdots & \omega^{2(N-1)} \\ \vdots & \vdots & & \vdots \\ \omega^0 & \omega^{N-1} & \cdots & \omega^{(N-1)^2} \end{bmatrix} \begin{bmatrix} y_0 \\ \vdots \\ y_{N-1} \end{bmatrix},$$

$$\text{bzw.} \qquad c = F\,y\,. \tag{15.1}$$

Die (komplexe symmetrische) Matrix F heißt **Fouriermatrix**. Aus Lemma 14.2 folgt

$$F^* F = N \cdot I\,,$$

d.h., $F/\sqrt{N}$ ist eine unitäre Matrix und

$$F^{-1} = \frac{1}{N} F^*\,. \tag{15.2}$$

Mit der Matrixvektormultiplikation (15.1) würde die Berechnung der Koeffizienten $\hat{\alpha}_k$ N^2 Multiplikationen kosten. Es gibt daneben aber auch Algorithmen, die nur mit $O(N \log N)$ Operationen auskommen; sie ergeben die sogenannte **schnelle Fouriertransformation** (FFT). Gemäß (15.2) kann die schnelle Fouriertransformation auch ausgenutzt werden, um die Werte $y_j = t(\theta_j)$ eines trigonometrischen Polynoms mit Koeffizienten $\hat{\alpha}_k$ zu berechnen: Mit den selben Vektoren c und y wie in (15.1) ist nämlich wegen der Symmetrie von F

$$y = F^{-1}c = \frac{1}{N} F^* c = \frac{1}{N} \overline{F\overline{c}}.$$

Diese Formel, von deren Gültigkeit man sich übrigens auch durch einen Vergleich von (14.1) und (14.3) vergewissern kann, ist die Grundlage für die **schnelle inverse Fouriertransformation** (IFFT).

Zur Herleitung der FFT nehmen wir im weiteren an, daß $N = 2^p$ eine Zweierpotenz ist und setzen wie zuvor $n = N/2$. Ausgangspunkt für das Verfahren ist die folgende Beobachtung.

Lemma 15.1 *Sei $M = 2m$, $\gamma_j = \sum_{\nu=0}^{M-1} \eta_\nu \omega_M^{\nu j}$, $j = 0, \ldots, M-1$, mit $\omega_M = e^{-i2\pi/M}$. Dann gilt*

$$\begin{aligned} \gamma_{2l} &= \sum_{\nu=0}^{m-1} \eta_\nu^{(+)} \omega_m^{\nu l}\,, & \eta_\nu^{(+)} &= \eta_\nu + \eta_{\nu+m}\,, \\ \gamma_{2l+1} &= \sum_{\nu=0}^{m-1} \eta_\nu^{(-)} \omega_m^{\nu l}\,, & \eta_\nu^{(-)} &= (\eta_\nu - \eta_{\nu+m})\omega_M^\nu\,, \end{aligned} \tag{15.3}$$

für $l = 0, \ldots, m-1$, wobei $\omega_m = \omega_M^2$ die entsprechende m-te Einheitswurzel ist.

Beweis. Für gerade Indizes ergibt sich zunächst

$$\gamma_{2l} \;=\; \sum_{\nu=0}^{M-1} \eta_\nu \omega_M^{\nu 2l} \;=\; \sum_{\nu=0}^{m-1} \big(\eta_\nu \omega_M^{2\nu l} + \eta_{\nu+m} \underbrace{\omega_M^{2\nu l + 2lm}}_{=\omega_M^{2\nu l}}\big) \;=\; \sum_{\nu=0}^{m-1} (\eta_\nu + \eta_{\nu+m}) \omega_m^{\nu l}\,.$$

Entsprechend erhält man für ungerade Indizes

$$\begin{aligned}\gamma_{2l+1} \;&=\; \sum_{\nu=0}^{m-1} \big(\eta_\nu \omega_M^{\nu(2l+1)} + \eta_{\nu+m} \omega_M^{(\nu+m)(2l+1)}\big) \;=\; \sum_{\nu=0}^{m-1} \big(\eta_\nu + \eta_{\nu+m} \underbrace{\omega_M^{(2l+1)m}}_{=-1}\big) \omega_M^{(2l+1)\nu} \\ &=\; \sum_{\nu=0}^{m-1} (\eta_\nu - \eta_{\nu+m}) \omega_M^{\nu} \omega_m^{\nu l}\,.\end{aligned}$$

□

Da die Summen in (15.3) wieder die gleiche Form wie die zu berechnende Summe haben (allerdings nur mit halb so vielen Summanden), kann die Berechnung rekursiv erfolgen. Dies ergibt den folgenden Algorithmus:

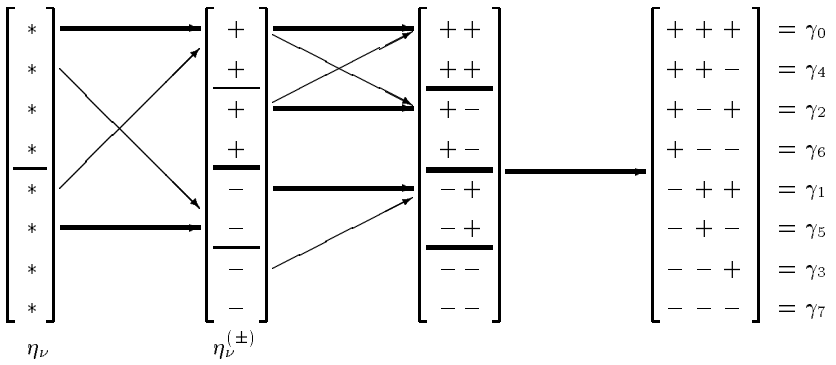
```
function γ = dft(η,ω,M)
   η^(+) = η(0 : M/2 − 1) + η(M/2 : M − 1)
   η^(−) = (η(0 : M/2 − 1) − η(M/2 : M − 1))
                          • [1, ω, ..., ω^(M/2−1)]
   if M = 2
   then
      γ = [η^(+), η^(−)]
   else
      γ(0 : 2 : M − 2) = dft(η^(+), ω^2, M/2)
      γ(1 : 2 : M − 1) = dft(η^(−), ω^2, M/2)
   end if
end    % dft
```

```
function c = fft(y, N)
   ω = e^(−i2π/N)
   c = dft(y, ω, N)
end    % fft
```

```
function y = ifft(c, N)
   c = fft(c̄, N) / N
end    % ifft
```

Die Operation • steht dabei für die komponentenweise Multiplikation zweier Vektoren.

Für eine effiziente Implementierung vermeidet man rekursive Funktionsaufrufe, da Funktionsaufrufe relativ langsam sind. Statt dessen überschreibt man auf jeder Rekursionsstufe die alten Größen η_ν (die nicht weiter gebraucht werden) mit den neuen Größen $\eta_\nu^{(\pm)}$ gemäß dem folgenden Schema (dargestellt für $N = 8$):



Man beachte, daß der Zielvektor nicht die richtige Reihenfolge hat. Ersetzt man hingegen in dem schematisch dargestellten Zielvektor jeweils "+" durch "0" und "−" durch "1" und schreibt diese Ziffern von hinten nach vorne auf, dann ergeben sich gerade die Binärdarstellungen der entsprechenden γ-Indizes:

$$\begin{array}{lccl} \gamma_0: & 000 & \longrightarrow & 000 = 0 \\ \gamma_4: & 001 & \longrightarrow & 100 = 4 \\ \gamma_2: & 010 & \longrightarrow & 010 = 2 \\ \gamma_6: & 011 & \longrightarrow & 110 = 6 \\ \vdots & \vdots & \vdots & \vdots \\ \gamma_7: & 111 & \longrightarrow & 111 = 7 \end{array}$$

Diese "bit-reversal" Methode liefert gerade die korrekte Zuordnung zwischen Speicherplatz und γ-Index.

Aufwand: Sei $N = 2^p$, also $p = \log_2 N$. Berechnet man alle Potenzen $\omega^0, \ldots, \omega^{N-1}$ im Vorfeld, dann müssen in jedem Rekursionsschritt N komplexe Additionen und $N/2$ komplexe Multiplikationen durchgeführt werden, und es gibt p Rekursionsschritte. Daher ist der Gesamtaufwand:

$$N \log_2 N \text{ komplexe Additionen}, \qquad \frac{N}{2} \log_2 N \text{ komplexe Multiplikationen}.$$

Rechnet man vier reelle Multiplikationen für eine komplexe Multiplikation und je zwei reelle Additionen für jede komplexe Multiplikation/Addition, dann ergeben sich somit

$$3N \log_2 N \text{ (reelle) Additionen}, \qquad 2N \log_2 N \text{ (reelle) Multiplikationen}.$$

Beispiel. Zum Abschluß noch ein Kommentar zu Beispiel 14.7. Um die in Abbildung 14.1 dargestellten Graphen der trigonometrischen Bestapproximationen zu berechnen, bestimmt man zunächst die FFT der $N = 2048$ Meßdaten. Für die Bestapproximation t_n vom Grad n im Sinne von Satz 14.3 werden anschließend alle Fourierkoeffizienten $\hat{\alpha}_k$ mit $|k| > n$ durch Null ersetzt. Die zugehörigen Funktionswerte $t_n(\theta_j)$ ergeben sich dann schließlich durch eine IFFT. Dabei gilt es allerdings zu beachten, daß die berechneten Werte $t_n(\theta_j)$ aufgrund von Rundefehlern in der Regel keine reellen Zahlen mehr sind; hat man jedoch alles richtig gemacht, dann ist der imaginäre Anteil im Bereich der Maschinengenauigkeit und kann getrost vernachlässigt werden.

16 Zirkulante Matrizen

Bisher haben wir die diskrete Fouriertransformation hauptsächlich im Kontext der Approximation 2π-periodischer Funktionen kennengelernt. Daneben hat die Fouriertransformation aber auch große Bedeutung in der numerischen linearen Algebra. Dies liegt letztendlich daran, daß die Vektoren $\mathbf{t}_k$, die uns im Beweis von Korollar 14.4 begegnet sind, eine Orthonormalbasis im $\mathbb{C}^N$ bilden, die für manche Anwendungen besser geeignet ist, als die herkömmliche Cartesische Basis. Wir beschränken uns im weiteren wieder auf gerade N, am besten auf Zweierpotenzen $N = 2^p$.

Zur Erinnerung: Eine **Toeplitz-Matrix** ist eine Matrix $T = [t_{j-i}]_{i,j} \in \mathbb{C}^{N\times N}$, deren Einträge entlang sämtlicher Diagonalen jeweils konstant sind.

Definition 16.1 *Eine **zirkulante Matrix** ist eine Toeplitz-Matrix $C = [c_{j-i}]_{i,j} \in \mathbb{C}^{N\times N}$ mit*

$$c_k = c_{N+k}\,, \qquad 1 - N \le k < 0\,,$$

d.h., es ist

$$C = \begin{bmatrix} c_0 & c_1 & \cdots & c_{N-2} & c_{N-1} \\ c_{N-1} & c_0 & c_1 & & c_{N-2} \\ \vdots & \ddots & \ddots & & \vdots \\ c_2 & & c_{N-1} & c_0 & c_1 \\ c_1 & c_2 & \cdots & c_{N-1} & c_0 \end{bmatrix}.$$

Zirkulante Matrizen sind dadurch ausgezeichnet, daß ihre Eigenvektoren gerade die Fouriervektoren $\mathbf{t}_k$ sind:

Satz 16.2 *Ist $C \in \mathbb{C}^{N\times N}$ eine zirkulante Matrix und F die N-dimensionale Fouriermatrix* (15.1)*, dann gilt*

$$CF^* = F^*D\,, \tag{16.1}$$

wobei die Diagonalmatrix D die Eigenwerte λ_k, $k = 0, \ldots, N-1$, von C enthält.

Beweis. Mit $\omega = e^{-2\pi i/N}$ ist die k-te Spalte (in der ungewöhnlichen Zählweise $k = 0, \ldots, N-1$) der Matrix F^* durch

$$v_k = [\,1, \omega^k, \omega^{2k}, \ldots, \omega^{(N-1)k}\,]^*$$

gegeben (bis auf den Vorfaktor $1/\sqrt{N}$ und die Indizierung stimmen die Vektoren v_k mit den Fouriervektoren $\mathbf{t}_\nu$, $\nu = 1 - N/2, \ldots, N/2$, überein). Die j-te Komponente von Cv_k, $j = 0, \ldots, N-1$, hat daher die Form

$$[Cv_k]_j \;=\; \sum_{\nu=0}^{N-1} c_{\nu-j}\omega^{-k\nu} \;=\; \omega^{-jk}\sum_{\nu=0}^{N-1} c_{\nu-j}\omega^{(j-\nu)k}\,.$$

Da nach Voraussetzung $c_{\nu-j} = c_{N-j+\nu}$ und $\omega^{(j-\nu)k} = \omega^{(j-\nu-N)k}$ ist, kann die letzte Summe umindiziert werden; es folgt, mit $c_N := c_0$,

$$[Cv_k]_j \;=\; \omega^{-jk}\sum_{\nu=0}^{N-1} c_{N-\nu}\omega^{k\nu} \;=\; \lambda_k\omega^{-jk}$$

mit

$$\lambda_k \;=\; \sum_{j=0}^{N-1} c_{N-j}\omega^{jk}\,. \tag{16.2}$$

Damit ist gezeigt, daß v_k ein Eigenvektor von C mit Eigenwert λ_k ist, $k = 0,\ldots,N-1$, und folglich gilt $CF^* = F^*D$ mit

$$D \;=\; \begin{bmatrix} \lambda_0 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \lambda_{N-1} \end{bmatrix}\,.$$

□

Die Eigenwerte einer zirkulanten Matrix sind also gerade durch (16.2) gegeben. Unter Berücksichtigung der Konvention $c_N = c_0$ ergibt dies in Vektornotation

$$\begin{bmatrix} \lambda_0 \\ \lambda_1 \\ \vdots \\ \lambda_{N-1} \end{bmatrix} \;=\; F \begin{bmatrix} c_0 \\ c_{N-1} \\ \vdots \\ c_1 \end{bmatrix}\,. \tag{16.3}$$

Offensichtlich entspricht dies gerade einer FFT, angewendet auf die erste Spalte der Matrix C. Daraus folgt: Für zirkulante $N \times N$ Matrizen können alle Eigenwerte in nur $O(N\log N)$ Operationen berechnet werden.

Auch die Matrixvektormultiplikation mit einer zirkulanten $N\times N$ Matrix kann in $O(N\log N)$ anstelle von N^2 Operationen erfolgen. Die Grundlage dazu ist die Darstellung (15.2) von F^{-1}. Somit folgt aus (16.1) die Identität

$$C \;=\; F^*DF^{-*} \;=\; F^{-1}DF\,,$$

die nach den Ergebnissen von Abschnitt 15 leicht in (I)FFTs übersetzt werden kann:

Algorithmus 16.3 (Berechnung von $y = Cx$)

- Setze $c = [\, c_0, c_{N-1}, \ldots, c_1 \,]^T$ und berechne $d = \text{FFT}\,(c)$, vgl. (16.3)
- Transformiere $z = \text{FFT}\,(x)$
- Dann ist $Dz = d \bullet z$ und $y = \text{IFFT}\,(Dz) = \text{IFFT}\,(d \bullet z)$.

Dieser Algorithmus benötigt also drei (I)FFTs und hat daher einen Aufwand von $6N \log_2 N$ Multiplikationen.

Von besonderer Bedeutung ist Algorithmus 16.3 aber vor allem wegen seiner Bedeutung für beliebige Toeplitz-Matrizen. Jede Toeplitz-Matrix $T = [t_{j-i}]_{i,j} \in \mathbb{C}^{n \times n}$ kann nämlich in eine zirkulante Toeplitz-Matrix C der Dimension $N = 2n$ eingebettet werden:

$$C = \begin{bmatrix} T & E \\ E & T \end{bmatrix} \qquad \text{mit} \quad E = \begin{bmatrix} 0 & t_{1-n} & \ldots & t_{-1} \\ t_{n-1} & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & t_{1-n} \\ t_1 & \ldots & t_{n-1} & 0 \end{bmatrix} . \tag{16.4}$$

Mit Hilfe von C läßt sich nun Tx für $x \in \mathbb{C}^n$ wie folgt ausrechnen:

$$\begin{bmatrix} Tx \\ Ex \end{bmatrix} = C \begin{bmatrix} x \\ \mathbf{0} \end{bmatrix} ,$$

wobei $\mathbf{0}$ einen Nullvektor der Dimension n bezeichnet. Dies ergibt den folgenden Algorithmus:

Algorithmus 16.4 (Berechnung von $y = Tx$)

- Setze $c = [\, t_0, t_{-1}, \ldots, t_{1-n}, 0, t_{n-1}, \ldots, t_1 \,]^T$ und berechne $d = \text{FFT}\,(c)$
- Transformiere $z = \text{FFT}\left(\begin{bmatrix} x \\ 0 \end{bmatrix} \right)$
- Berechne $\tilde{y} = \text{IFFT}\,(d \bullet z)$
- Die ersten n Komponenten von $\tilde{y}$ enthalten das Resultat $y = Tx$.

Da die Fouriertransformationen in Algorithmus 16.4 doppelte Größe $N = 2n$ haben, ist der Aufwand entsprechend höher, nämlich etwa $12n \log_2 n$ Multiplikationen.

Eine weitere Anwendung zirkulanter Matrizen, die besonders in den letzten Jahren intensiv untersucht wurde, betrifft die Lösung linearer Gleichungssysteme mit Toeplitz-Matrizen. Solche Gleichungssysteme treten (unter anderem) in der Signalverarbeitung auf.

Beispiel 16.5 Nehmen wir an, eine Antenne empfängt in regelmäßigen Abständen verrauschte Signale $y_i \in \mathbb{C}$, $i \in \mathbb{Z}$. Ziel ist die Rekonstruktion des tatsächlich ausgesendeten Signals $\hat{y}_i$. Falls aufeinanderfolgende Signalwerte nicht völlig unkorreliert sind, kann eine solche Rekonstruktion prinzipell mittels eines **endlichen linearen Filters** geschehen:

$$\hat{y}_i \approx \sum_{k=1}^{n} \xi_k y_{i-k} \,. \tag{16.5}$$

Hierbei wird also der tatsächliche Wert des Signals anhand der vorher bereits empfangenen Daten *vorhergesagt*. Die (gesuchten) Koeffizienten $\{\xi_k\}$ des Filters (16.5) sollten dabei so gewählt werden, daß einerseits Informationen über etwaige Eigenschaften des Signals verwendet werden und andererseits die Datenstörungen in den gemessenen Daten y_{i-k} nach Möglichkeit herausgefiltert werden (daher der Name *Filter*).

Für die Wahl der Koeffizienten $\{\xi_k\}$ werden daher in der Regel statistische Annahmen über das Signal getroffen: Zunächst soll o.B.d.A. angenommen werden, daß $\mathcal{E} y_i = 0$ und $\mathcal{E}\,|y_i|^2 < \infty$ für alle $i \in \mathbb{Z}$ ist. Dabei steht $\mathcal{E}$ für den statistischen Erwartungswert des darauffolgenden Arguments. Eine natürliche Forderung ist

$$\Phi(\xi_1, \ldots, \xi_n) \;:=\; \mathcal{E}\,\big|\hat{y}_i - y_i\big|^2 \longrightarrow \min \,.,$$

die sich mittels (16.5) wie folgt umformen läßt:

$$\begin{aligned}
\Phi(\xi_1, \ldots, \xi_n) \;&=\; \mathcal{E}\,\Big|\sum_{k=1}^{n} \xi_k y_{i-k} - y_i\Big|^2 \\
&=\; \mathcal{E}\,|y_i|^2 \;-\; \sum_{k=1}^{n} \xi_k\, \mathcal{E}(y_{i-k}\overline{y_i}) \;-\; \sum_{k=1}^{n} \overline{\xi}_k\, \mathcal{E}(y_i \overline{y_{i-k}}) \;+\; \sum_{j,k=1}^{n} \xi_j \overline{\xi}_k\, \mathcal{E}(y_{i-j}\overline{y_{i-k}}) \,.
\end{aligned}$$

Mit den Abkürzungen

$$A = \big[\, \mathcal{E}(y_{i-j}\overline{y_{i-k}}) \,\big]_{j,k=1}^{n} \in \mathbb{C}^{n\times n} \,, \qquad x = \big[\, \xi_k \,\big]_{k=1}^{n} \,,\; b = \big[\, \mathcal{E}(y_i \overline{y_{i-k}}) \,\big]_{k=1}^{n} \in \mathbb{C}^n \,,$$

kann das auch folgendermaßen geschrieben werden:

$$\begin{aligned}
\Phi(\xi_1, \ldots, \xi_n) \;&=\; \mathcal{E}\,|y_i|^2 \;-\; b^* x \;-\; x^* b \;+\; x^* A x \\
&=\; \mathcal{E}\,|y_i|^2 \;-\; b^* A^{-1} b \;+\; (x - A^{-1}b)^* A (x - A^{-1}b) \,.
\end{aligned} \tag{16.6}$$

Man beachte, daß die sogenannte **Kovarianzmatrix** A hermitesch und (zumindest) positiv semidefinit ist, denn

$$x^* A x \;=\; \sum_{j,k=1}^{n} \xi_j \overline{\xi}_k\, \mathcal{E}(y_{i-j}\overline{y_{i-k}}) \;=\; \mathcal{E}\,\Big|\sum_{k=1}^{n} \xi_k y_{i-k}\Big|^2 \;\geq\; 0 \,.$$

Wir werden im weiteren darüberhinaus annehmen, daß der Wert Null nur für $\xi_k = 0$, $k = 1, \ldots, n$, also nur für $x = 0$ angenommen werden kann, so daß A positiv definit ist. In diesem Fall definieren

$$\langle x, y \rangle_A := y^* A x \qquad \text{und} \qquad \|x\|_A := \langle x, x \rangle_A^{1/2}$$

bekanntlich ein Skalarprodukt und die dazugehörige Norm in $\mathbb{C}^n$; das Funktional (16.6) bekommt mit diesen weiteren Definitionen die endgültige Form

$$\Phi(\xi_1, \ldots, \xi_n) \;=\; \mathcal{E}\,|y_i|^2 \;-\; \|A^{-1}b\|_A^2 \;+\; \|x - A^{-1}b\|_A^2 \,. \tag{16.7}$$

Daran sieht man unmittelbar, daß der Vektor x mit den optimalen Koeffizienten ξ_k, $k = 1, \ldots, n$, für (16.5) das lineare Gleichungssystem

$$Ax = b \tag{16.8}$$

löst.

In der Signalverarbeitung interessiert man sich nun vor allem für Signale aus sogenannten **stationären Prozessen**; das bedeutet, daß die Korrelation zwischen den Signalwerten y_i und y_{i-k} nicht vom aktuellen Zeitpunkt (also dem Index i) abhängt. In anderen Worten,

$$\mathcal{E}(y_{i-j}\overline{y_{i-k}}) \;=\; a_{j-k} \qquad \text{für alle } i, j, k \in \mathbb{Z}$$

mit gewissen $a_\nu \in \mathbb{C}$, $\nu \in \mathbb{Z}$. Es sind also genau die stationären Prozesse, für die die Kovarianzmatrix A eine Toeplitz-Matrix ist. Das lineare Gleichungssystem (16.8) ist in diesem Fall gerade die **Yule-Walker-Gleichung**.

Unklar ist noch eine "optimale" Wahl des Parameters n in (16.5). Intuitiv ist einsichtig, daß eine größere Filterlänge n den minimalen Wert von $\mathcal{E}\,|\hat{y}_i - y_i|^2$ weiter reduziert. Unter Umständen ist es also sinnvoll, die entsprechenden Gleichungssysteme (16.8) für verschiedene, größer werdende Werte von n zu lösen. Bezeichnen wir die zugehörigen Matrizen aus (16.8) entsprechend mit A_n, um ihre Abhängigkeit von n zu dokumentieren, dann ist offensichtlich die Matrix A_n gerade die $n \times n$ linke obere Untermatrix von $A_{n'}$ für alle $n' > n$.

Abschließend sei noch darauf hingewiesen, daß bei diesem Beispiel die Beträge $|a_k|$ für größer werdende k üblicherweise schnell klein werden; dies beruht darauf, daß zwischen (zeitlich) weit auseinanderliegenden Signalwerten in der Regel fast keine Korrelation mehr vorliegt.

Der Einsatz zirkulanter Matrizen zur Lösung des Toeplitz-Gleichungssystems (16.8) geht zurück auf eine Idee von Strang. Sein Vorschlag besteht darin, die Koeffizientenmatrix A durch eine geeignete zirkulante Matrix S zu ersetzen, um dann die Lösung $x = A^{-1}b$ durch die entsprechende Näherung $\tilde{x} = S^{-1}b$ zu approximieren. Die Näherung $\tilde{x}$ kann mittels schneller Fouriertransformationen mit nur $O(n \log n)$ Operationen berechnet werden (ersetze hierzu lediglich in Algorithmus 16.3 die Einträge von d durch ihre Kehrwerte).

Der Vektor $\tilde{x}$ ist allerdings im allgemeinen keine ausreichend gute Näherung für x. Man wird daher $\tilde{x}$ durch Nachiteration weiter verbessern: Dies ergibt eine Folge $x^{(k)}$ mit

$$x^{(k+1)} \;=\; x^{(k)} + S^{-1}(b - Ax^{(k)}) \,, \qquad\qquad k = 0, 1, \ldots ,$$

wobei $x^{(0)} = S^{-1}b$, bzw. $x^{(0)} = 0$ ist. Alternativ kann auch das Verfahren der konjugierten Gradienten zur Lösung der Gleichung

$$AS^{-1}z \;=\; b\,, \qquad x = S^{-1}z\,, \tag{16.9}$$

verwendet werden; in diesem Zusammenhang nennt man S den **Vorkonditionierer** und die entsprechende CG-Variante das **vorkonditionierte CG-Verfahren**. Der Vorteil des vorkonditionierten CG-Verfahrens liegt darin, daß die Iterierten dieses Verfahrens in jedem Schritt das Fehlerfunktional (16.7) in einem gewissen Teilraum minimieren.

Unter der (oben motivierten) Annahme, daß die Einträge a_{j-i} der Toeplitz Matrix für weiter entfernte Nebendiagonalen betragsmäßig schnell abnehmen, schlägt Strang die folgende Wahl von S vor (n sei wieder gerade und $\nu = n/2$ gesetzt): Man übernimmt in S die Hauptdiagonale sowie die $\nu - 1$ ersten oberen wie unteren Nebendiagonalen von A und setzt dann S durch "wrap around" zu einer $n \times n$ zirkulanten Matrix fort:

$$S \;=\; \begin{bmatrix} a_0 & a_1 & \dots & a_{\nu-1} & 0 & \overline{a_{\nu-1}} & \dots & \overline{a_1} \\ \overline{a_1} & a_0 & \ddots & & a_{\nu-1} & 0 & & \overline{a_2} \\ \vdots & & \ddots & \ddots & & \ddots & \ddots & \vdots \\ \overline{a_{\nu-1}} & \overline{a_{\nu-2}} & & a_0 & a_1 & & a_{\nu-1} & 0 \\ 0 & \overline{a_{\nu-1}} & & \overline{a_1} & a_0 & a_1 & & a_{\nu-1} \\ \vdots & \ddots & \ddots & & \overline{a_1} & \ddots & \ddots & \vdots \\ a_2 & & \ddots & \ddots & & \ddots & a_0 & a_1 \\ a_1 & a_2 & \dots & 0 & \overline{a_{\nu-1}} & \dots & \overline{a_1} & a_0 \end{bmatrix}\,. \tag{16.10}$$

Bei der Untersuchung von $A - S$ bietet es sich an, wieder $A = A_n$ und $S = S_n$ zu schreiben, um die Abhängigkeit von n festzuhalten. Ferner setzen wir im weiteren voraus, daß die Koeffizienten a_k so schnell abfallen, daß

$$\sum_{k=-\infty}^{\infty} |a_k| \;<\; \infty\,. \tag{16.11}$$

Dann gilt der folgende Satz.

Satz 16.6 *Unter der Voraussetzung* (16.11) *existiert zu jedem* $\varepsilon > 0$ *ein gewisses* $r = r(\varepsilon) \in \mathbb{N}$, *so daß für alle* $n \in \mathbb{N}$ *Matrizen* $E_n, R_n \in \mathbb{C}^{n\times n}$ *existieren mit*

$$A_n - S_n \;=\; E_n + R_n\,, \tag{16.12}$$

wobei $\|E_n\| \le \varepsilon$ *und* $\operatorname{Rang} R_n \le 2r$ *ist.*

Beweis. Sei $\varepsilon > 0$ beliebig, aber fest. Dann wird r folgendermaßen fixiert: Gemäß der Voraussetzung (16.11) existiert $r = r(\varepsilon)$ derart, daß

$$\sum_{|k|>r} |a_k| \;<\; \varepsilon/2\,.$$

O.B.d.A. beschränken wir uns im weiteren auf den Fall $n > 2r$. Wir definieren eine Matrix C_n, indem wir alle a_k mit $|k| > r$ in (16.10) durch Null ersetzen. Natürlich ist auch C_n zirkulant und aufgrund der Konstruktion gilt

$$\|S_n - C_n\|_1 \leq \varepsilon/2\,, \qquad \|S_n - C_n\|_\infty \leq \varepsilon/2\,.$$

Damit folgt jedoch unmittelbar aus Satz **??**

$$\|S_n - C_n\|_2 \;\leq\; \Big(\|S_n - C_n\|_1\,\|S_n - C_n\|_\infty\Big)^{1/2} \;\leq\; \varepsilon/2\,. \tag{16.13}$$

Die Diagonalen von C_n, deren Werte von Null verschieden sind, beschränken sich nun auf die $2r+1$ zentralen Diagonalen, sowie auf die jeweils r äußersten Diagonalen rechts oben und links unten. Entsprechend zerlegen wir noch einmal $C_n = D_n - R_n$ in die Toeplitz-Matrix D_n, die die zentralen Diagonalwerte übernimmt und ansonsten Null ist, sowie die Toeplitz-Matrix R_n, die (bis auf das Vorzeichen) die äußeren Diagonalelemente übernimmt und ansonsten Null ist:

$$D_n = \begin{bmatrix} a_0 & \cdots & a_r & & \mathbf{0} \\ \vdots & \ddots & & \ddots & \\ \overline{a_r} & & \ddots & & \ddots \\ & \ddots & & \ddots & a_r \\ & & \ddots & & \ddots \\ \mathbf{0} & & & \overline{a_r} & \cdots & a_0 \end{bmatrix}, \qquad R_n = -\begin{bmatrix} \mathbf{0} & & & \overline{a_r} & \cdots & \overline{a_1} \\ & \ddots & & & \ddots & \vdots \\ & & \mathbf{0} & & & \overline{a_r} \\ a_r & & & \mathbf{0} & & \\ \vdots & \ddots & & & \ddots & \\ a_1 & \cdots & a_r & & & \mathbf{0} \end{bmatrix}.$$

Offensichtlich enthält das Bild von R_n nur Vektoren, bei denen höchstens die ersten r und die letzten r Komponenten von Null verschieden sein können; folglich ist der Rang von R_n höchstens $2r$. Damit haben wir

$$\begin{aligned} A_n - S_n &= A_n - C_n + C_n - S_n = \underbrace{A_n - D_n}_{E_{n,1}} + R_n + \underbrace{C_n - S_n}_{E_{n,2}} \\ &= E_{n,1} + E_{n,2} + R_n\,. \end{aligned}$$

Nach (16.13) ist $\|E_{n,2}\|_2 \leq \varepsilon/2$, und mit dem gleichen Argument ist auch $\|E_{n,1}\|_2 \leq \varepsilon/2$; folglich haben wir mit $E_n := E_{n,1} + E_{n,2}$ die gewünschte Zerlegung (16.12) gefunden. □

Unter der Voraussetzung, daß S_n invertierbar ist und ein $c > 0$ existiert mit $\|S_n^{-1}\|_2 \leq c$ für alle $n \in \mathbb{N}$ (eine Voraussetzung, die in der Praxis zumindest für hinreichend große n immer erfüllt ist), läßt sich zeigen, daß das vorkonditionierte CG-Verfahren, angewendet auf (16.9), wegen Satz 16.6 eine von n unabhängige Anzahl von Schritten benötigt, um das Minimum des Fehlerfunktionals (16.7) bis auf eine vorgegebene Toleranz δ zu finden. Da in jedem Schritt des CG-Verfahrens lediglich Matrix-Vektormultiplikationen mit S^{-1} und mit A nötig sind und diese mit Hilfe der beiden Algorithmen 16.3 und 16.4 in $O(n \log n)$ Multiplikationen durchgeführt werden können, kann der optimale lineare Filter (16.5) also mit $O(n \log n)$ Operationen im Rahmen einer vorgegebenen Genauigkeit bestimmt werden.

Beispiel 16.7 Anhand eines numerischen Beispiels sollen diese Resultate illustriert werden. Unter einem **AR(1)-Prozeß** versteht man einen stationären Prozeß

$$y_i = \rho y_{i-1} + z_i\,, \qquad i \in \mathbb{Z}\,. \tag{16.14}$$

Dabei sei $-1 < \rho < 1$ und die $z_i \in \mathbb{C}$, $i \in \mathbb{Z}$, Realisierungen unabhängiger Gauß-verteilter Zufallsvariablen mit Mittelwert Null und Varianz η^2. Man kann sich beispielsweise unter (16.14) ein sehr vereinfachtes Sprachmodell vorstellen, nach dem gesprochene Tonsequenzen proportional zur vergangenen Zeit gedämpft werden (mit dem Faktor ρ); die z_i modellieren zukünftige Sprachsequenzen, die nicht deterministisch vorhersagbar sind.

Durch Auflösen der Rekursion (16.14) ergibt sich

$$y_i \;=\; \sum_{\nu=0}^{\infty} \rho^\nu z_{i-\nu}\,,$$

so daß für $j \geq k$ aus der Unabhängigkeit der Zufallsvariablen folgt, daß

$$\mathcal{E}(y_{i-j}\overline{y_{i-k}}) \;=\; \sum_{\mu,\nu=0}^{\infty} \rho^\nu \rho^\mu \mathcal{E}(z_{i-j-\nu}\overline{z_{i-k-\mu}}) \;=\; \sum_{\nu=0}^{\infty} \rho^\nu \rho^{\nu+j-k}\eta^2 \;=\; \frac{\eta^2\rho^{j-k}}{1-\rho^2}\,.$$

Der Vektor $x \in \mathbb{C}^n$ mit den optimalen Koeffizienten des endlichen linearen Filters löst also das lineare Gleichungssystem (16.8), wobei

$$A = [a_{j-k}]_{j,k=1}^n\,, \quad a_k = \frac{\eta^2\rho^{|k|}}{1-\rho^2}\,; \qquad b = [b_j]_{j=1}^n\,, \quad b_j = \frac{\eta^2\rho^j}{1-\rho^2}\,.$$

Die Lösung x ist offensichtlich unabhängig von η, so daß wir im folgenden einfach $\eta = 1$ annehmen.

Ein Vorteil dieses simplen Beispiels liegt darin, daß die Lösung x dieses Gleichungssystems explizit angegeben werden kann; es ist nämlich

$$x = [\rho, 0, \ldots, 0]^T\,.$$

Trotzdem ist das Beispiel hinreichend allgemein, um die Effizienz des oben skizzierten Lösungsalgorithmus zu erkennen.

Betrachten wir hierzu Abbildung 16.1. Sie zeigt zunächst die Eigenwerte λ_k von A in absteigender Reihenfolge (die Kringel, aufgetragen über dem Index k am oberen Bildrand), zusammen mit den Werten des sogenannten **Symbols** $f(\theta)$ (über den Argumenten $\theta \in [0, \pi]$ am unteren Bildrand), das zu dieser Toeplitzmatrix gehört, vgl. Aufgabe **??**:

$$\begin{aligned} f(\theta) \;&:=\; \sum_{k=-\infty}^{\infty} a_k e^{ik\theta} \;=\; \sum_{k=-\infty}^{\infty} \frac{\rho^{|k|}}{1-\rho^2}\, e^{ik\theta} \;=\; \frac{1}{1-\rho^2}\Big(\sum_{k=0}^{\infty}(\rho e^{i\theta})^k + \sum_{k=0}^{\infty}(\rho e^{-i\theta})^k \;-\; 1\Big) \\ &=\; \frac{1}{1-\rho^2}\Big(\frac{1}{1-\rho e^{i\theta}} + \frac{1}{1-\rho e^{-i\theta}} \;-\; 1\Big) \;=\; \frac{1}{1-\rho^2}\Big(\frac{1-\rho e^{-i\theta}+1-\rho e^{i\theta}}{1-2\rho\cos\theta+\rho^2} \;-\; 1\Big) \\ &=\; \frac{1}{1-2\rho\cos\theta+\rho^2}\,. \end{aligned}$$

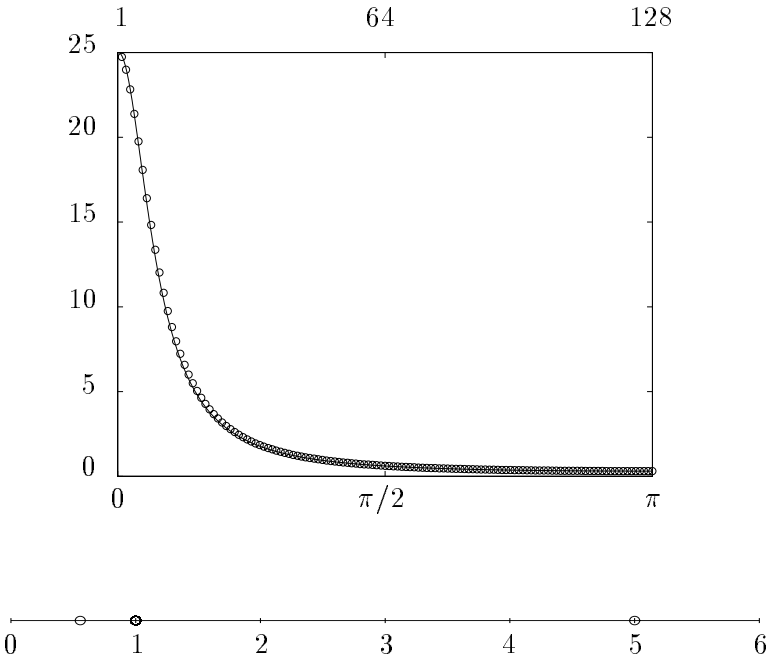


Fig. 16.1. Eigenwerte von A (oben) und AS^{-1} (unten)

f ist eine gerade Funktion mit Maximum $f(0) = (1-\rho)^{-2}$ und Minimum $f(\pi) = (1+\rho)^{-2}$. Für die Rechnung wurden die Parameter $\rho = 0.8$ und $n = 128$ gewählt. Die Übereinstimmung zwischen den Eigenwerten von A und den Werten des Symbols (dies ist Gegenstand der Aufgabe **??**) ist frappierend. Man sieht außerdem, daß das Spektrum von A den Wertebereich von f recht gleichmäßig ausfüllt; dies ist für das Verfahren der konjugierten Gradienten eine denkbar ungünstige Eigenwertverteilung.

Zum Vergleich betrachte man die Eigenwerte von AS^{-1}. Lediglich zwei der 128 Eigenwerte liegen nicht in einer ε-Umgebung um $\lambda = 1$, wobei $\varepsilon \approx 7 \cdot 10^{-6}$. Man kann dies dahingehend interpretieren, daß bei der in Satz 16.6 angesprochenen Zerlegung der Rang von R_n ungefähr zwei, und die Norm von E_n von der Größenordnung 10^{-6} ist.

Grob gesprochen wird man daher davon ausgehen, daß das (vorkonditionierte) konjugierte Gradienten Verfahren etwa zwei Iterationen benötigt, um die Eigenvektorkomponenten in x der beiden "Ausreißer-Eigenwerte" zu rekonstruieren und die verbliebenen Komponenten des Lösungsvektors danach auch schnell gefunden werden, da die anderen Eigenwerte sich derart stark um $\lambda = 1$ häufen. Dies wird durch die numerischen Resultate belegt: Abbildung 16.2 zeigt die Entwicklung des relativen Fehlers $\|x - x_k\|_2 / \|x\|_2$ für die Iterierten x_k der beiden Varianten des konjugierten Gradienten Verfahrens mit und ohne Vorkonditionierung: die durchgezogene Kurve gibt das Ergebnis mit Vorkonditionierer, die gebrochene Linie das Ergebnis ohne Vorkonditionierer wider.

Mit Vorkonditionierung benötigt das Verfahren lediglich fünf Iterationen, um die Lösung auf Maschinengenauigkeit zu bestimmen. Für die in der Praxis sicherlich ausreichende Ge-

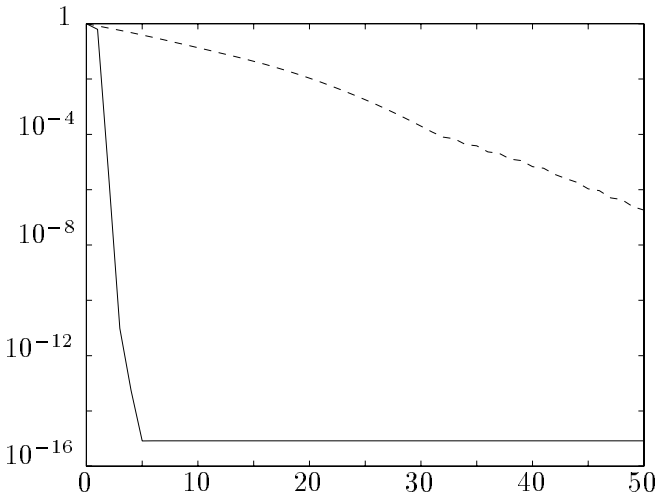


Fig. 16.2. Entwicklung des relativen Fehlers beim CG-Verfahren

nauigkeit von 10^{-6} sind bereits drei Iterationen hinreichend, während die Variante ohne Vorkonditionierung hierfür deutlich über vierzig Iterationen benötigt. Beachtet man noch, daß die Matrix-Vektorprodukte mit dem Vorkonditionierer S in der Regel billiger sind als jene mit A (da nach Algorithmus 16.4 für die Multiplikation mit A FFTs doppelter Länge notwendig sind), ergibt sich auf diese Weise eine Reduktion der ursprünglichen Rechenzeit auf fast zehn Prozent.

III. Multiskalenbasen

17 Das Haar-Wavelet

Die Bedeutung trigonometrischer Polynome oder Fourierreihen in der Numerik beruht ganz wesentlich auf zwei Eigenschaften:

- über die Fourierreihe läßt sich eine vorgegebene (periodische) Funktion elegant in einen niederfrequenten ("glatten") und einen hochfrequenten Anteil zerlegen;
- die dazu nötige Basistransformation kann mittels FFT sehr effizient durchgeführt werden.

Andererseits haben Fourierreihen auch ihre Nachteile. Das folgende Beispiel soll dies erläutern.

Beispiel. Die Fourierreihe der charakteristischen Funktion $f = \chi_{[a,b]}$ des Intervalls $[a,b] \subset (0,1)$ ist nach Beispiel 12.5 gegeben durch

$$f(x) \sim b - a + \frac{1}{\pi} \sum_{|k|=1}^{\infty} e^{-ikc} \frac{\sin kd}{k} e^{2k\pi ix}, \qquad 0 \le x \le 1,$$

wobei $c = \pi(a+b)$ und $d = \pi(b-a)$. Die Entwicklungskoeffizienten verhalten sich dabei (bis auf die durch den sinus-Term bedingte Oszillation) im wesentlichen wie $1/k$, fallen also nur sehr langsam ab; bezeichnet t_n die nach n Termen abgebrochene Reihe, dann ergibt sich der Fehler $\|f - t_n\|_{\mathcal{L}^2} \sim n^{-1/2}$, $n \to \infty$, vgl. (12.6). Für eine gute Approximation der charakteristischen Funktion sind also sehr viele Entwicklungsterme der Fourierreihe notwendig.

Das gleiche Phänomen beobachtet man bei jeder anderen $\mathcal{L}^2$-Funktion mit Sprungunstetigkeiten. Man sagt daher, daß rapide Änderungen im "Ortsbereich" einer Funktion (also bezüglich der x-Variablen) durch trigonometrische Polynome nur schlecht approximiert werden können; sie haben eine schlechte *Lokalisierungseigenschaft* bezüglich der Ortsvariablen. Besonders in der Signalverarbeitung ist dieses schlechte Lokalisierungsverhalten ein großer

Nachteil, vor allem dann, wenn man ein abrupt anfangendes, bzw. aufhörendes Signal erkennen will.

In den vergangenen Jahren wurde eine ganze Reihe alternativer Funktionenbasen vorgeschlagen, die einerseits eine Unterscheidung in hoch- und niederfrequente Komponenten erlauben, andererseits aber eine verbesserte örtlich Lokalisierung aufweisen. Auch für die Numerik sind derartige Basen von Bedeutung, vor allem in Anwendungen, in denen unterschiedliche "Längenskalen" im Ortsbereich interessant sind. Man spricht in diesem Zusammenhang von **Multiskalenbasen**. Besonders wichtig sind solche Basen für die traditionellen Splineräume. Wir werden im weiteren Multiskalenbasen für stückweise konstante und lineare Splines vorstellen.

Dazu betrachten wir die äquidistanten Zerlegungen

$$\Delta_k \;=\; \{jh_k \;:\; j = 0, 1, \ldots, 2^k; h_k = 2^{-k}\} \tag{17.1}$$

des Intervalls $[0,1]$ für $k = 0, 1, 2, \ldots, p$. Δ_{k+1} ergibt sich also durch eine einmalige Verfeinerung aus Δ_k (d.h., einer Halbierung aller Teilintervalle von Δ_k). Das feinste Gitter mit Maschenweite $h_p = 2^{-p}$ ergibt sich für $k = p$, während das gröbste Gitter Δ_0 lediglich aus einem einzigen Intervall besteht.

Mit V_k bezeichnen wir in diesem Abschnitt den Raum der stückweise konstanten Splines über Δ_k, also die Menge aller Funktionen $f \in \mathcal{L}^2(0,1)$, die im Innern aller Teilintervalle $I_{k,j} := \big[jh_k, (j+1)h_k\big]$ mit $j \in \{0, \ldots, 2^k - 1\}$ jeweils konstant sind. Eine naheliegende Basis für V_k sind die charakteristischen Funktionen der Teilintervalle $I_{k,j}$, $j = 0, \ldots, 2^k - 1$. Diese Basis (bzw. Basen, wenn wir alle $k = 0, \ldots, p$ betrachten) hat die interessante Eigenschaft, daß alle Basisfunktionen durch Verschiebung (Translation) und Stauchung aus der Grundfunktion $\chi := \chi_{[0,1]}$ hervorgehen; skaliert man die Funktionen zusätzlich so, daß die $\mathcal{L}^2$-Norm immer die selbe ist, ergeben sich die Basiselemente gerade als

$$\chi_{k,j}(x) = 2^{k/2}\chi(2^k x - j)\,, \qquad k = 0, \ldots, p\,,\; j = 0, \ldots, 2^k - 1\,;$$

etwas illustrativer ist die äquivalente Darstellung

$$\begin{gathered}\chi_{k,0}(x) = 2^{k/2}\chi(2^k x)\,, \qquad \chi_{k,j}(x) = \chi_{k,0}(x - jh_k)\,, \\ k = 0, \ldots, p\,, \quad j = 1, \ldots, 2^k - 1\,.\end{gathered} \tag{17.2}$$

Die Transformation $x \mapsto 2^k x$ bewirkt dabei eine Stauchung der Ausgangsfunktion χ, während die Transformation $x \mapsto x - jh_k$ einer Verschiebung der Grundfunktion $\chi_{k,0}$ um jh_k *nach rechts* bewirkt.

Bei festem k können wir in dem Raum V_k nur bestimmte Frequenzen einer Funktion f darstellen (ähnlich dem Raum $\mathcal{T}_n$ der trigonometrischen Polynome vom Grad kleiner gleich $n = 2^k$). Zur Darstellung höherer Frequenzen muß das Gitter Δ_k verfeinert werden; dies führt auf Δ_{k+1} und den zugehörigen Funktionenraum V_{k+1}. Nun ist aber V_k ein Unterraum von V_{k+1}; daher bietet es sich an, die Basis $\{\chi_{k,j}\}_j$ von V_k (zunächst) beizubehalten und lediglich in geeigneter Weise zu einer neuen Basis von V_{k+1} zu ergänzen. In anderen Worten: Wir suchen eine Zerlegung

$$V_{k+1} = V_k \oplus W_k \tag{17.3}$$

in den Unterraum V_k der (relativ) niederfrequenten Funktionen und einem Komplementärraum W_k der (relativ zu V_k) hochfrequenten Funktionen.

Ideal erscheint es dabei, die Zerlegung (17.3) so zu wählen, daß die beiden Teilräume V_k und W_k zueinander orthogonal sind. In Analogie zu (17.2) wäre es darüberhinaus wünschenswert, daß auch die Basisfunktionen $\psi_{k,j}$ von W_k aus einer einzigen Funktion ψ wie in (17.2) durch

$$\psi_{k,j}(x) = 2^{k/2}\psi(2^k x - j)\,, \qquad k = 0, \ldots, p\,,\ j = 0, \ldots, 2^k - 1\,,$$

beziehungsweise

$$\begin{gathered}\psi_{k,0}(x) = 2^{k/2}\psi(2^k x)\,, \qquad \psi_{k,j}(x) = \psi_{k,0}(x - jh_k)\,,\\ k = 0, \ldots, p\,, \quad j = 1, \ldots, 2^k - 1\,.\end{gathered} \tag{17.4}$$

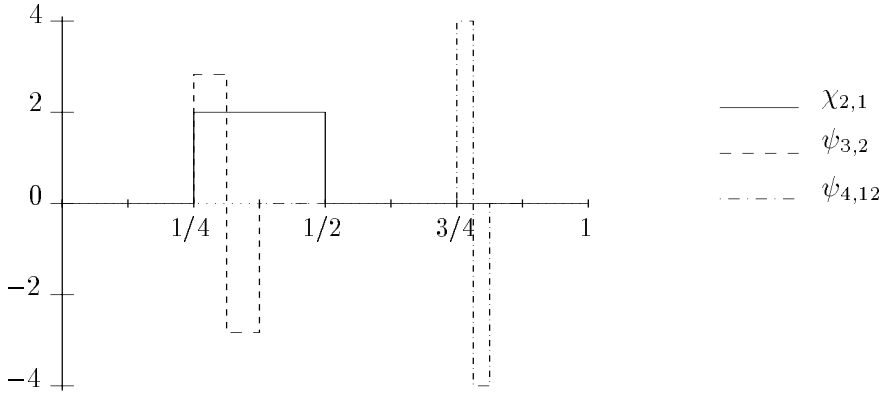
erzeugt werden können. Wegen der Forderung $W_k \subset V_{k+1}$ muß ψ ein stückweise konstanter Spline über dem **Referenzgitter**

$$\Delta_* := \{j/2 : j \in \mathbb{Z}\} \tag{17.5}$$

sein. Als geeignet erweist sich die Funktion

$$\psi(x) := \begin{cases} 1 & 0 < x \le 1/2\,,\\ -1 & 1/2 < x \le 1\,,\\ 0 & \text{sonst}\,, \end{cases} \tag{17.6}$$

das sogenannte **Haar-Wavelet***. Der englische Begriff *Wavelet* läßt sich mit "Well-chen" übersetzen, bezeichnet also eine kleine Welle.



Im weiteren bezeichnen wir mit W_k die lineare Hülle der $\{\psi_{k,j} : j = 0, \ldots, 2^k - 1\}$. Bevor wir das zentrale Resultat dieses Abschnitts beweisen werden, führen wir noch den folgenden Begriff ein:

*Die Funktion ψ aus (17.6) wurde erstmals von Haar im Jahr 1910 eingeführt; ihre Wiederentdeckung in der Wavelet-Theorie erfolgte allerdings erst in den 80er Jahren.

Definition 17.1 *Unter dem* ***Träger*** $\operatorname{supp}(f)$ *einer stückweise stetigen Funktion* f *über* $I \subset \mathbb{R}$ *verstehen wir den Abschluß aller* $x \in I$ *mit* $f(x) \neq 0$ (*engl.* support). *Der Träger von* $\chi_{k,j}$ *ist also gerade das Intervall* $I_{k,j}$.

Proposition 17.2 *Sei* $\chi = \chi_{[0,1]}$ *und* ψ *wie in* (17.6) *definiert. Dann bilden die Funktionensysteme* $\{\chi_{k,j} : j = 0, \ldots, 2^k - 1\}$ *und* $\{\psi_{k,j} : j = 0, \ldots, 2^k - 1\}$ *Orthonormalbasen* (*bezüglich* $\mathcal{L}^2$) *der Unterräume* V_k *und* W_k *von* V_{k+1}. *Zudem sind die Unterräume* V_k *und* W_k *zueinander orthogonal und bilden eine Zerlegung von* V_{k+1}.

Beweis. Sowohl $\chi_{k,j}$ als auch $\psi_{k,j}$ haben jeweils den Träger $I_{k,j}$. Da sich diese Intervalle für verschiedene j und festes k in maximal einem gemeinsamen Punkt berühren, verschwinden alle Integrale

$$\int_0^1 \chi_{k,j}\chi_{k,j'}\,dx\,, \quad \int_0^1 \psi_{k,j}\psi_{k,j'}\,dx \quad \text{und} \quad \int_0^1 \chi_{k,j}\psi_{k,j'}\,dx$$

für $j \neq j', j, j' \in \{0, \ldots, 2^k - 1\}$. Zudem gilt für $j = j'$

$$\int_0^1 \chi_{k,j}^2(x)\,dx = \int_{j2^{-k}}^{(j+1)2^{-k}} \chi^2(2^k x - j)\,2^k dx = \int_0^1 \chi^2(t)\,dt = 1\,,$$

$$\int_0^1 \psi_{k,j}^2(x)\,dx = \int_{j2^{-k}}^{(j+1)2^{-k}} \psi^2(2^k x - j)\,2^k dx = \int_0^1 \psi^2(t)\,dt = 1\,.$$

Folglich bilden die beiden Funktionensysteme in der Tat Orthonormalbasen von V_k und W_k. Für die Orthogonalität zwischen V_k und W_k verbleibt noch der Nachweis, daß $\chi_{k,j}$ und $\psi_{k,j}$ zueinander orthogonal sind; dies folgt aus der Orthogonalität von χ und ψ, vgl. (17.6):

$$\begin{aligned}\int_0^1 \chi_{k,j}(x)\psi_{k,j}(x)\,dx &= \int_{j2^{-k}}^{(j+1)2^{-k}} \chi(2^k x - j)\psi(2^k x - j)\,2^k dx = \int_0^1 \chi(t)\psi(t)\,dt \\ &= \int_0^{1/2} dt - \int_{1/2}^1 dt = 1/2 - 1/2 = 0\,.\end{aligned}$$

Klar ist schließlich, daß $W_k \subset V_{k+1}$. Da zudem $\dim V_k + \dim W_k = 2 \cdot 2^k = \dim V_{k+1}$, ist der Satz vollständig bewiesen. □

Neben der bereits genannten Basis $\{\chi_{k+1,j}\}_j$ bildet das Funktionensystem

$$\{\chi_{k,j}, \psi_{k,j} : j = 0, \ldots, 2^k - 1\}$$

folglich eine weitere Orthonormalbasis von V_{k+1} entsprechend der orthogonalen Zerlegung (17.3). Wir nennen dies die **Zweiskalenbasis** von V_{k+1}; der Anteil in V_k einer Funktion $f \in V_{k+1}$ heißt **Trend**, der Anteil von f in W_k heißt **Fluktuation** auf dem Level k.

Aus der Zweiskalenbasis erhält man eine **Multiskalenbasis**, wenn die Zerlegung (17.3) auf jedem Level rekursiv vorgenommen wird, also

$$\begin{aligned} V_p &= V_{p-1} \oplus W_{p-1} = V_{p-2} \oplus W_{p-2} \oplus W_{p-1} = \dots \\ &= V_0 \oplus W_0 \oplus W_1 \oplus \dots \oplus W_{p-1} \end{aligned} \tag{17.7}$$

mit den entsprechenden Basisfunktionen (der **Haarbasis**) von $V_0, W_0, \dots, W_{p-2}$ und W_{p-1}.

Beispiel. Zur Erläuterung der Haarbasis betrachten wir wieder das einführende Beispiel, bei dem es darum ging, die charakteristische Funktion $f = \chi_{[a,b]}$ eines Teilintervalls $[a,b] \subset (0,1)$ zu approximieren. Oben haben wir gesehen, daß zur Approximation von f sehr viele trigonometrische Funktionen benötigt werden: Es ist

$$\inf_{t \in \mathcal{T}_n} \|f - t_n\|_{\mathcal{L}^2(0,1)} \sim n^{-1/2}, \qquad n \to \infty ,$$

und für eine entsprechende Genauigkeit, müssen $\sim n$ Koeffizienten abgespeichert werden.

Mit der Haarbasis geht das wesentlich besser. Wir definieren

$$a_k := \sup\{x \in \Delta_k \,:\, x \le a\}, \qquad b_k := \inf\{x \in \Delta_k \,:\, x \ge b\},$$

und setzen dann $\varphi_k := \chi_{[a_k,b_k]}$. Aufgrund der Konstruktion gehört φ_k zu V_k und daher gilt

$$\inf_{\varphi \in V_k} \|f - \varphi\|^2_{\mathcal{L}^2(0,1)} \le \|f - \varphi_k\|^2_{\mathcal{L}^2(0,1)} \le |a - a_k| + |b - b_k| \le 2 \cdot 2^{-k} .$$

Das heißt, mit $n = 2^k$ Ansatzfunktionen erhalten wir in etwa die gleiche Approximationsgüte wie für die trigonometrische Basis. Die Bestapproximation an f aus V_k hat allerdings weit weniger als n Terme in der Basisdarstellung. Schließlich tauchen dort nur diejenigen Basisfunktionen auf, die nicht orthogonal zu f sind. Neben $\chi_{0,0}$ sind das lediglich diejenigen $\psi_{k,j}$, deren Träger entweder den Punkt a, den Punkt b oder alle beide im Innern enthalten, also höchstens zwei Basisfunktionen auf jedem Level. Somit müssen für die gleiche Approximationsgüte wie bei trigonometrischen Polynomen mit n Entwicklungskoeffizienten bei der Haar-Multiskalenbasis lediglich $O(\log n)$ Koeffizienten abgespeichert werden.

Die Frage ist nun, wie man aus der üblichen Darstellung einer Funktion $f \in V_p$ (d.h., einer Entwicklung bzgl. der $\{\chi_{p,j}\}_j$) auf die Basisdarstellung in der Haar-Basis kommt. Eine entsprechende Transformation ist jedoch leicht in Analogie zu (17.7) möglich. Zentrale Grundlage dieser Transformation sind die beiden Gleichungen

$$\begin{aligned} \sqrt{2}\,\chi_{k,j} &= \chi_{k+1,2j} + \chi_{k+1,2j+1} , \\ \sqrt{2}\,\psi_{k,j} &= \chi_{k+1,2j} - \chi_{k+1,2j+1} , \end{aligned} \qquad j = 0, \dots, n-1 , \tag{17.8}$$

wobei $n = 2^k$ ist. Entsprechend gilt dann natürlich

$$\begin{aligned} \chi_{k+1,2j} &= \tfrac{1}{\sqrt{2}}\chi_{k,j} + \tfrac{1}{\sqrt{2}}\psi_{k,j} , \\ \chi_{k+1,2j+1} &= \tfrac{1}{\sqrt{2}}\chi_{k,j} - \tfrac{1}{\sqrt{2}}\psi_{k,j} , \end{aligned} \qquad j = 0, \dots, n-1 . \tag{17.9}$$

```
function w = fhwt(x,N)
  if N = 1
  then
    w = x
  else
    ξ = ( x(0:2:N-2) + x(1:2:N-1) )/√2
    η = ( x(0:2:N-2) - x(1:2:N-1) )/√2
    w = [ fhwt(ξ, N/2) ]
        [      η       ]
  end if
end    % fhwt
```

```
function x = ifhwt(w,N)
  η = w(N/2:N-1)
  if N = 2
  then
    ξ = w(0)
  else
    ξ = ifhwt( w(0:N/2-1), N/2 )
  end if
  x(0:2:N-2) = (ξ + η)/√2
  x(1:2:N-1) = (ξ - η)/√2
end    % ifhwt
```

Algorithmus 17.1: Schnelle Haar-Wavelet Transformation

Aus diesen Basisdarstellungen schließt man leicht auf die Matrixformulierungen für die Basistransformation: Da diese Vorgehensweise in den weiteren Abschnitten immer nach dem gleichen Muster erfolgt, sei das hier einmal allgemein vorgeführt.

Lemma 17.3 *Gegeben sei eine Basis $\{\varphi_k\}$ eines N-dimensionalen Vektorraums, sowie N weitere Funktionen*

$$\phi_j = \sum_{k=1}^{N} a_{jk}\varphi_k\,, \qquad \alpha_{jk} \in \mathbb{R}\,, \quad j = 1,\ldots,N\,.$$

Dann hat die Funktion $f = \sum_{j=1}^{N} \zeta_j\phi_j$ die Basisentwicklung $f = \sum_{k=1}^{N} \xi_k\varphi_k$, wobei für die Vektoren $x = [\xi_k]_k$ und $z = [\zeta_j]_j$ im $\mathbb{R}^N$ die Beziehung

$$x = A^T z\,, \qquad A = [a_{jk}]_{j,k=1}^{N}\,,$$

gültig ist.

Beweis. Nach Voraussetzung gilt

$$f = \sum_{j=1}^{N} \zeta_j\phi_j = \sum_{j=1}^{N} \zeta_j \sum_{k=1}^{N} a_{jk}\varphi_k = \sum_{k=1}^{N} \Big(\sum_{j=1}^{N} a_{jk}\zeta_j\Big)\,\varphi_k\,.$$

Da die Basisdarstellung eindeutig ist, folgt die Behauptung. □

Ist daher eine Funktion $f = \sum_{j=0}^{N-1} \xi_{k+1,j}\chi_{k+1,j} \in V_{k+1}$ gegeben, $N = 2n = 2^{k+1}$, dann folgt aus Lemma 17.3 unmittelbar die Entwicklung

$$f = \sum_{j=0}^{n-1} \Big(\xi_{k,j}\chi_{k,j} + \eta_{k,j}\psi_{k,j}\Big)\,, \qquad \begin{aligned} \xi_{k,j} &= \tfrac{1}{\sqrt{2}}\Big(\xi_{k+1,2j} + \xi_{k+1,2j+1}\Big)\,, \\ \eta_{k,j} &= \tfrac{1}{\sqrt{2}}\Big(\xi_{k+1,2j} - \xi_{k+1,2j+1}\Big)\,, \end{aligned} \tag{17.10}$$

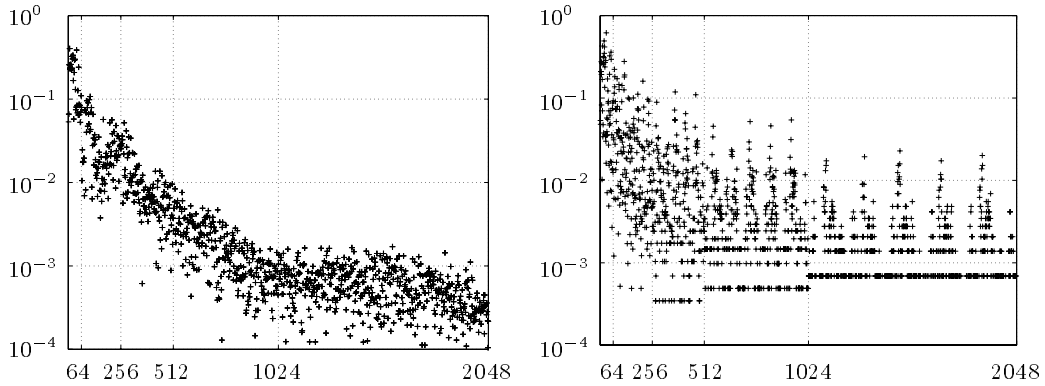


Fig. 17.1. Fouriertransformation von f (links) und Wavelettransformation (rechts)

von f in der Zweiskalenbasis und dann rekursiv die entsprechende Multiskalenentwicklung. Falls der Vektor w der Waveletkoeffizienten so angelegt wird, daß

$$w = [\xi_{0,0}, \eta_{0,0}, \eta_{1,0}, \eta_{1,1}, \eta_{2,0}, \ldots, \eta_{p,n-1}]^T, \qquad n = N/2,$$

dann kann die gesamte Basistransformation wie in Algorithmus 17.1 implementiert werden.

Die Transformation (17.10) kostet genau $2 \cdot 2^k = 2^{k+1} = N$ Multiplikationen und genauso viele Additionen. Die Mehrskalentransformation einer Funktion $f \in V_p$ kostet daher insgesamt $\sum_{k=0}^{p-1} 2^{k+1} \approx 2^{p+1} = 2 \dim V_p$ Multiplikationen wie Additionen und ist daher noch billiger zu implementieren als die FFT.

Beispiel 17.4 Wir greifen noch einmal das Beispiel 14.7 auf. Zur Trennung der hoch- und niederfrequenten Strukturen des EKG-Signals $f \in V_{11}$ aus Abbildung 14.1 kann neben der Fouriertransformation auch die Waveletdarstellung verwendet werden. Abbildung 17.1 zeigt die Absolutbeträge der jeweiligen Entwicklungskoeffizienten in einer logarithmischen Skala. Dabei sind die Entwicklungskoeffizienten in beiden Bildern der Übersicht halber nach zunehmender Frequenz der entsprechenden Basisfunktionen sortiert. Die gepunktelten vertikalen Linien zeigen die Grenzen der entsprechenden Skalen an. In beiden Fällen werden die Entwicklungskoeffizienten von f mit zunehmender Frequenz immer kleiner. Dieser Abfall ist bei den Fourierkoeffizienten etwas ausgeprägter und vor allem gleichmäßiger. Dies liegt daran, daß die Waveletdarstellung das *lokale* Frequenzverhalten analysiert. Tatsächlich sind die zum Teil erheblichen Schwankungen dieser speziellen Funktion f sehr stark ortsabhängig, und dies erkennt man nur an den Waveletkoeffizienten. Hier werden erstmals Vorteile der Waveletdarstellung deutlich.

Nicht zu vernachlässigen ist darüberhinaus der Aspekt der Datenkompression. Von den 2048 Waveletkoeffizienten sind 594, also deutlich mehr als 25% betragsmäßig kleiner als 10^{-4}; sie können daher vernachlässigt werden. Zum Vergleich: Lediglich 24 Fourierkoeffizienten, also weniger als 1.2%, liegen unterhalb dieser Toleranzschwelle.

Eine genaueres Verständnis der Wavelettransformation ist anhand von Abbildung 17.2 möglich. Dabei beschränken wir uns der Einfachheit halber auf den Anteil des EKG-Signals in V_8 (Bild oben Mitte). Die darunterliegenden Abbildungen zeigen jeweils die Anteile von f in V_k (links) und W_k (rechts), also den Trend und die jeweilige Fluktuation auf den einzelnen Skalen $k = 7$, $k = 6$, usw., bis $k = 2$ ganz unten. Man kann gut erkennen, wie sich das Ausgangssignal aus diesen einzelnen Komponenten zusammensetzt.

18 Semiorthogonale Wavelets

Trotz der einfachen Struktur hat sich das Haar-Wavelet in den Anwendungen nicht durchsetzen können (genauso wenig wie stückweise konstante Splines). Der Grund liegt in der schlechten Approximationsordnung $O(h)$, die für die Funktionenräume V_p (mit $h = 2^{-p}$) des vorigen Abschnitts charakteristisch ist ($\rightsquigarrow$ Übungen). Für numerische Anwendungen verwendet man statt dessen zumeist lineare Splines. Wir wollen daher im weiteren Multiskalenbasen für lineare Splines vorstellen.

Bei linearen Splines übernimmt die Hutfunktion

$$B(x) := \begin{cases} x & 0 < x \leq 1\,, \\ 2 - x & 1 < x \leq 2\,, \\ 0 & \text{sonst} \end{cases}$$

in natürlicher Weise die Rolle der charakteristischen Funktion aus dem vorigen Abschnitt. Dabei ist lediglich zu beachten, daß ihr Träger $\mathrm{supp}(B) = [0, 2]$ doppelt so groß ist. Zudem ergeben sich gewisse Probleme am Rand des Splinegitters: So haben wir in dem Kapitel über Splines die nodale Basis aus Hutfunktionen noch durch zwei "abgeschnittene" Randfunktionen ergänzen müssen, die in den Randpunkten den Wert eins haben und an allen anderen Gitterpunkten null sind. Dies wollen wir hier vermeiden und behelfen uns statt dessen mit der Einschränkung, lediglich periodische lineare Splines zuzulassen.

Wir bezeichnen also im weiteren mit V_k den Raum der stückweise linearen Splines über

$$\tilde{\Delta}_k := \{jh_k \,:\, j \in \mathbb{Z}; h_k = 2^{-k}\}$$

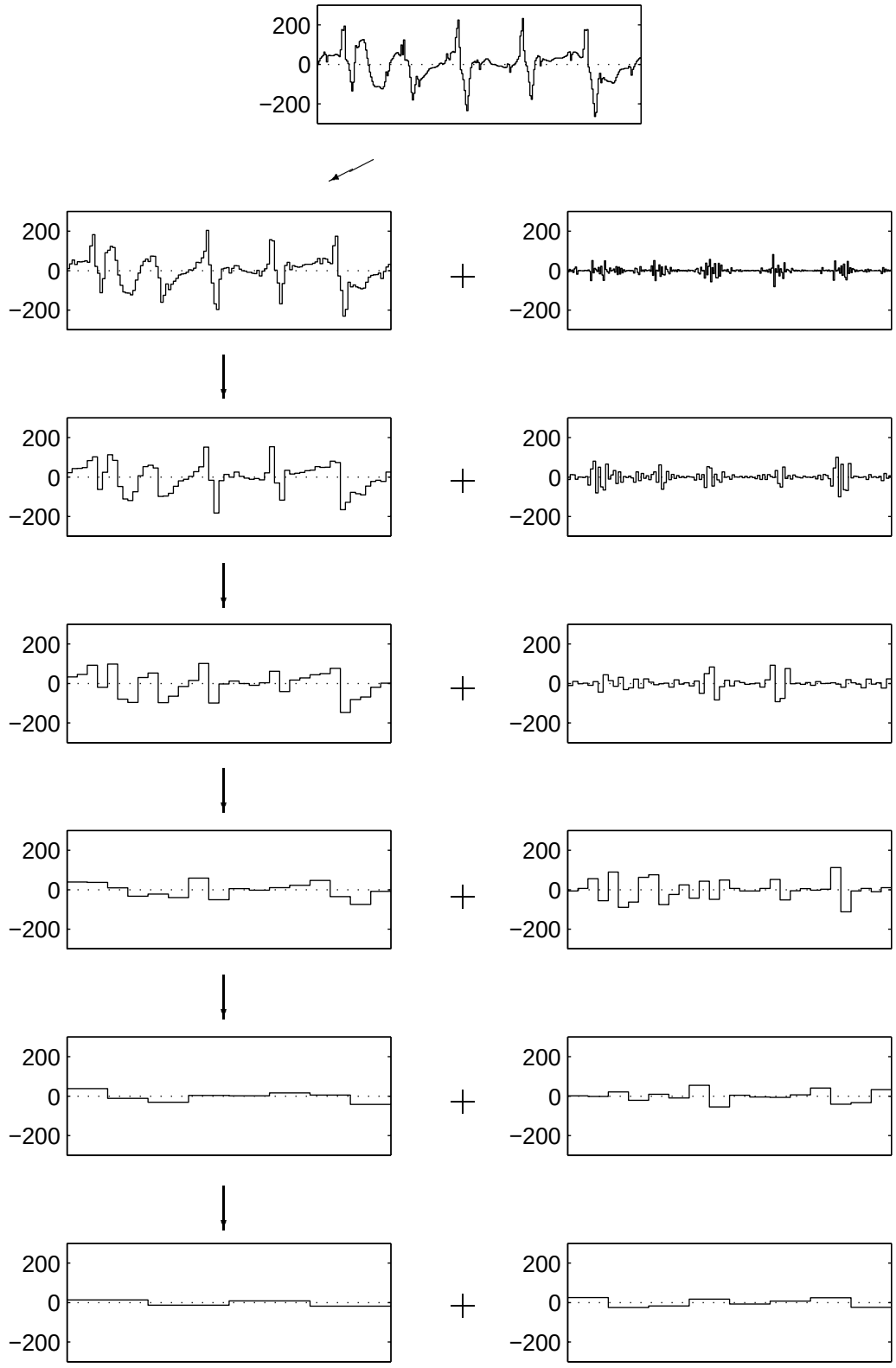
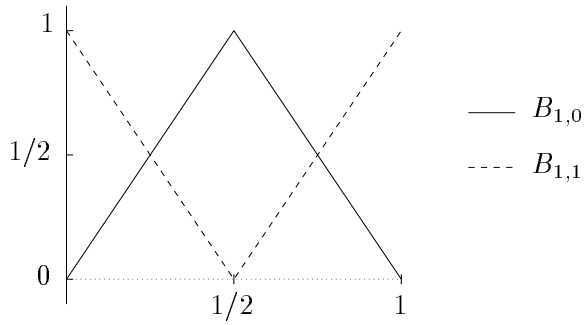


Fig. 17.2. Waveletzerlegung des EKG-Signals

mit Periode 1. Damit enthält V_0 lediglich die konstanten Funktionen, während der Raum V_1 durch zwei Basisfunktionen aufgespannt wird, $B_{1,0}$ und $B_{1,1}$:



Man beachte allerdings, daß diese Basis von V_1 keine Orthogonalbasis ist; tatsächlich müssen wir bei linearen Splines auf einige der angenehmen Orthogonalitätseigenschaften des vorigen Abschnitts verzichten.

Da V_0 nur aus konstanten Funktionen besteht, beschränken wir uns im weiteren auf die Räume V_k mit $k \geq 1$. In Analogie zum vorigen Abschnitt definieren wir in V_k mit $k \geq 1$ die sogenannte **nodale Basis** $\{B_{k,j}, j = 0, \ldots, 2^k - 1\}$ aus 1-periodischen umskalierten Hutfunktionen durch ihre Funktionswerte über $[0, 1]$:

$$\begin{aligned} B_{k,0}(x) = 2^{k/2} B(2^k x)\,, \qquad & B_{k,j}(x) = B_{k,0}(x - jh_k)\,, \qquad 0 \leq x \leq 1\,, \\ k = 0, \ldots, p\,, \quad & j = 1, \ldots, 2^k - 1\,. \end{aligned} \tag{18.1}$$

Für $k = 1$ stimmt die Definition (18.1) mit der obigen Skizze überein. Besondere Aufmerksamkeit verdient dabei die Basisfunktion $B_{1,1}$, oder allgemein die Basisfunktion $B_{k,2^k-1}$: für sie gilt nämlich

$$\mathrm{supp}(B_{k,2^k-1}) \cap [0,1] = I_{k,0} \cup I_{k,2^k-1}\,.$$

Wie im vorigen Abschnitt suchen wir nun einen Komplementärraum W_k zu V_k in V_{k+1}, d.h.,

$$V_{k+1} = V_k \oplus W_k\,.$$

Der Raum W_k soll dabei durch eine Basis $\{\psi_{k,j}, j = 0, \ldots, 2^k - 1\}$ aus 1-periodischen linearen Splines aufgespannt werden, deren Restriktion auf $[0, 1]$ durch eine geeignete Funktion ψ erzeugt wird:

$$\begin{aligned} \psi_{k,0}(x) = 2^{k/2} \psi(2^k x)\,, \qquad & \psi_{k,j}(x) = \psi_{k,0}(x - jh_k)\,, \qquad 0 \leq x \leq 1\,, \\ k = 0, \ldots, p\,, \quad & j = 1, \ldots, 2^k - 1\,. \end{aligned} \tag{18.2}$$

Wegen $W_k \subset V_{k+1}$ muß dabei ψ ein linearer Spline über dem Referenzgitter Δ_* aus (17.5) sein. Wir wollen zudem fordern, daß ψ kompakten Träger hat und k so groß ist, daß der Träger von ψ vollständig im Intervall $[0, 2^k]$ enthalten ist; damit ist garantiert, daß die

Einschränkung von $\psi_{k,0}$ auf das Intervall $[0,1]$ eine vollständige gestauchte Kopie von ψ darstellt. Für unsere Zwecke erweist sich die Einschränkung

$$\operatorname{supp}\psi \subset [0,3]\,, \qquad k \geq 2\,,$$

als ausreichend. Für größere Träger können die folgenden Überlegungen aber prinzipiell nach dem selben Schema durchgeführt werden.

Definition 18.1 *Ein linearer Spline ψ über Δ_* mit* $\operatorname{supp}(\psi) \subset [0,3]$ *heißt* ***Wavelet****, falls für jedes $k \geq 2$ der Raum*

$$W_k := span\{\psi_{k,j}\,, \,, j = 0, \ldots, 2^k - 1\} \tag{18.3}$$

einen Komplementärraum von V_k in V_{k+1} darstellt. Die den Räumen V_k zugrundeliegende Funktion B heißt ***Skalierungsfunktion****; manchmal spricht man dann auch von* ***Mutterwavelet*** *ψ und* ***Vaterwavelet*** *B.*

Ist ψ ein Wavelet, dann existiert eine Multiskalenzerlegung $V_p = V_2 \oplus W_2 \oplus W_3 \oplus \cdots \oplus W_{p-1}$; die zugehörige Basis heißt ***Waveletbasis****.*

Unter den getroffenen Annahmen ist die Funktion ψ durch fünf Parameter $\psi_1, \ldots, \psi_5$ festgelegt, nämlich die Werte

$$\psi_j = \psi(j/2)\,, \qquad j = 1, \ldots, 5\,.$$

Ferner gilt die Darstellung

$$\psi(x) = \psi_1 B(2x) + \psi_2 B(2x-1) + \psi_3 B(2x-2) + \psi_4 B(2x-3) + \psi_5 B(2x-4)\,. \tag{18.4}$$

Die Funktionen $\{B_{k,j}\}$ und $\{\psi_{k,j}\}$ auf Level k können wie in (17.8) durch die nodale Basis des $(k+1)$-ten Levels ausgedrückt werden, nämlich (für $j = 0, 1, \ldots, 2^k - 1$)*

$$\begin{aligned} \sqrt{2}\, B_{k,j} &= \tfrac{1}{2}\, B_{k+1,2j} + B_{k+1,2j+1} + \tfrac{1}{2}\, B_{k+1,2j+2}\,, \\ \sqrt{2}\, \psi_{k,j} &= \psi_1\, B_{k+1,2j} + \psi_2\, B_{k+1,2j+1} + \psi_3\, B_{k+1,2j+2} + \\ &\qquad \psi_4\, B_{k+1,2j+3} + \psi_5\, B_{k+1,2j+4}\,. \end{aligned} \tag{18.5}$$

Um diese Darstellung zu erhalten, muß man lediglich (18.4) in (18.2) einsetzen. Mit Lemma 17.3 ergibt sich nun unmittelbar eine Vorschrift für die Transformation von einer Zweiskalenentwicklung in die Darstellung bezüglich der nodalen Basis von V_{k+1}. Ist nämlich (mit $n = 2^k$ und $N = 2^{k+1}$)

$$f \;=\; \sum_{j=0}^{n-1} \Big(\xi_{k,j} B_{k,j} + \eta_{k,j} \psi_{k,j} \Big) \;=\; \sum_{j=0}^{N-1} \xi_{k+1,j} B_{k+1,j}\,,$$

dann folgt aus Lemma 17.3 die folgende Beziehung:

*Im weiteren soll der zweite Index j bei einer Basisentwicklung in $\{B_{k,j}\}_j$ bzw. $\{\psi_{k,j}\}_j$ immer modulo 2^k verstanden werden; z.B. für $j = 0$ ist $\xi_{k,-1} \equiv \xi_{k,n-1}$ mit $n = 2^k$.

$$\begin{bmatrix} \xi_{k+1,0} \\ \xi_{k+1,1} \\ \xi_{k+1,2} \\ \xi_{k+1,3} \\ \xi_{k+1,4} \\ \vdots \\ \xi_{k+1,N-2} \\ \xi_{k+1,N-1} \end{bmatrix} = \frac{1}{\sqrt{2}} \left[\begin{array}{ccccccc} 1/2 & & & & & 1/2 & \\ 1 & & & & & & \\ 1/2 & 1/2 & & & & & \\ & 1 & & & & & \\ & 1/2 & 1/2 & & & & \\ & & & \ddots & \vdots & & \\ & & & & 1/2 & 1/2 & \\ & & & & & 1 & \end{array}\right| \left[\begin{array}{ccccccc} \psi_1 & & & & & \psi_5 & \psi_3 \\ \psi_2 & & & & & & \psi_4 \\ \psi_3 & \psi_1 & & & & & \psi_5 \\ \psi_4 & \psi_2 & & & & & \\ \psi_5 & \psi_3 & \psi_1 & & & & \\ & \vdots & \vdots & \ddots & \vdots & & \\ & & & & & \psi_3 & \psi_1 \\ & & & & & \psi_4 & \psi_2 \end{array}\right| \begin{bmatrix} \xi_{k,0} \\ \xi_{k,1} \\ \vdots \\ \xi_{k,n-1} \\ \hline \eta_{k,0} \\ \eta_{k,1} \\ \vdots \\ \eta_{k,n-1} \end{bmatrix} . \tag{18.6}$$

Ohne Zusatzforderungen ist der Raum W_k aus (18.3) im allgemeinen kein Komplementärraum von V_k, also ψ kein Wavelet. Um zu erkennen, ob ψ ein Wavelet ist, bilden wir für jedes gerade $j = 0, 2, 4, \ldots, N-2$ die Summen $2\xi_{k+1,j} - \xi_{k+1,j-1} - \xi_{k+1,j+1}$ und erhalten somit aus (18.6) das $n \times n$ lineare Gleichungssystem

$$\begin{bmatrix} 2\xi_{k+1,2} - \xi_{k+1,1} - \xi_{k+1,3} \\ 2\xi_{k+1,4} - \xi_{k+1,3} - \xi_{k+1,5} \\ \vdots \\ 2\xi_{k+1,N-2} - \xi_{k+1,N-3} - \xi_{k+1,N-1} \\ 2\xi_{k+1,0} - \xi_{k+1,N-1} - \xi_{k+1,1} \end{bmatrix} = \sqrt{2} \begin{bmatrix} c_0 & c_1 & & & c_{-1} \\ c_{-1} & c_0 & c_1 & & \\ & \ddots & \ddots & \ddots & \\ & & c_{-1} & c_0 & c_1 \\ c_1 & & & c_{-1} & c_0 \end{bmatrix} \begin{bmatrix} \eta_{k,0} \\ \eta_{k,1} \\ \vdots \\ \eta_{k,n-2} \\ \eta_{k,n-1} \end{bmatrix} \tag{18.7}$$

mit

$$c_{-1} = \psi_5 - \psi_4/2, \qquad c_0 = \psi_3 - \psi_4/2 - \psi_2/2, \qquad c_1 = \psi_1 - \psi_2/2\,. \tag{18.8}$$

Satz 18.2 *Falls die $n \times n$-Matrix C aus* (18.7) *diagonaldominant ist, ist ψ ein Wavelet.*

Beweis. Eine diagonaldominante Matrix ist nicht singulär, folglich hat das lineare Gleichungssystem (18.7) in diesem Fall genau einen Lösungsvektor $\eta \in \mathbb{R}^n$. Aus den Gleichungen (18.6) für jene $\xi_{k+1,j}$ mit ungeradem j lassen sich dann die Koeffizienten $\{\xi_{k,j}\}$ durch Rücksubstitution in eindeutiger Weise bestimmen:

$$\xi_{k,j} = \sqrt{2}\,\xi_{k+1,2j+1} - \psi_4 \eta_{k,j-1} - \psi_2 \eta_{k,j}\,, \qquad j = 0, \ldots, n-1\,. \tag{18.9}$$

Dies bedeutet, daß es für jede Funktion $f \in V_{k+1}$ genau eine Darstellung als Linearkombination der $\{B_{k,j}, \psi_{k,j}\}_j$ gibt. Folglich ist $V_{k+1} = W_k \oplus V_k$ und ψ ein Wavelet. □

Aufwand: Zur Transformation von der nodalen Basis in die Zweiskalenbasis löst man zunächst das lineare Gleichungssystem (18.7) mit dem Gaußalgorithmus und erhält dann die restlichen Koeffizienten $\{\xi_{k,j}\}$ durch Rücksubstitution aus (18.9). Das Aufstellen des linearen Gleichungssystems (18.7) erfordert dabei n Multiplikationen und $2n$ Additionen; für die Gaußelimination sind dann etwa $8n + 3n$ multiplikative Operationen notwendig. Die Berechnung der $\{\xi_{k,j}\}_j$ gemäß (18.9) kostet schließlich weitere $3n$ Multiplikationen. Zusammen macht das ungefähr $15n$ Multiplikationen. Die Rücktransformation in die nodale Basis geschieht hingegen einfach via (18.6); da die entsprechende Matrix $8n$ von Null verschiedene Einträge hat, kostet diese Transformation maximal $8n$ Multiplikationen.

Die vollständige Transformation von der nodalen Basis in die Multiskalenbasis läßt sich somit durch $\sum_{k=2}^{p-1} 15 \cdot 2^k \approx 15 \cdot 2^p$ multiplikative Operationen bewerkstelligen; die Rücktransformation kostet etwa $8 \cdot 2^p$ Multiplikationen.

Welche Werte für $\psi_1, \ldots, \psi_5$ ergeben nun ein sinnvolles Wavelet? Als ein erstes Beispiel wählen wir die Parameter derart, daß die Teilräume V_k und W_k zueinander senkrecht sind. Dazu ist das folgende Lemma hilfreich:

Lemma 18.3 *Es seien φ und ψ zwei lineare Splines über dem Gitter $\{0, h, 2h, \ldots, lh\}$ mit $l \in \mathbb{N}$ und $h > 0$. Sind*

$$\varphi_j := \varphi(jh)\,, \quad \psi_j := \psi(jh)\,, \qquad j = 0, \ldots, l\,,$$

die Funktionswerte der beiden Splines, dann ist

$$\int_0^{lh} \varphi(x)\psi(x)\,dx = \frac{h}{6}\Big[\big(2\,\varphi_0 + \varphi_1\big)\,\psi_0 + \sum_{j=1}^{l-1}\big(\varphi_{j-1} + 4\,\varphi_j + \varphi_{j+1}\big)\,\psi_j + \big(2\,\varphi_l + \varphi_{l-1}\big)\,\psi_l\Big].$$

Beweis. Das Produkt $\varphi\psi$ ist auf jedem Teilintervall ein Polynom zweiten Grades, für das die Simpson-Regel eine exakte Quadraturformel ist. Daher gilt

$$\begin{aligned}
\int_0^{lh} \varphi(x)\psi(x)\,dx &= \sum_{j=1}^{l} \frac{h}{6}\left[\varphi(x_{j-1})\psi(x_{j-1}) + 4\,\varphi(\tfrac{x_{j-1}+x_j}{2})\psi(\tfrac{x_{j-1}+x_j}{2}) + \varphi(x_j)\psi(x_j)\right] \\
&= \sum_{j=1}^{l} \frac{h}{6}\left[\varphi_{j-1}\psi_{j-1} + 4\,\frac{\varphi_{j-1}+\varphi_j}{2}\,\frac{\psi_{j-1}+\psi_j}{2} + \varphi_j\psi_j\right] \\
&= \sum_{j=1}^{l} \frac{h}{6}\left[2\,\varphi_{j-1}\psi_{j-1} + \varphi_{j-1}\psi_j + \varphi_j\psi_{j-1} + 2\,\varphi_j\psi_j\right] \\
&= \frac{h}{6}\Big[\big(2\,\varphi_0 + \varphi_1\big)\,\psi_0 + \sum_{j=1}^{l-1}\big(\varphi_{j-1} + 4\,\varphi_j + \varphi_{j+1}\big)\,\psi_j + \big(2\,\varphi_l + \varphi_{l-1}\big)\,\psi_l\Big].
\end{aligned}$$

□

Damit $B_{k,j}$ und $\psi_{k,j'}$ bei festem k für alle j und j' paarweise orthogonal sind, müssen die vier Orthogonalitätsbedingungen

$$B \perp \psi(\cdot - j)\,, \qquad j = -2, -1, 0, 1\,,$$

erfüllt sein. Für alle anderen Werte von j haben die genannten Funktionen disjunkte Träger und sind daher sowieso orthogonal zueinander. Lemma 18.3 mit $\varphi = B$ und $h = 1/2$, $l = 6$, (also $\varphi_0 = 0$, $\varphi_1 = 1/2$, $\varphi_2 = 1$, $\varphi_3 = 1/2$ und $\varphi_4 = \varphi_5 = \varphi_6 = 0$) führt auf das lineare Gleichungssystem

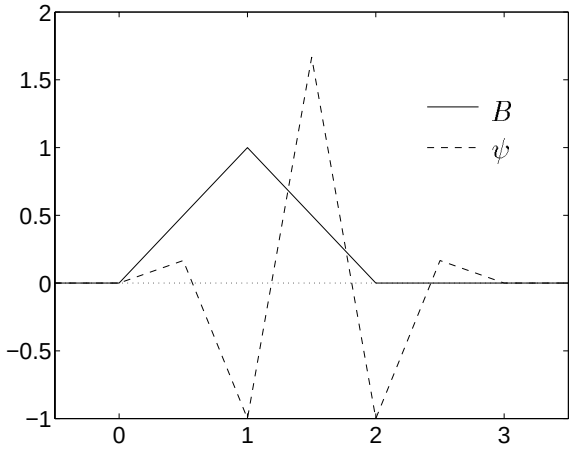


Fig. 18.1. Skalierungsfunktion B und $\mathcal{L}^2$-semiorthogonales Wavelet ψ

$$
\begin{array}{rcrcrcrcrcl}
3\,\psi_1 & + & 1/2\,\psi_2 & & & & & & & = & 0\,, \\
3\,\psi_1 & + & 5\,\psi_2 & + & 3\,\psi_3 & + & 1/2\,\psi_4 & & & = & 0\,, \\
 & & 1/2\,\psi_2 & + & 3\,\psi_3 & + & 5\,\psi_4 & + & 3\,\psi_5 & = & 0\,, \\
 & & & & & & 1/2\,\psi_4 & + & 3\,\psi_5 & = & 0
\end{array}
$$

für die Koeffizienten $\psi_j = \psi(j/2)$, $j = 1, \ldots, 5$. Es folgt $\psi_2 = -6\psi_1$, $\psi_4 = -6\psi_5$, und die verbliebenen beiden Gleichungen vereinfachen sich zu

$$
\begin{array}{rcrcrcl}
-27\,\psi_1 & + & 3\,\psi_3 & - & 3\,\psi_5 & = & 0\,, \\
-3\,\psi_1 & + & 3\,\psi_3 & - & 27\,\psi_5 & = & 0\,.
\end{array}
$$

Durch Elimination von ψ_3 ergibt sich unmittelbar $\psi_1 = \psi_5$ und damit die (bis auf einen multiplikativen Faktor eindeutige) Lösung

$$\psi_1 = 1/6, \quad \psi_2 = -1, \quad \psi_3 = 5/3, \quad \psi_4 = -1, \quad \psi_5 = 1/6\,.$$

Für die Transformation (18.7) sind die entsprechenden Einträge von C gegeben durch

$$c_0 = 8/3\,, \qquad c_{-1} = c_1 = 2/3\,. \tag{18.10}$$

Folglich ist C diagonaldominant und die Funktion ψ (mit Norm $\|\psi\|_2 = 1$) ist nach Satz 18.2 ein Wavelet; es ist in Abbildung 18.1 dargestellt. Man beachte wieder den (oszillierenden) Wellencharakter dieser Funktion.

Man beachte, daß die Unterräume der Multiskalenzerlegung

$$V_p = V_2 \oplus W_2 \oplus W_3 \oplus \cdots \oplus W_{p-1}$$

von V_p paarweise orthogonal sind. Dies gilt allerdings nicht für die ausgewählten Basisfunktionen innerhalb eines Unterraums; daher nennt man ψ **semiorthogonales Wavelet**.

19 Biorthogonale Spline-Wavelets

Die Waveletbasis des vorigen Abschnitts, die durch das semiorthogonale Wavelet ψ aus Abbildung 18.1 erzeugt wird, hat allerdings auch einen Nachteil: Wie wir später sehen werden, sind zur Darstellung einer Funktion mit kleinem Träger (etwa einer Hutfunktion $B_{p,j} \in V_p$; vgl. Definition 19.1) in der Regel alle Basiselemente dieser Waveletbasis notwendig.

Auf der positiven Seite stehen hingegen die folgenden beiden Eigenschaften:

- ψ hat kompakten Träger, $\operatorname{supp} \psi = [0, 3]$;
- Für jedes $k \geq 2$ und $j \in \mathbb{Z}$ steht $\psi_{k,j}$ orthogonal auf dem Spline-Raum V_k.

Darüberhinaus hatten wir gesehen, daß ψ die einzige Funktion mit diesen beiden Eigenschaften ist. Um ein Wavelet mit besseren Lokalisierungseigenschaften zu konstruieren, müssen wir eine der beiden obigen Eigenschaften abschwächen. Wie zuvor geht dies zu Lasten der Orthogonalität des Wavelets; der Träger von ψ bleibt unverändert.

Bevor wir mit der Herleitung solcher Wavelets beginnen, soll jedoch erst der etwas unklare Begriff der "Lokalisierungseigenschaft" genauer spezifiziert werden.

Definition 19.1 *Wir sagen, daß ein Wavelet ψ die **Lokalisierungseigenschaft** besitzt, falls $l, l' \in \mathbb{Z}$, $l \leq 0 \leq l'$, und reelle Koeffizienten α_j, β_j, $\tilde{\alpha}_j$, $\tilde{\beta}_j$ existieren mit*

$$\begin{aligned} B(2x) &= \sum_{j=l}^{l'} \big(\alpha_j B(x-j) + \beta_j \psi(x-j)\big)\,, \\ B(2x-1) &= \sum_{j=l}^{l'} \big(\tilde{\alpha}_j B(x-j) + \tilde{\beta}_j \psi(x-j)\big)\,. \end{aligned} \tag{19.1}$$

Durch Übergang auf die einzelnen Skalen einer Multiskalenbasis gemäß (18.1), (18.2), erhält man unmittelbar die Darstellungen von $B_{k+1,0}$ und $B_{k+1,1}$ in der Zweiskalenbasis auf Level $k+1$ (vorausgesetzt, daß k hinreichend groß ist):

$$B_{k+1,2i} = \sqrt{2} \sum_{j=l}^{l'} \big(\alpha_j B_{k,i+j} + \beta_j \psi_{k,i+j}\big)\,, \qquad B_{k+1,2i+1} = \sqrt{2} \sum_{j=l}^{l'} \big(\tilde{\alpha}_j B_{k,i+j} + \tilde{\beta}_j \psi_{k,i+j}\big)\,.$$

Jede Basisfunktion der nodalen Basis auf Level $k+1$ kann also durch eine feste Anzahl von Basisfunktionen der Zweiskalenbasis ausgedrückt werden. Mit anderen Worten: Eine Störung einer Funktion $f \in V_p$ in einem einzigen Gitterpunkt $x \in \Delta_p$ beeinflußt lediglich die Koeffizienten jener Basisfunktionen der Waveletbasis, deren Träger in unmittelbarer Nachbarschaft des Punktes x liegen.

Wavelets, die die Lokalisierungseigenschaft erfüllen, lassen sich wie folgt charakterisieren.

Satz 19.2 *Ein lineares Spline-Wavelet ψ über Δ_* mit* $\operatorname{supp}\psi \subset [0,3]$ *und Werten $\psi_j = \psi(j/2)$, $j = 1,\ldots,5$, hat genau dann die Lokalisierungseigenschaft, falls genau zwei der drei Parameter c_{-1}, c_0 und c_1 aus* (18.8) *gleich Null sind.*

Beweis. Nehmen wir zunächst an, daß c_{-1} und c_1 gleich Null sind. Gemäß (18.8) ist dies äquivalent zu den Bedingungen

$$\psi_2 = 2\psi_1 \quad \text{und} \quad \psi_4 = 2\psi_5\,. \tag{19.2}$$

Eingesetzt in (18.7) folgt daraus unmittelbar (mit $n = 2^k$)

$$\eta_{k,j} \;=\; \frac{1}{c_0\sqrt{2}}\left(2\,\xi_{k+1,2(j+1)} - \xi_{k+1,2(j+1)-1} - \xi_{k+1,2(j+1)+1}\right), \qquad j = 0,\ldots,n-1\,. \tag{19.3}$$

Speziell für die Entwicklungskoeffizienten $\xi_{k+1,j} = \delta_{j0}$ von $B_{k+1,0}$ ergibt sich dann

$$\eta_{k,n-1} = \sqrt{2}/c_0\,, \qquad \eta_{k,j} = 0, \quad j \neq n-1\,,$$

und aus (18.9) folgt ferner

$$\xi_{k,0} = -\,\frac{\psi_4\sqrt{2}}{c_0}\,, \quad \xi_{k,n-1} = -\,\frac{\psi_2\sqrt{2}}{c_0}\,; \qquad \xi_{k,j} = 0, \quad j \notin \{0, n-1\}\,.$$

Damit hat $B_{k+1,0}$ die Basisentwicklung

$$B_{k+1,0} \;=\; -\,\frac{\sqrt{2}}{c_0}\left(\psi_2 B_{k,n-1} + \psi_4 B_{k,0} - \psi_{k,n-1}\right);$$

setzt man hier $B_{k+1,0}(x) = 2^{(k+1)/2}B(2^{k+1}x)$ ein und ersetzt $2^{k+1}x$ durch t, dann folgt die erste Gleichung aus (19.1) mit $l = -1$ und $l' = 0$, sowie geeigneten α_j und β_j. Entsprechend zeigt man, daß

$$B_{k+1,1} \;=\; \frac{1}{c_0\sqrt{2}}\left(\psi_2 B_{k,n-1} + (2c_0 + \psi_4 + \psi_2)B_{k,0} + \psi_4 B_{k,1} - \psi_{k,n-1} - \psi_{k,0}\right),$$

und erhält hieraus die zweite Gleichung von (19.1) mit $l = -1$ und $l' = 1$. Mit anderen Worten, das Wavelet ψ erfüllt die Lokalisiserungseigenschaft. Der Beweis der Lokalisierungseigenschaft in den anderen beiden Fällen $c_{-1} = c_0 = 0$, bzw. $c_0 = c_1 = 0$, geht ganz analog.

Hat umgekehrt das Wavelet ψ die Lokalisierungseigenschaft, dann existieren nach Definition 19.1 Koeffizienten α_j und β_j mit

$$B_{k+1,2i} \;=\; \sqrt{2}\sum_{j=l}^{l'}\left(\alpha_j B_{k,i+j} \;+\; \beta_j \psi_{k,i+j}\right)$$

für hinreichend große k. Gemäß (18.7) erfüllen die β_j dann das folgende lineare Gleichungssystem:

$$\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 2 \end{bmatrix} = 2 \begin{bmatrix} c_0 & c_1 & & & c_{-1} \\ c_{-1} & c_0 & c_1 & & \\ & \ddots & \ddots & \ddots & \\ & & c_{-1} & c_0 & c_1 \\ c_1 & & & c_{-1} & c_0 \end{bmatrix} \begin{bmatrix} \vdots \\ \beta_{l'} \\ 0 \\ \beta_l \\ \vdots \end{bmatrix} . \tag{19.4}$$

Ohne Beschränkung der Allgemeinheit soll dabei k so groß gewählt werden, daß wir l und l' in folgendem Intervall fixieren können:

$$-n/2 + 1 < l \leq l' < n/2 - 1 , \qquad n = 2^k .$$

Ferner sei der Einfachheit halber angenommen, daß $\beta_l \beta_{l'} \neq 0$.

In (19.4) gibt es genau eine Gleichung, in der c_{-1} mit $\beta_{l'}$ multipliziert wird:

$$\text{linke Seite } = 2\big(c_{-1}\beta_{l'} + c_0 \cdot 0 + c_1 \cdot 0\big) . \tag{19.5}$$

Falls die linke Seite von (19.5) gleich Null ist, ergibt sich zwangsläufig $c_{-1} = 0$, da $\beta_{l'} \neq 0$ angenommen ist. Die linke Seite ist genau dann gleich Null, wenn (19.5) nicht zufällig die letzte Zeile von (19.4) darstellt, also genau dann, wenn $l' \neq -2$. Demnach ist $l' = -2$ oder $c_{-1} = 0$.

Entsprechend gibt es genau eine Gleichung in (19.4), in der c_1 mit β_l multipliziert wird:

$$\text{linke Seite } = 2\big(c_{-1} \cdot 0 + c_0 \cdot 0 + c_1\beta_l\big) ,$$

und wie zuvor ergibt sich $l = 0$ oder $c_1 = 0$.

Drei Fälle gilt es nun zu unterscheiden: Im ersten Fall ist $l \neq 0$ und $l' \neq -2$ (und somit $c_1 = c_{-1} = 0$ und die Behauptung nachgewiesen), im zweiten Fall ist $l = 0$, und im dritten Fall ist $l' = -2$. Betrachten wir etwa den Fall $l = 0$. Wegen $l \leq l'$ ist dann $l' \neq -2$ und daher $c_{-1} = 0$. Die erste Gleichung des Gleichungssystems (19.4) lautet nun wegen $l = 0$ wie folgt:

$$0 = c_0\beta_0 + c_1\beta_1 = c_0\beta_l + c_1 \cdot 0 .$$

Damit ergibt sich aber notwendigerweise $c_0 = 0$, und somit ist $c_0 = c_{-1} = 0$ und die Behauptung erfüllt. Der verbliebene Fall $l' = -2$ führt entsprechend auf die Bedingung $c_0 = c_1 = 0$. □

Beispiel. Für das semiorthogonale Wavelet aus Abbildung 18.1 sind c_0, c_{-1} und c_1 jeweils von Null verschieden, vgl. (18.10). Daher hat das semiorthogonale Wavelet *nicht* die Lokalisierungseigenschaft.

Zur Konstruktion eines Wavelet ψ mit der Lokalisierungseigenschaft beschränken wir uns im weiteren auf den Fall $c_{-1} = c_1 = 0$, da nur dieser auf ein achsensymmetrisches Wavelet

führt. In diesem Fall muß c_0 von Null verschieden sein: Man macht sich mittels (18.8) nämlich schnell klar, daß andernfalls ψ nicht nur ein linearer Spline über Δ_*, sondern sogar über dem nächstgröberen Gitter ist und damit nicht linear unabhängig zu allen Translaten der Hutfunktion sein kann.

Die Bedingung $c_{-1} = c_1 = 0$ führt auf die beiden Gleichungen (19.2) und diese stellen zwei lineare Bedingungen an die fünf gesuchten Koeffizienten ψ_j, $j = 1, \ldots, 5$, von ψ dar. Da ψ ohnehin höchstens bis auf (multiplikative) Normierung eindeutig ist, können lediglich zwei weitere Nebenbedingungen an ψ gestellt werden, zum Beispiel Orthogonalitätsbedingungen. Dabei ist aber zu beachten, daß diese Bedingungen nicht auf $c_0 = 0$ führen dürfen.

Zwei weitere Nebenbedingungen an ψ sind allerdings zu wenig, um Orthogonalität von ψ zu allen Translaten von B zu erzwingen; dazu wären mindestens drei Nebenbedingungen notwendig. Daher gibt man andere Orthogonalitätsbedingungen vor, etwa

$$\psi \perp \Pi_1 .$$

(Man beachte, daß Π_1, der Raum aller Polynome ersten Grades, ein Teilraum der linearen Splines ist.) Π_1 wird von der Konstanten $y \equiv 1$ und der Geraden $y = x$ erzeugt; Die beiden Orthogonalitätsbedingungen lauten also

$$\int_0^3 \psi(x)\, dx = 0 \qquad \text{und} \qquad \int_0^3 x\psi(x)\, dx = 0 \tag{19.6}$$

und führen gemäß Lemma 18.3 auf die beiden Bedingungsgleichungen

$$\begin{array}{rcrcrcrcrcl} \psi_1 & + & \psi_2 & + & \psi_3 & + & \psi_4 & + & \psi_5 & = & 0\,, \\ \psi_1 & + & 2\,\psi_2 & + & 3\,\psi_3 & + & 4\,\psi_4 & + & 5\,\psi_5 & = & 0\,. \end{array}$$

Zusammen mit (19.2) ergibt sich die (bis auf Vielfache) eindeutig bestimmte Lösung

$$\psi_1 = -1/6\,, \quad \psi_2 = -1/3\,, \quad \psi_3 = 1\,, \quad \psi_4 = -1/3\,, \quad \psi_5 = -1/6\,.$$

Damit ist $c_0 = 4/3$ in (18.8).

Die Transformation von der nodalen Basis in die Zweiskalenbasis erfolgt anhand der Gleichungen (19.3) und (18.9) und kostet $4n$ multiplikative Operationen (wenn man in (18.9) den gemeinsamen Faktor $\psi_2 = \psi_4 = -1/3$ ausklammert). Die Rücktransformation gemäß (18.6) kostet $5n$ Multiplikationen.

Das so bestimmte Wavelet ψ findet man in der Literatur unter dem Namen **biorthogonales Wavelet**. Der Grund liegt in der sogenannten Biorthogonalitätsrelation (19.7) des folgenden Satzes, den wir hier allerdings nicht beweisen wollen und können.

Satz 19.3 *Es gibt zwei Funktionen* $\tilde{B}, \tilde{\psi} \in \mathcal{L}^2(\mathbb{R})$ *mit* $\operatorname{supp} \tilde{B} \subset [-1, 3]$ *und* $\operatorname{supp} \tilde{\psi} \subset [-1, 2]$, *so daß*

$$\begin{gathered} \psi_{k,j} \perp \tilde{\psi}_{k',j'}\,, \quad (k,j) \neq (k',j')\,, \\ B_{k,j} \perp \tilde{B}_{k,j'}\,, \quad j \neq j', \qquad B_{k,j} \perp \tilde{\psi}_{k',j'}\,, \quad k' \geq k\,. \end{gathered} \tag{19.7}$$

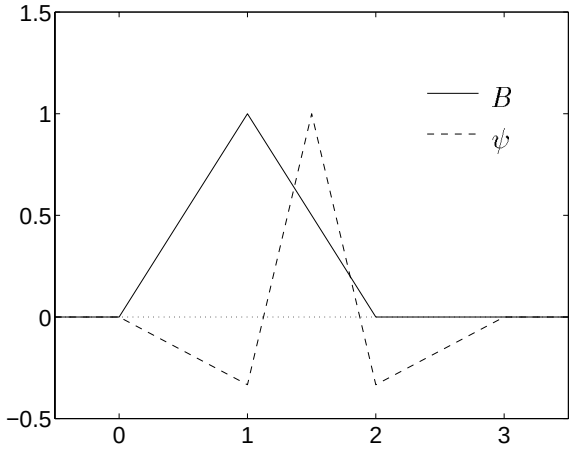


Fig. 19.1. Skalierungsfunktion B und biorthogonales Wavelet ψ

Hierbei werden $\tilde{B}_{k,j}$ und $\tilde{\psi}_{k,j}$ wie zuvor durch Stauchung und Translation aus $\tilde{B}$ und $\tilde{\psi}$ erzeugt. Darüberhinaus genügen (für $k \geq 2$) die Funktionen $\tilde{B}_{k,j}$ und $\tilde{\psi}_{k,j}$ den Rekursionsgleichungen

$$\begin{aligned} \tilde{B}_{k-1,0} &= -\tfrac{3}{2\sqrt{2}}\left(\tfrac{1}{6}\,\tilde{B}_{k,-1} - \tfrac{1}{3}\,\tilde{B}_{k,0} - \tilde{B}_{k,1} - \tfrac{1}{3}\,\tilde{B}_{k,2} + \tfrac{1}{6}\,\tilde{B}_{k,3}\right), \\ \tilde{\psi}_{k-1,0} &= -\tfrac{3}{2\sqrt{2}}\left(\tfrac{1}{2}\,\tilde{B}_{k,-1} - \tilde{B}_{k,0} + \tfrac{1}{2}\,\tilde{B}_{k,1}\right). \end{aligned} \tag{19.8}$$

Die Funktionen $\tilde{B}$ und $\tilde{\psi}$ werden **duale Skalierungsfunktion**, bzw. **duales Wavelet** genannt. Ihre Bedeutung liegt in der folgenden Eigenschaft: Ist

$$f = \sum_{j=0}^{3} \xi_{2,j} B_{2,j} + \sum_{k=2}^{p-1} \sum_{j=0}^{2^k-1} \eta_{k,j} \psi_{k,j}$$

die Darstellung von f in der Waveletbasis von $V_p = V_2 \oplus W_2 \oplus \ldots \oplus W_{p-1}$, dann folgt aus (19.7), daß

$$\langle f, \tilde{\psi}_{k,j} \rangle = \eta_{k,j} \langle \psi_{k,j}, \tilde{\psi}_{k,j} \rangle = \eta_{k,j} \langle \psi, \tilde{\psi} \rangle ,$$

beziehungsweise

$$\eta_{k,j} = \frac{\langle f, \tilde{\psi}_{k,j} \rangle}{\langle \psi, \tilde{\psi} \rangle} , \qquad k \geq 2 .$$

Die Waveletkoeffizienten der Multiskalendarstellung ergeben sich also aus Innenprodukten von f mit den entsprechenden dualen Wavelets.

Dies ist allerdings in erster Linie von theoretischer Bedeutung, da man $\tilde{B}$ und $\tilde{\psi}$ nicht explizit kennt. Man kann die beiden dualen Funktionen aber numerisch approximieren. Durch

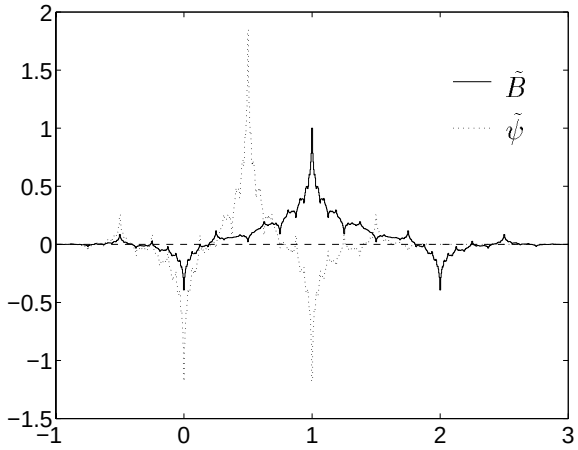


Fig. 19.2. Duale Skalierungsfunktion $\tilde{B}$ und duales Wavelet $\tilde{\psi}$

die Transformation $\tilde{B}_{k,j} = 2^{k/2}\tilde{B}(2^k \cdot -j)$ erhält man nämlich aus (19.8) eine sogenannte **Skalierungsgleichung**

$$\begin{aligned}\tilde{B}(x) \;=\; -\frac{1}{4}\,\tilde{B}(2x+1) + \frac{1}{2}\,\tilde{B}(2x) + \frac{3}{2}\,\tilde{B}(2x-1) \\ + \frac{1}{2}\,\tilde{B}(2x-2) - \frac{1}{4}\,\tilde{B}(2x-3)\end{aligned} \tag{19.9}$$

für $\tilde{B}$, die numerisch gelöst werden kann ($\rightsquigarrow$ Übungen). Das Ergebnis ist in Abbildung 19.2 dargestellt. Man beachte das "fraktale Aussehen" von $\tilde{B}$.

20 Ein Anwendungsbeispiel

Als eine Anwendung der Waveletbasen betrachten wir folgende Aufgabe: Gesucht sei eine Funktion u mit

$$\Delta u = 0 \quad \text{in} \quad \mathbb{R}^2 \setminus \overline{\Omega}\,, \qquad u = g \quad \text{auf} \quad \Gamma = \partial\Omega\,. \tag{20.1}$$

Hierbei ist Δ der **Laplace-Operator**,

$$\Delta u = \frac{\partial^2}{\partial x_1^2}\,u + \frac{\partial^2}{\partial x_2^2}\,u\,, \qquad x = (x_1, x_2)\,,$$

und das Gebiet Ω bezeichne entweder den Einheitskreis oder eine Ellipse um den Nullpunkt mit Halbachsen der Länge α und β.

Das Problem (20.1) ist eine partielle Differentialgleichung (**Laplace-Gleichung**) im *Außenraum*, also in einem unbeschränkten Gebiet. Ist beispielsweise g die Spannungsvorgabe

auf einer geschlossenen Oberfläche Γ, dann beschreibt die Lösung u von (20.1) das Potential des resultierenden elektrischen Felds $E = -\operatorname{grad} u$. Für eine physikalisch sinnvolle Lösung dieses Problems muß zusätzlich noch gefordert werden, daß $u(x)$ für $|x| \to \infty$ beschränkt bleibt*.

Ein elegantes numerisches Lösungsverfahren für die Differentialgleichung (20.1) ist die sogenannte **Randelementmethode**. Dazu macht man einen Lösungsansatz der Form

$$u(x) = \int_\Gamma \Big(1 + \frac{\partial G(x,y)}{\partial \nu(y)}\Big)\, \varphi(y)\, ds(y)\,, \qquad x \in \mathbb{R}^2 \setminus \overline{\Omega}\,. \tag{20.2}$$

Die Integration über Γ bezüglich der Variablen y ist dabei bezüglich der Bogenlänge im mathematisch positiven Sinn zu verstehen; die Funktion

$$G(x,y) \;=\; \frac{1}{2\pi}\, \log \frac{1}{|x-y|} \tag{20.3}$$

ist die sogenannte **Grundlösung** der Laplace-Gleichung und $\partial/\partial\nu(y)$ bezeichnet die Ableitung bezüglich y in Richtung der äußeren Normalen ν des Gebiets Ω.

Lemma 20.1 *φ sei eine stetige Funktion. Dann ist u unendlich oft differenzierbar in $\mathbb{R}^2 \setminus \overline{\Omega}$ und erfüllt die Laplacegleichung $\Delta u = 0$. Ferner bleibt u beschränkt für $|x| \to \infty$.*

Beweis. In der Umgebung eines Punktes $x \in \mathbb{R}^2 \setminus \overline{\Omega}$ ist der Integrand von (20.2) beliebig oft nach x differenzierbar und die partiellen Ableitungen von u können durch Vertauschung von Integration und Differentiation bestimmt werden. Da weiterhin bei der Differentiation von G auch die Differentiationsreihenfolge bezüglich x und y vertauscht werden kann, ergibt sich

$$\Delta u(x) = \int_\Gamma \varphi(y)\, \frac{\partial}{\partial \nu(y)}\, \Delta_x G(x,y)\, ds(y)\,; \tag{20.4}$$

hierbei bezeichnet Δ_x den Laplace-Operator bezüglich der x-Variablen. Eine einfache Rechnung ergibt

$$\frac{\partial}{\partial x_i}\, G(x,y) \;=\; -\,\frac{1}{2\pi}\, \frac{\partial}{\partial x_i}\, \frac{1}{2}\, \log |x-y|^2 \;=\; -\,\frac{1}{2\pi}\, \frac{x_i - y_i}{|x-y|^2} \tag{20.5}$$

und weiterhin

$$\frac{\partial^2}{\partial x_i^2}\, G(x,y) \;=\; \frac{1}{2\pi}\Big(\frac{x_i - y_i}{|x-y|^4}\, 2(x_i - y_i) \;-\; \frac{1}{|x-y|^2}\Big)\,, \qquad i = 1,2\,.$$

Durch Summation ergibt sich daher

$$\Delta_x\, G(x,y) \;=\; \frac{1}{2\pi}\Big(\frac{2(x_1-y_1)^2 + 2(x_2-y_2)^2}{|x-y|^4} \;-\; 2\,\frac{1}{|x-y|^2}\Big) \;=\; 0\,,$$

*In diesem Abschnitt bezeichnet $|x|$, $x \in \mathbb{R}^2$, immer die Euklidnorm im $\mathbb{R}^2$.

und eingesetzt in (20.4) ergibt dies die Behauptung $\Delta u(x) = 0$.

Aus (20.5) schließt man ferner unmittelbar, daß $|\operatorname{grad}_y G(x,y)| = |-\operatorname{grad}_x G(x,y)| = O(1/|x|)$, gleichmäßig für $|x| \to \infty$; da der Normalenvektor die Norm $|\nu(y)| = 1$ hat, ergibt sich

$$1 + \frac{\partial G(x,y)}{\partial \nu(y)} \;=\; 1 + O(1/|x|)$$

und damit folgt aus (20.2)

$$u(x) \longrightarrow \int_\Gamma \varphi(y)\, ds(y)\,, \qquad |x| \to \infty\,,$$

und zwar gleichmäßig in $|x|$. □

Um die Differentialgleichung (20.1) zu lösen, gilt es lediglich noch die Randbedingung $u|_\Gamma = g$ zu erfüllen. An dieser Stelle tritt allerdings eine Schwierigkeit auf. Dies sieht man am einfachsten, wenn man für Ω den Einheitskreis wählt. In dem Fall ist $\nu(y) = y$, $y \in \Gamma$, und aus (20.5) ergibt sich

$$\frac{\partial}{\partial \nu(y)} G(x,y) \;=\; \nu(y) \operatorname{grad}_y G(x,y) \;=\; \frac{1}{2\pi}\Big(\frac{x_1 y_1 - y_1^2}{|x-y|^2} + \frac{x_2 y_2 - y_2^2}{|x-y|^2}\Big)\,. \tag{20.6}$$

Speziell für $x = \rho y$ mit $y \in \Gamma$ und $\rho > 1$ vereinfacht sich das zu

$$\frac{\partial}{\partial \nu(y)} G(\rho y, y) \;=\; \frac{1}{2\pi}\,(\rho - 1)\,\frac{y_1^2 + y_2^2}{(\rho-1)^2 |y|^2} \;=\; \frac{1}{2\pi}\,\frac{1}{\rho - 1} \;\longrightarrow +\infty\,, \qquad \rho \to 1\,.$$

Mit anderen Worten, wenn x einem Punkt $x^* \in \Gamma$ aus radialer Richtung nahe kommt, dann ist der Integrand von (20.2) für $y \approx x^*$ von der Größenordnung $1/|x^* - x|$. Die Bestimmung des Randwerts $u(x^*)$ mit $x^* \in \Gamma$ gemäß (20.2) ist also eine nichttriviale Angelegenheit.

Um die Verwirrung zu vervollständigen, betrachten wir noch $\partial G(x^*, y)/\partial \nu(y)$ für ein x^* *auf* dem Kreisrand Γ. Parametrisieren wir $y = (\cos\tau, \sin\tau)$ und $x^* = (\cos\theta, \sin\theta)$, dann ergibt sich aus (20.6)

$$\begin{aligned}\frac{\partial}{\partial \nu(y)} G(x^*, y) \;&=\; \frac{1}{2\pi}\,\frac{\cos\theta\cos\tau - \cos^2\tau + \sin\theta\sin\tau - \sin^2\tau}{(\cos\theta - \cos\tau)^2 + (\sin\theta - \sin\tau)^2} \;=\; \frac{1}{2\pi}\,\frac{\cos(\theta-\tau) - 1}{2 - 2\cos(\theta - \tau)} \\ &=\; -\frac{1}{4\pi}\,,\end{aligned}$$

d.h., das Integral in (20.2) vereinfacht sich zu

$$\int_\Gamma \Big(1 + \frac{\partial G(x^*, y)}{\partial \nu(y)}\Big)\,\varphi(y)\, ds(y) \;=\; \int_0^{2\pi} \big(1 - \frac{1}{4\pi}\big)\,\varphi(\theta)\, d\theta\,, \tag{20.7}$$

wobei wir einfach die Funktion φ mit einer Funktion über dem θ-Intervall $[0, 2\pi)$ identifiziert haben. Man beachte, daß der Integrand von (20.7) keinerlei Singularität aufweist.

Trotz des Zusammenhangs zu (20.2) ist jedoch (20.7) nicht der Randwert $u(x^*)$ der Funktion u aus (20.2). Eine sehr technische und langwierige Rechnung ergibt nämlich die folgende **Sprungrelation** (für einen Beweis sei auf [Kress:Linear Integral Equations] verwiesen):

Lemma 20.2 *Sei φ stetig. Dann hat die Funktion u aus* (20.2) *eine stetige Fortsetzung auf den Rand Γ von Ω, und es gilt*

$$\lim_{x \to x^*} u(x) \;=\; \frac{1}{2}\,\varphi(x^*) \;+\; \int_\Gamma \Big(1 \,+\, \frac{\partial G(x^*,y)}{\partial \nu(y)}\Big)\,\varphi(y)\,ds(y)\,, \qquad x^* \in \Gamma\,.$$

Dabei ist der Grenzwert $x \to x^*$ für $x \in \mathbb{R}^2 \setminus \overline{\Omega}$ zu betrachten.

Für den Einheitskreis ergibt sich also anstelle von (20.7) der Grenzwert

$$\frac{1}{2}\,\varphi(x^*) \;+\; \int_0^{2\pi} \big(1 \,-\, \frac{1}{4\pi}\big)\,\varphi(\theta)\,d\theta\,. \tag{20.8}$$

Aus Lemma 20.1 und Lemma 20.2 folgt nun unmittelbar das folgende Resultat:

Satz 20.3 *Die Funktion u aus* (20.2) *löst das Außenraumproblem* (20.1)*, falls φ eine stetige Lösung der* ***Integralgleichung***

$$\varphi(x) \,+\, \int_\Gamma \Big(2 \,+\, 2\,\frac{\partial G(x\,,y)}{\partial \nu(y)}\Big)\,\varphi(y)\,ds(y) \;=\; 2g(x) \tag{20.9}$$

ist.

Für den Einheitskreis ist eine solche Funktion φ schnell gefunden. Wegen der speziellen Form (20.8) der linken Seite von (20.9) (wobei sich allerdings (20.8) und (20.9) um den Faktor $1/2$ unterscheiden) ergibt sich durch Integration von 0 bis 2π

$$\Big(1 + 2\pi\big(2 - 1/(2\pi)\big)\Big) \int_0^{2\pi} \varphi(\theta)\,d\theta \;=\; 2\int_0^{2\pi} g(\theta)\,d\theta\,.$$

Daher ist

$$\int_0^{2\pi} \varphi(\theta)\,d\theta \;=\; \frac{1}{2\pi}\int_0^{2\pi} g(\theta)\,d\theta$$

und eingesetzt in (20.9) folgt

$$\varphi(\theta) = 2g(\theta) - \gamma \quad \text{mit} \quad \gamma = \frac{1}{\pi}\big(1 - \frac{1}{4\pi}\big)\int_0^{2\pi} g(\theta)\,d\theta\,.$$

Für allgemeinere Gebiete Ω als den Kreis ist die Lösung der Integralgleichung allerdings schwieriger. Für den Fall einer Ellipse Ω mit Halbachsenlängen α und β wird beispielsweise (20.9) zu ($\rightsquigarrow$ Übungen)

$$\begin{aligned} \varphi(\theta) \;+\; \int_0^{2\pi} k(\tau,\theta)\varphi(\tau)\,d\tau \;&=\; 2g(\theta)\,, \\ k(\tau,\theta) \;=\; 2\sqrt{\beta^2\cos^2\tau + \alpha^2\sin^2\tau} \;&-\; \frac{\alpha\beta}{\pi}\,\frac{1}{\alpha^2+\beta^2-(\alpha^2-\beta^2)\cos(\tau+\theta)}\;. \end{aligned} \tag{20.10}$$

Wir suchen im folgenden eine Näherungslösung $\varphi_n \in V_p$ von φ aus dem Raum der periodischen linearen Splines über $[0, 2\pi)$ mit $n = 2^p$ äquidistanten Stützstellen wie in den vorigen beiden Abschnitten (die Länge des Intervalls spielt natürlich keine wesentliche Rolle).

Bezeichnet $\{\phi_0, \phi_1, \ldots, \phi_{n-1}\}$ eine beliebige Basis von V_p, dann können wir $\varphi_n \in V_p$ bezüglich dieser Basis entwickeln und erhalten

$$\varphi_n = \sum_{i=0}^{n-1} \zeta_i \phi_i\,. \tag{20.11}$$

Als Basis stehen hierfür die konventionelle nodale Basis von V_p oder eine der Waveletbasen aus den vorangegangenen Abschnitten zur Verfügung. Für eine geeignete Bestimmung der n unbekannten Koeffizienten ζ_i werden n lineare Gleichungen benötigt; beim **Galerkin-Verfahren** bildet man dazu Skalarprodukte der Gleichung (20.10) mit den Basisfunktionen ϕ_i von V_p, $i = 0, \ldots, n-1$: Das ergibt das lineare Gleichungssystem

$$Az = b\,,$$

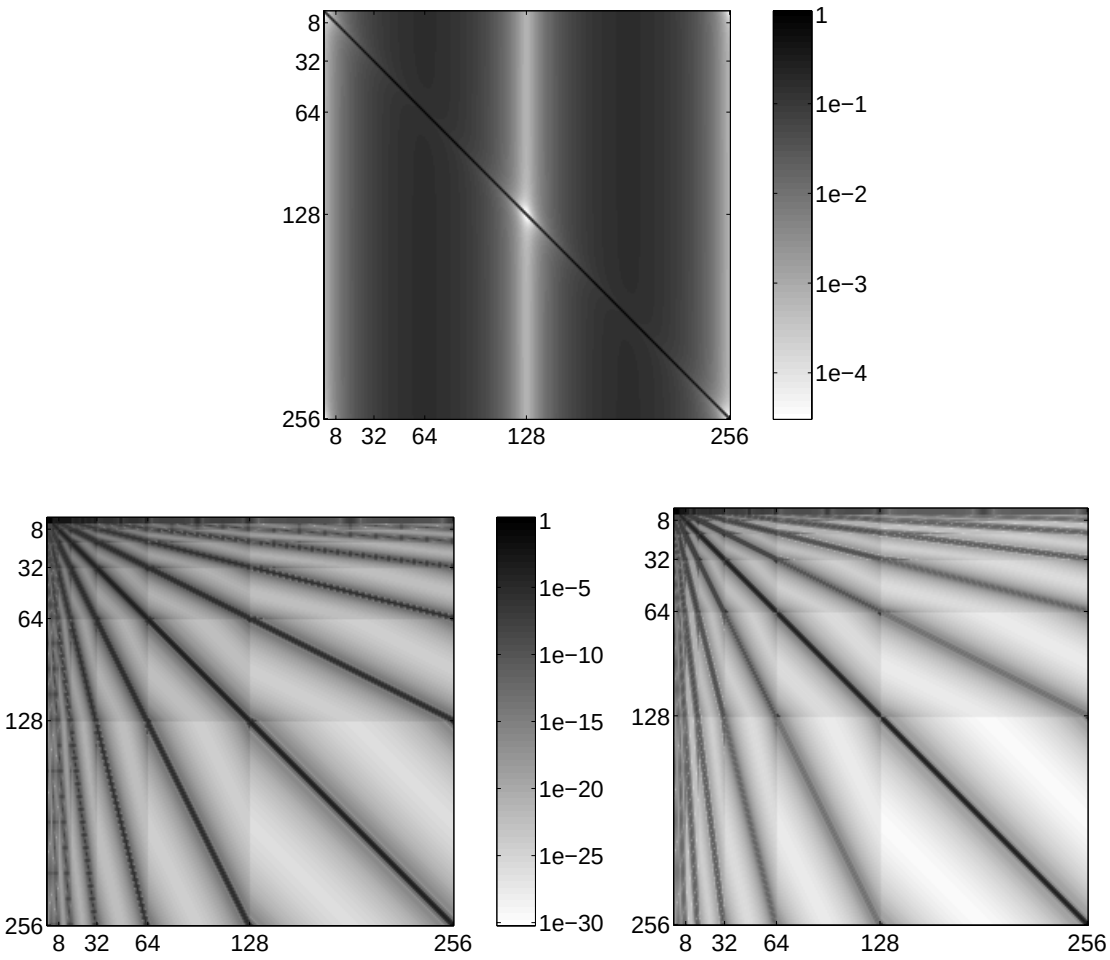
mit $z = [\zeta_0, \ldots, \zeta_{n-1}]^T$, $b = 2\,[\,\langle\,\phi_i, g\,\rangle\,]_i$ und der Matrix

$$A = \left[\,\langle\,\phi_i, \phi_j\,\rangle + \langle\,\phi_i, K\phi_j\,\rangle\,\right]_{i,j}\,; \tag{20.12}$$

hierbei bezeichnet K den Integraloperator

$$K : \phi \;\mapsto\; K\phi \;=\; \int_0^{2\pi} k(\tau,\theta)\phi(\tau)\,d\tau$$

mit k wie in (20.10) (mit ϕ ist also auch $K\phi$ eine stetige Funktion von θ).



Die obige Abbildung veranschaulicht im Fall einer Ellipse mit Halbachsen $\alpha = 10$ und $\beta = 1$ die Größenordnung der Einträge der Koeffizientenmatrix $A \in \mathbb{R}^{256\times 256}$ in einer logarithmischen Skala. Für die drei Darstellungen wurden die folgenden Basen verwendet:

(a) nodale Basis: Bild oben Mitte
(b) biorthogonales Wavelet: Bild unten links
(c) semiorthogonales Wavelet: Bild unten rechts

Die Dunkelheit der Bildpunkte entspricht der absoluten Größe der jeweiligen Matrixeinträge: je dunkler der Bildpunkt, umso größer ist der Eintrag in der Matrix. Dabei ist zu beachten, daß die obere Abbildung eine andere Graustufeneinteilung verwendet als die unteren beiden (vgl. die beiden unterschiedlichen Skalen).

Interessant sind die Strukturen in den Abbildungen für die Waveletbasen. An diesen Strukturen kann man sehr genau die Zuordnung zu den unterschiedlichen Skalen erkennen. Zunächst fallen vor allem die fingerähnlichen Linien in den Abbildungen auf. Sie kommen

hauptsächlich von den Innenprodukten $\langle \phi_i, \phi_j \rangle$ in der Matrix A, vgl. (20.12). Dies erkennt man daran, daß die Intensität der "Finger" in den beiden Abbildungen unterschiedlich ist: Für das semiorthogonale Wavelet sind die "nebendiagonalen Finger" um einige Zehnerpotenzen kleiner, da die Wavelets auf unterschiedlichen Skalen zueinander orthogonal sind.

Daß für das semiorthogonale Wavelet überhaupt Finger auftreten, kann man dadurch erklären, daß die Funktion $k(\tau, \theta)$ und ihre Ableitungen umso kleiner werden, je weiter die zu τ und θ gehörenden Ellipsenrandpunkte voneinander entfernt sind (nachrechnen!). Da in dem Ausdruck

$$\langle \phi_i, K\phi_j \rangle = \iint k(\tau, \theta)\, \phi_i(\theta)\phi_j(\tau)\, d(\tau, \theta) \tag{20.13}$$

nur diejenigen Werte $k(\tau, \theta)$ eine Rolle spielen, für die $\tau \in \operatorname{supp} \phi_j$ und $\theta \in \operatorname{supp} \phi_i$, wird das Innenprodukt also umso kleiner, je weiter die Träger von ϕ_i und ϕ_j auseinander liegen. Die Fingerlinien gehören tatsächlich zu den Indexpaaren (i, j), für die sich die Träger von ϕ_i und ϕ_j überlappen.

Auf beide Wavelets trifft darüberhinaus die folgende Beobachtung zu: Je feiner die Skala ist, umso heller sind die entsprechenden Blöcke. Dies liegt daran, daß die Kernfunktion sehr glatt ist, und daher nur eine sehr kleine Fluktuation auf den feinen Skalen aufweist.

Insgesamt kann aus der Abbildung das Fazit gezogen werden, daß in den beiden Matrizen, die zu den Waveletbasen gehören, in jeder Zeile alle bis auf etwa $O(\log n)$ Einträge ohne merklichen Genauigkeitsverlust durch Null ersetzt werden können. Dies spart Speicherplatz und verbilligt beispielsweise den Einsatz iterativer Methoden wie das Gauß-Seidel Verfahren, da Matrix-Vektor Produkte mit dünn besetzten Matrizen sehr viel billiger sind: Für eine Matrix-Vektor Multiplikation benötigt man beispielsweise nach der Kompression nur noch $O(n \log n)$ anstelle der üblichen n^2 Operationen.

IV. Eigenwerte

21 Eigenwerteinschließungen

Erinnerung. Ist $A \in \mathbb{K}^{n\times n}$ dann ist $p(\lambda) = \det(A - \lambda I)$ ein (komplexwertiges) Polynom über $\mathbb{C}$ vom Grad n. Jede der n Nullstellen von p ist ein Eigenwert von A, d.h., zu einer solchen Nullstelle λ gibt es einen Eigenvektor $0 \neq x \in \mathbb{C}^n$ mit $Ax = \lambda x$; umgekehrt ist auch jeder Eigenwert eine Nullstelle von p. Die Menge aller Eigenwerte nennt man das Spektrum $\sigma(A)$ von A.

Ist $\lambda \in \sigma(A)$ dann ist $\overline{\lambda} \in \sigma(A^*)$. Folglich gibt es einen Vektor $y \neq 0$ mit $A^*y = \overline{\lambda}y$ und es folgt

$$\lambda y^* = (\overline{\lambda}y)^* = (A^*y)^* = y^*A\,;$$

daher heißt y auch linker Eigenvektor von A zu λ. Ist $\tilde{x}$ ein Eigenvektor zu einem anderen Eigenwert $\tilde{\lambda}$ von A, dann gilt $y^*\tilde{x} = 0$ (wegen $y^*A\tilde{x} = \lambda y^*\tilde{x} = y^*(\tilde{\lambda}\tilde{x}) = \tilde{\lambda}y^*\tilde{x}$).

Eigenwerte sind selbst bei reellen Matrizen i. a. nicht reell. Ist aber $A \in \mathbb{R}^{n\times n}$ und $\lambda \in \sigma(A)$, dann ist auch $\overline{\lambda} \in \sigma(A)$, denn aus $Ax = \lambda x$ folgt

$$A\overline{x} = \overline{Ax} = \overline{\lambda x} = \overline{\lambda}\overline{x}\,.$$

Die Eigenwertgleichung $Ax = \lambda x$ ist *nichtlinear* bezüglich der gemeinsamen Unbekannten λ und x; daher sind die meisten numerischen Verfahren zur Berechnung von $\sigma(A)$ iterativ und manchmal nur lokal konvergent. Aus diesem Grund ist die folgende Sammlung relativ einfacher Ergebnisse über die Lage der Eigenwerte einer Matrix von Bedeutung.

Satz 21.1 (Satz von Gerschgorin) *Sei $A = (a_{ij}) \in \mathbb{K}^{n\times n}$, λ ein beliebiger Eigenwert von A. Dann gilt*

$$\lambda \in \bigcup_{i=1}^{n} K_i = \bigcup_{i=1}^{n} \{\zeta : |\zeta - a_{ii}| \leq \sum_{j=1, j\neq i}^{n} |a_{ij}|\}. \tag{21.1}$$

Beweis. Sei $Ax = \lambda x$, $x \neq 0$. Dann existiert x_i mit $|x_j| \leq |x_i|$ für $j \neq i$. Folglich ist

$$\lambda x_i = (Ax)_i = \sum_{j=1}^{n} a_{ij} x_j$$

und weiter

$$|\lambda - a_{ii}| = \left| \sum_{j=1, j\neq i}^{n} a_{ij} \underbrace{\frac{x_j}{x_i}}_{|.|\leq 1} \right| \leq \sum_{j=1, j\neq i}^{n} |a_{ij}| \,.$$

Also ist $\lambda \in K_i \subset \bigcup_{j=1}^{n} K_j$. □

Wegen $\overline{\lambda} \in \sigma(A^*)$ gilt entsprechend der Satz von Gerschgorin angewendet auf A^*,

$$\overline{\lambda} \;\in\; \bigcup_{i=1}^{n} \{\zeta : \; |\zeta - \overline{a}_{ii}| \leq \sum_{j=1, j\neq i}^{n} |a_{ji}|\}$$

bzw.

$$\lambda \;\in\; \bigcup_{i=1}^{n} \overline{K_i} \;:=\; \bigcup_{i=1}^{n} \{\zeta : \; |\zeta - a_{ii}| \leq \sum_{j=1, j\neq i}^{n} |a_{ji}|\}. \tag{21.2}$$

Ist A eine beliebige $n \times n$-Matrix, dann ist $(A + A^*)/2$ hermitesch, und $(A - A^*)/2$ **schiefhermitesch**, d. h.

$$\left(\frac{A - A^*}{2}\right)^* = -\frac{A - A^*}{2}.$$

Es gilt

$$A \;=\; \frac{A + A^*}{2} + \frac{A - A^*}{2}.$$

Definition 21.2 *Unter dem* ***Wertebereich*** *einer Matrix $A \in \mathbb{K}^{n\times n}$ versteht man die Menge aller* ***Rayleigh-Quotienten*** *x^*Ax/x^*x mit $x \in \mathbb{C}^n \setminus \{0\}$:*

$$\mathcal{W}(A) := \left\{\zeta = \frac{x^*Ax}{x^*x} \;:\; x \in \mathbb{C}^n \setminus \{0\}\right\} \subset \mathbb{C}.$$

Lemma 21.3

(a) $\mathcal{W}(A)$ ist zusammenhängend.

(b) Ist A hermitesch, dann ist $\mathcal{W}(A)$ das reelle Intervall $[\lambda_{min}, \lambda_{max}]$.

(c) Ist A schiefhermitesch, dann ist $\mathcal{W}(A)$ ein rein imaginäres Intervall, nämlich die konvexe Hülle seiner Eigenwerte.

Beweis. *(a)* Sei $\zeta_1 \neq \zeta_0 \in \mathcal{W}(A), \zeta_1 = x_1^* A x_1 / x_1^* x_1$, $\zeta_0 = x_0^* A x_0 / x_0^* x_0$ mit $x_1, x_0 \in \mathbb{C}^n \setminus \{0\}$. Offensichtlich ist $x_1 \neq \lambda x_0$, da sonst $\zeta_1 = \zeta_0$. Aus diesem Grund enthält das Intervall $[x_0, x_1] := \{x_t = x_0 + t(x_1 - x_0) : t \in [0,1]\}$ nicht den Nullpunkt und daher definiert $\zeta_t := x_t^* A x_t / x_t^* x_t \in \mathcal{W}(A)$ eine stetige Kurve, die ζ_0 mit ζ_1 verbindet.

(b) Das haben wir bereits im Beweis von Satz **??** gesehen.

(c) Wegen $A^* = -A$ ist iA hermitesch: $(iA)^* = \overline{i} A^* = -iA^* = iA$. Da $\mathcal{W}(iA) = i\,\mathcal{W}(A)$ und $\sigma(iA) = i\,\sigma(A)$ ist, folgt die Behauptung daher aus Teil *(b)*.

□

Offensichtlich gilt immer

$$\sigma(A) \subset \mathcal{W}(A)\,. \tag{21.3}$$

Satz 21.4 (Satz von Bendixson) *Sei $A \in \mathbb{K}^{n\times n}$ beliebig. Dann liegt das Spektrum von A in dem Rechteck*

$$\sigma(A) \;\subset\; R \;:=\; \mathcal{W}\left(\frac{A+A^*}{2}\right) + \mathcal{W}\left(\frac{A-A^*}{2}\right). \tag{21.4}$$

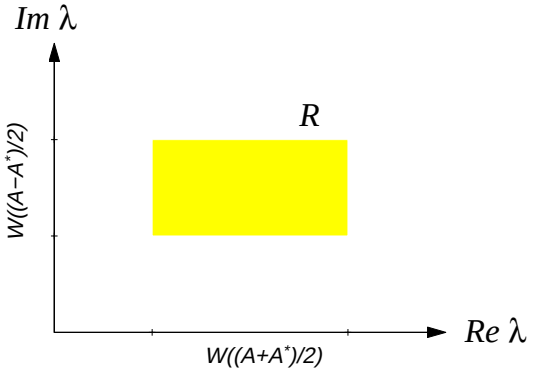


Fig. 21.1. Satz von Bendixson.

Beweis. Wir zeigen die stärkere Aussage, daß $\mathcal{W}(A) \subset \mathcal{W}\left(\frac{A+A^*}{2}\right) + \mathcal{W}\left(\frac{A-A^*}{2}\right)$.

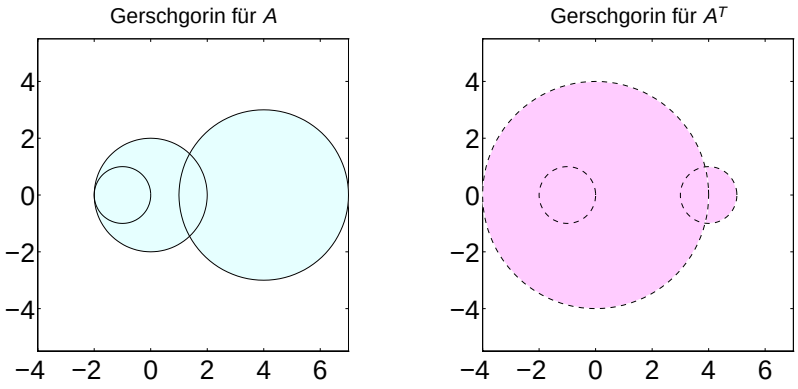
Sei $x \in \mathbb{C}^n \setminus \{0\}$: Dann gilt

$$\frac{x^*Ax}{x^*x} = \frac{x^*\left[\frac{1}{2}(A+A^*)+\frac{1}{2}(A-A^*)\right]x}{x^*x}$$
$$= \frac{x^*\frac{1}{2}(A+A^*)x}{x^*x} + \frac{x^*\frac{1}{2}(A-A^*)x}{x^*x} \quad \in \quad \mathcal{W}\left(\frac{A+A^*}{2}\right) + \mathcal{W}\left(\frac{A-A^*}{2}\right).$$

□

Beispiel.

$$A = \begin{bmatrix} 4 & 0 & -3 \\ 0 & -1 & 1 \\ -1 & 1 & 0 \end{bmatrix}$$



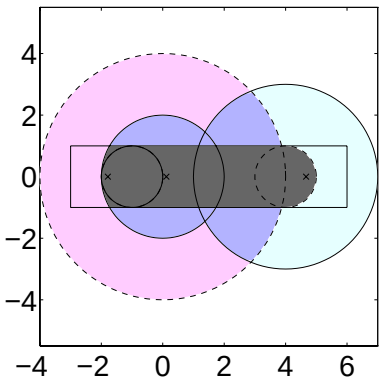
Der symmetrische und schiefsymmetrische Anteil von A ist

$$H = \frac{A+A^T}{2} = \begin{bmatrix} 4 & 0 & 2 \\ 0 & -1 & 1 \\ 2 & 1 & 0 \end{bmatrix}, \qquad S = \frac{A-A^T}{2} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

Zur Anwendung des Satzes von Bendixson schließen wir die Spektren von H und S wieder mit Hilfe des Satzes von Gerschgorin ein: Somit ergibt sich das Rechteck

$$R \;=\; [-3,6] + [-i,i]\,.$$

Das Spektrum von A muß im Schnitt *aller* drei Einschlußmengen liegen:



Tatsächlich ist das Spektrum $\sigma(A) = \{-1.7878, 0.1198, 4.6679\}$.

22 Kondition des Eigenwertproblems

Wir betrachten die Matrix

$$A = \begin{bmatrix} 0 & \dots & \dots & \dots & 0 & -a_0 \\ 1 & 0 & & & \vdots & -a_1 \\ & 1 & 0 & & \vdots & -a_2 \\ & & \ddots & \ddots & \vdots & \vdots \\ & 0 & & 1 & 0 & -a_{n-2} \\ & & & & 1 & -a_{n-1} \end{bmatrix} . \tag{22.1}$$

Entwickelt man die Determinante von $A - \lambda I$ nach der letzten Spalte, dann ergibt sich

$$(-1)^n \det(A - \lambda I) \;=\; \lambda^n + a_{n-1}\lambda^{n-1} + \ldots + a_1\lambda + a_0 \;\equiv\; p(\lambda).$$

Ist umgekehrt p ein beliebig vorgegebenes Polynom vom Grad n, dann nennt man die Matrix A aus (22.1) die **Frobenius-Begleitmatrix** von p.

Das spezielle Polynom $p_0(\lambda) = (\lambda - a)^n$ hat eine n-fache Nullstelle $\hat{\lambda} = a$, während $p_\varepsilon(\lambda) = (\lambda - a)^n - \varepsilon$ (mit $\varepsilon > 0$) die Nullstellen λ_k besitzt mit

$$\lambda_k = a + \varepsilon^{1/n} e^{i2\pi k/n}, \qquad k = 0, \ldots, n-1 .$$

Obwohl sich also die entsprechenden Frobenius-Begleitmatrizen nur um ε sowohl in der ∞, 1, 2 als auch in der Frobeniusnorm-Norm unterscheiden, haben die Eigenwerte der beiden Matrizen den Abstand

$$|\lambda_k - \hat{\lambda}| = \varepsilon^{1/n} . \tag{22.2}$$

Für $a \neq 0$ ist daher

$$\frac{|\triangle\lambda|}{|\lambda|} = \frac{\varepsilon^{1/n}}{|a|} = \underbrace{\frac{\|A\|}{|a|}\frac{\varepsilon^{1/n}}{\varepsilon}}_{\to\infty \text{ für } \varepsilon\to 0} \frac{\|\triangle A\|}{\|A\|} ;$$

mit anderen Worten: Die relative Konditionszahl ist im allgemeinen ∞ !

Man kann jedoch zeigen, daß die Eigenwerte stetig von den Einträgen der Matrix abhängen und der gefundene Exponent $1/n$ in (22.2) schlimmstmöglich ist, vgl. [Axelsson, A. 12, S. 627].

Definition 22.1 *Eine Matrix A heißt **diagonalisierbar**, falls es eine Basis aus Eigenvektoren gibt. Ist $X = (x_1, \ldots, x_n)$ die Matrix aus Eigenvektoren, dann gilt*

$$A = XDX^{-1}, \quad D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n) .$$

$$\left[\textit{Probe:} \qquad Ax_i = XDX^{-1}Xe_i = XDe_i = \lambda_i Xe_i = \lambda_i x_i\right]$$

*Ist zudem X unitär, also $X^{-1} = X^*$, dann heißt A **normal**. Normale Matrizen lassen sich auch durch die Gleichung $AA^* = A^*A$ charakterisieren.*

Die normalen Matrizen enthalten die hermiteschen Matrizen als Spezialfall.

Achtung: In der Regel ist eine Matrix *nicht* diagonalisierbar; es treten "Hauptvektoren" und "Jordankästchen" auf!

Satz 22.2 (Satz von Bauer und Fike) *Sei $A = XDX^{-1}$ diagonalisierbar und $\tilde{\lambda}$ ein Eigenwert von $A + E$. Dann existiert ein Eigenwert λ von A mit*

$$|\lambda - \tilde{\lambda}| \leq \operatorname{cond}(X)\, \|E\| .$$

Hierbei bezeichnet $\|.\|$ wahlweise die 1, 2 oder ∞-Norm und $\operatorname{cond}(X)$ die entsprechende Konditionszahl von X.

Beweis. Für $\tilde{\lambda} \in \sigma(A)$ ist die Behauptung trivial. Andernfalls existiert $(\tilde{\lambda}I - A)^{-1}$ und für einen Eigenvektor x von $A + E$ zu $\tilde{\lambda}$ gilt

$$Ex = (A + E - A)x = (\tilde{\lambda}I - A)x ,$$

also

$$(\tilde{\lambda}I - A)^{-1}Ex = x .$$

Folglich ist

$$\begin{aligned} 1 &\leq \|(\tilde{\lambda}I - A)^{-1}E\| = \|X(\tilde{\lambda}I - D)^{-1}X^{-1}E\| \\ &\leq \|X\|\,\|X^{-1}\|\,\|E\|\,\|(\tilde{\lambda}I - D)^{-1}\| = \|E\|\operatorname{cond}(X) \max_{\lambda\,\in\,\sigma(A)} |\tilde{\lambda} - \lambda|^{-1}. \end{aligned}$$

□

Korollar 22.3 *Ist A normal (z.B. hermitesch) und $\tilde{\lambda}$ ein Eigenwert von $A+E$ (E nicht unbedingt normal), dann existiert $\lambda \in \sigma(A)$ mit*

$$|\lambda - \tilde{\lambda}| \leq \|E\|_2 .$$

Beweis. Im betrachteten Fall ist X eine unitäre Marix und daher $\|X\| = 1$, sowie $\left\|X^{-1}\right\| = \|X^*\| = 1$. □

Der folgende Satz ist das Analogon von Korollar 22.3 für die Frobenius-Norm:

Satz 22.4 (Satz von Wielandt-Hoffmann) *A und E seien hermitesch, $\lambda_n \leq \ldots \leq \lambda_1$ und $\tilde{\lambda}_n \leq \ldots \leq \tilde{\lambda}_1$ seien die Eigenwerte von A bzw. $A+E$. Dann gilt*

$$\sum_{i=1}^{n} (\lambda_i - \tilde{\lambda}_i)^2 \leq \|E\|_F^2 .$$

Der Beweis ist allerdings komplizierter, vgl. [Wilkinson, The Algebraic Eigenvalue Problem, S. 108ff].

Satz 22.2 gibt der Zahl $\mathrm{cond}(X)$, X die Eigenvektormatrix, die Bedeutung einer "normweisen absoluten Konditionszahl" für die einzelnen Eigenwerte von A, ähnlich wie in Definition **??** $\mathrm{cond}(A)$ die (normweise relative) Konditionszahl für das Lösen eines linearen Gleichungssystems mit der Matrix A genannt wurde.

Ohne Normen läßt sich die absolute Kondition eines einzelnen Eigenwerts wie folgt durch Differentiation bestimmen:

Sei $A(t), t \in (-\varepsilon, \varepsilon)$, eine differenzierbare Matrixfunktion mit $A(0) = A$. $\lambda(t)$ sei ein einfacher Eigenwert von $A(t)$ mit rechtem und linkem Eigenvektor $x(t)$, bzw. $y(t)$. Dann gilt

$$A(t)x(t) = \lambda(t)x(t) ,$$

und nach Differentiation (Punkte bezeichnen die Ableitung nach t),

$$\dot{A}(t)x(t) + A(x)\dot{x}(t) \;=\; \dot{\lambda}(t)x(t) + \lambda(t)\dot{x}(t) .$$

Für $t = 0$ ergibt das (mit $\dot{A} = \dot{A}(0)$, $x = x(0)$, $\lambda = \lambda(0)$, $\dot{\lambda} = \dot{\lambda}(0)$)

$$\dot{A}x + A\dot{x}(0) \;=\; \dot{\lambda}x + \lambda\dot{x}(0) , \tag{22.3}$$

und durch Multiplikation von links mit $y = y(0)$ folgt

$$y^*\dot{A}x + \underbrace{y^*A\dot{x}(0)}_{=\lambda y^*\dot{x}(0)} \;=\; \dot{\lambda}y^*x + \lambda y^*\dot{x}(0) ,$$

bzw.

$$\dot{\lambda} = \frac{y^* \dot{A} x}{y^* x} \ . \tag{22.4}$$

Dieser Ausdruck wird i.A. groß, falls $y^*x \approx 0$. Die Kondition eines Eigenwerts ist also schlecht, wenn rechter und linker Eigenvektor fast zueinander orthogonal sind.

23 Die Potenzmethode

Als erstes konstruktives Verfahren zur näherungsweisen Bestimmung einzelner Eigenwerte und Eigenvektoren betrachten wir die **Potenzmethode von von Mises**.

Um die Ideen des Verfahrens so klar wie möglich herauszustellen, beschränken wir uns grundsätzlich im folgenden auf reelle diagonalisierbare $n \times n$ Matrizen $A \neq 0$ mit n betragsmäßig verschiedenen Eigenwerten

$$|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n| \geq 0 \qquad \text{(beachte: alle } \lambda_i \in \mathbb{R}\text{)}.$$

Alle Ergebnisse können mit entsprechendem technischen Aufwand auf den allgemeinen Fall übertragen werden. Sind $v_1, \ldots, v_n$ mit $\|v_i\| = 1$ die zugehörigen Eigenvektoren von A, dann gibt es für jeden Vektor x eine Entwicklung

$$x = \sum_{i=1}^{n} \xi_i v_i, \tag{23.1}$$

und folglich ist

$$A^k x = \sum_{i=1}^{n} \lambda_i^k \xi_i v_i. \tag{23.2}$$

Das von Mises-Verfahren beruht nun auf der asymptotischen Identität

$$A^k x \approx \lambda_1^k \xi_1 v_1.$$

Algorithmus 23.1 (Von Mises Potenzmethode) Setze $z^{(0)} := x$ mit $\|x\| = 1$ und iteriere

$$\tilde{z}^{(k)} := A z^{(k-1)}, \quad z^{(k)} := \frac{\tilde{z}^{(k)}}{\left\|\tilde{z}^{(k)}\right\|}, \qquad k = 1, 2, \ldots . \tag{23.3}$$

(Die Wahl der Norm ist dabei unerheblich.)

Es gelten die folgenden Eigenschaften:

Satz 23.2 *Ist* $\xi_1 \neq 0$ *in* (23.1) *und* $q := |\lambda_2/\lambda_1| < 1$, *dann gilt für* $k \to \infty$,

(i) $\quad \|\tilde{z}^{(k)}\| = |\lambda_1| + O(q^k)\,.$

Das Vorzeichen von λ_1 ergibt sich aus

(ii)
$$\begin{aligned} \|z^{(k)} - \operatorname{sign}(\xi_1)v_1\| &= O(q^k)\,, & \lambda_1 > 0\,,\\ \|(-1)^k z^{(k)} - \operatorname{sign}(\xi_1)v_1\| &= O(q^k)\,, & \lambda_1 < 0\,. \end{aligned}$$

Beweis. Aus (23.2) folgt

$$A^k x \;=\; \lambda_1^k \xi_1 \Big[v_1 + \underbrace{\sum_{i=2}^{n} \left(\frac{\lambda_i}{\lambda_1}\right)^k \frac{\xi_i}{\xi_1} v_i}_{=:w^{(k)}} \Big]$$

mit

$$\|w^{(k)}\| \;\le\; q^k \underbrace{\sum_{i=2}^{n} \left|\frac{\xi_i}{\xi_1}\right|}_{=:C} \;=\; Cq^k\,. \tag{23.4}$$

Damit ist

$$z^{(k)} \;=\; \frac{A^k x}{\|A^k x\|} \;=\; \operatorname{sign}(\lambda_1^k \xi_1)\, \frac{v_1 + w^{(k)}}{\|v_1 + w^{(k)}\|} \;=\; \operatorname{sign}(\lambda_1^k \xi_1) v_1 + e^{(k)} \tag{23.5}$$

mit

$$e^{(k)} \;=\; \frac{\operatorname{sign}(\lambda_1^k \xi_1)}{\|v_1 + w^{(k)}\|} \big(w^{(k)} \;+\; (1 - \|v_1 + w^{(k)}\|) v_1 \big)\,.$$

Nun ist aber $\|v_1\| - \|w^{(k)}\| \le \|v_1 + w^{(k)}\| \le \|v_1\| + \|w^{(k)}\|$ und $\|v_1\| = 1$, so daß

$$\big|\, 1 - \|v_1 + w^{(k)}\| \,\big| \;\le\; \|w^{(k)}\|$$

gilt. Zudem ist wegen (23.4) $\|w^{(k)}\| \le 1/2$ für hinreichend große k, etwa $k \ge k_0$, und daher $\|v_1 + w_k\| \ge 1/2$; daraus folgt

$$\|e^{(k)}\| \;\le\; 2\big(\|w^{(k)}\| + \|w^{(k)}\|\,\|v_1\|\big) \;=\; 4\|w^{(k)}\|\,, \qquad k \ge k_0\,. \tag{23.6}$$

Also ist

$$\begin{aligned} z^{(k)} &= \operatorname{sign}(\lambda_1^k \xi_1)\, v_1 \;+\; e^{(k)}\,,\\ \tilde{z}^{(k+1)} &= \lambda_1 \operatorname{sign}(\lambda_1^k \xi_1)\, v_1 \;+\; A e^{(k)}\,, \end{aligned}$$

und wegen (23.6) folgt hieraus unmittelbar die behauptete Aussage. □

Bemerkungen.

- Die Normierung $\tilde{z}^{(k)} \mapsto z^{(k)}$ in (23.3) ist sinnvoll (bzw. notwendig) um overflow/underflow zu vermeiden.
- Aus *(i)* ergibt sich $|\lambda_1|$ und aus dem Vorzeichenverhalten von $z^{(k)}$ schließt man dann auf das Vorzeichen von λ_1: Alternieren die Vorzeichen von $z^{(k)}$, dann ist $\lambda_1 < 0$; ansonsten ist $\lambda_1 > 0$.
- Die Voraussetzung $\xi_1 \neq 0$ kann natürlich nicht a priori überprüft werden. Wegen Rundungsfehlereinflüssen wird jedoch in der Regel immer eine Komponente von $z^{(k)}$ längs v_1 im Verlauf der Iteration eingeschleppt.

Varianten: Die Potenzmethode von v. Mises kann in dieser Form nur verwendet werden, um λ_1 zu bestimmen. Zur Berechnung anderer Eigenwerte von A kann man jedoch A zunächst geeignet transformieren:

(a) Ist $\lambda_n \neq 0$ und verwendet man A^{-1} statt A in (23.3), dann spricht man von **inverser Iteration**. A^{-1} hat die Eigenwerte

$$|\lambda_n^{-1}| > |\lambda_{n-1}^{-1}| > \ldots > |\lambda_1^{-1}|$$

mit den gleichen Eigenvektoren wie A, also approximiert die inverse Iteration $|\lambda_n^{-1}|$ und den entsprechenden Eigenvektor v_n.

(b) Ist λ eine Näherung an einen Eigenwert von A, liegt aber selber nicht in $\sigma(A)$, dann ergibt (23.3) mit $(A - \lambda I)^{-1}$ statt A die **gebrochene Iteration von Wielandt**. $(A - \lambda I)^{-1}$ hat die Eigenwerte $(\lambda_i - \lambda)^{-1}$, $i = 1, \ldots, n$, und die gebrochene Iteration approximiert den Eigenvektor zu dem Eigenwert $\lambda_i \in \sigma(A)$, welcher am nächsten an λ liegt.

Für die inverse Iteration und für die gebrochene Iteration muß jeweils ein LGS pro Iterationsschritt gelöst werden, allerdings mit der gleichen Matrix. Man bildet daher geschickterweise eine LR-Zerlegung von $A - \lambda I$.

Bemerkung. Ist λ_i der gesuchte Eigenwert, dann konvergiert die gebrochene Iteration um so schneller, je näher λ an λ_i liegt, da dann der Konvergenzfaktor

$$q = \max_{j \neq i} \frac{|\lambda_j - \lambda|^{-1}}{|\lambda_i - \lambda|^{-1}} = \max_{j \neq i} \frac{|\lambda_i - \lambda|}{|\lambda_j - \lambda|}$$

entsprechend klein wird.

Ist $A = A^T$ reell symmetrisch (oder auch $A = A^*$) und $\|\cdot\|$ die *Euklidnorm*, dann kann man die Näherung $\|\tilde{z}^{(k)}\| \approx |\lambda|$ verbessern, indem man statt dessen die Näherung

$$\lambda \approx \frac{z^{(k)^*} A z^{(k)}}{\underbrace{z^{(k)^*} z^{(k)}}_{=1}} = z^{(k)^*} A z^{(k)} = z^{(k)^*} \tilde{z}^{(k+1)} \tag{23.7}$$

verwendet. Es gilt nämlich:

Korollar 23.3 *Ist A neben den Voraussetzungen in Satz 23.2 auch noch symmetrisch und ist $\|\cdot\| = \|\cdot\|_2$, dann gilt*

$$\left|\lambda_1 - z^{(k)^*} \tilde{z}^{(k+1)}\right| = O(q^{2k}), \quad k \to \infty.$$

Beweis. Es bezeichne

$$\gamma_k := \operatorname{sign}(\lambda_1^k \xi_1)\ \|v_1 + w^{(k)}\|_2^{-1}.$$

Dann ist

$$\begin{aligned} \lambda_1 - z^{(k)^*} \tilde{z}^{(k+1)} &= \lambda_1 z^{(k)^*} z^{(k)} - z^{(k)^*} \tilde{z}^{(k+1)} = z^{(k)^*} \left(\lambda_1 z^{(k)} - \tilde{z}^{(k+1)}\right) \\ &= z^{(k)^*} (\lambda_1 I - A) z^{(k)} \\ &\overset{(23.5)}{=} z^{(k)^*} (\lambda_1 I - A) \gamma_k w^{(k)}. \end{aligned}$$

Da A hermitesch ist, ergibt sich weiter

$$z^{(k)^*} (\lambda_1 I - A) \gamma_k w^{(k)} = \gamma_k w^{(k)^*} (\lambda_1 I - A) z^{(k)} \overset{(23.5)}{=} \gamma_k w^{(k)^*} (\lambda_1 I - A) \gamma_k w^{(k)},$$

so daß also

$$|\lambda_1 - z^{(k)^*} \tilde{z}^{(k+1)}| \leq \gamma_k^2 \left(|\lambda_1| + \|A\|_2\right) \|w^{(k)}\|_2^2 \leq 2 \|A\|_2 \ \|w^{(k)}\|_2^2 ;$$

letztere Abschätzung ist gültig, da für hermitesche Matrizen v_1 und $w^{(k)}$ zueinander senkrecht sind und somit nach dem Satz von Pythagoras $\left\|v_1 + w^{(k)}\right\|_2 \geq 1$, bzw. $|\gamma_k| \leq 1$ ist. Aus (23.4) folgt nun die Behauptung. □

Entsprechend kann man für "innere" Eigenwerte verfahren. Dies wird besonders effizient, wenn man die erhaltenen Näherungen unmittelbar als neuen "Shift" in der gebrochenen Iteration verwendet:

Algorithmus 23.4 (Rayleigh-Quotient Iteration)

- Bestimme Näherungseigenvektor $z^{(0)}$ mit $\|z^{(0)}\|_2 = 1$
- for $k = 1, 2, \ldots$

– $\mu_{k+1} := z^{(k)^*} A z^{(k)}$

– $\tilde{z}^{(k+1)} := (A - \mu_{k+1} I)^{-1} z^{(k)}\,, \quad z^{(k+1)} = \dfrac{\tilde{z}^{(k+1)}}{\|\tilde{z}^{(k+1)}\|_2}\,.$

Aufgrund der obigen Konvergenzdiskussion wird man vermuten, daß Algorithmus 23.4 superlinear konvergiert. Im allgemeinen ist dies auch tatsächlich der Fall, und für $A = A^*$ ist die Konvergenz dann sogar *lokal kubisch*.

Satz 23.5 *Es sei $A = A^*$ und die Vektorfolge $(z^{(k)})$ aus Algorithmus 23.4 konvergiere gegen einen Eigenvektor x von A. Dann konvergieren die Näherungen μ_k aus Algorithmus 23.4 lokal kubisch gegen den zugehörigen Eigenwert λ.*

Beweisidee. Wir verzichten hier auf einen exakten Beweis, da dieser einige subtile technische Fallunterscheidungen nötig macht (der Beweis kann in [B.N. Parlett, The Symmetric Eigenvalue Problem, Prentice-Hall], nachgelesen werden).

Um dennoch ein Gefühl für das Konvergenzverhalten zu bekommen, skizzieren wir kurz den Beweis für den wichtigsten Fall, in dem die Iterierten $z^{(k)}$ für große k im wesentlichen von zwei Eigenvektoren aufgespannt werden. Im weiteren sei $\hat{\lambda}$ der Eigenwert von A, der am nächsten an λ ist und

$$z^{(k)} = \xi_k x + \zeta_k v + y^{(k)}\,, \tag{23.8}$$

wobei x ein Eigenvektor zum Eigenwert λ ist, v mit $\|v\|_2 = 1$ einen Eigenvektor zum Eigenwert $\hat{\lambda}$ bezeichnet, und $y^{(k)}$ den Anteil von $z^{(k)}$ in den verbliebenen Eigenräumen enthält. Nach Vorausstzung konvergiert $\xi_k \to 1$, $\zeta_k \to 0$ und $\|y^{(k)}\|_2 \to 0$ für $k \to \infty$. Wir nehmen ferner an, daß $y^{(k)}$ gegenüber $\zeta_k v$ vernachlässigbar ist.

Dann gilt wegen der Orthogonalität der Eigenräume von A, daß

$$\begin{aligned} |\lambda - \mu_{k+1}| &= |\lambda - z^{(k)^*} A z^{(k)}| = |z^{(k)^*}(\lambda I - A) z^{(k)}| \\ &= |\,|\xi_k|^2\, x^*(\lambda I - A)x + |\zeta_k|^2\, v^*(\lambda I - A)v + O(\|y^{(k)}\|_2^2)\,| \\ &= |\zeta_k|^2\,|\lambda - \hat{\lambda}| + O(\|y^{(k)}\|_2^2) \;\approx\; |\zeta_k|^2\,|\lambda - \hat{\lambda}|\,. \end{aligned} \tag{23.9}$$

Aufgrund der Iterationsvorschrift und wegen (23.8) ist

$$z^{(k)} = \frac{\tilde{z}^{(k)}}{\|\tilde{z}^{(k)}\|_2} \qquad \text{mit} \qquad \tilde{z}^{(k)} = \frac{\xi_{k-1}}{\lambda - \mu_k}\, x + \frac{\zeta_{k-1}}{\hat{\lambda} - \mu_k}\, v + \tilde{y}^{(k)}$$

und daher nach dem Satz von Pythagoras $\|\tilde{z}^{(k)}\|_2 \geq |\xi_{k-1}/(\lambda - \mu_k)|$. Wegen $\zeta_k = \zeta_{k-1}/\big(\|\tilde{z}^{(k)}\|_2(\hat{\lambda} - \mu_k)\big)$ folgt daraus, daß

$$|\zeta_k| = \frac{1}{\|\tilde{z}^{(k)}\|_2}\left|\frac{\zeta_{k-1}}{\hat{\lambda} - \mu_k}\right| \leq \left|\frac{\zeta_{k-1}(\lambda - \mu_k)}{\xi_{k-1}(\hat{\lambda} - \mu_k)}\right|\,,$$

und eingesetzt in (23.9) ergibt sich

$$|\lambda - \mu_{k+1}| \quad \approx \quad |\zeta_k|^2 \, |\lambda - \hat{\lambda}| \;\leq\; |\zeta_{k-1}|^2 \, |\lambda - \hat{\lambda}| \, \frac{(\lambda - \mu_k)^2}{|\xi_{k-1}|^2 \, (\hat{\lambda} - \mu_k)^2}$$

$$\overset{(23.9)}{\approx} \; |\lambda - \mu_k| \, \frac{(\lambda - \mu_k)^2}{|\xi_{k-1}|^2 (\hat{\lambda} - \mu_k)^2} \;,$$

also wegen $\xi_k \to 1$,

$$|\lambda - \mu_{k+1}| \lesssim \frac{1}{(\hat{\lambda} - \mu_k)^2} \, |\lambda - \mu_k|^3 \,.$$

Unter den genannten Vereinfachungen ergibt sich also kubische Konvergenz. □

Algorithmus 23.4 kann aber auch bei nichtsymmetrischen Matrizen eingesetzt werden. Die Konvergenz ist dann lokal quadratisch, vgl. [G. W. Stewart, Introduction to Matrix Computations, S. 345-348].

24 Das QR-Verfahren

Wir stellen im weiteren ein iteratives Verfahren vor, das QR-Verfahren, mit dem alle Eigenwerte einer Matrix simultan berechnet werden können. Es ist das in der Praxis am häufigsten eingesetzte Verfahren. Wie wir sehen werden, verbindet es

- die globale Konvergenzeigenschaften der Potenzmethode mit der
- schnellen lokalen Konvergenz der inversen Rayleigh-Quotient Iteration.

Wir leiten im folgenden zunächst die theoretischen Eigenschaften her; die effiziente numerische Implementierung ist Gegenstand des nächsten Abschnitts.

Das Verfahren an sich ist sehr einfach: Sei $A_0 = A$ und $(\mu_k)_{k \geq 0}$ eine Folge von "Shifts". Dann berechnet man im k-ten Iterationsschritt ($k \geq 0$)

$$\begin{array}{lrcll} \text{(a)} & A_k - \mu_k I & = & Q_k R_k & (QR\text{-Zerlegung, vgl. § ??}) \\ \text{(b)} & A_{k+1} & = & R_k Q_k + \mu_k I \,. & \end{array} \tag{24.1}$$

Es gelten die folgenden Identitäten:

Lemma 24.1

(a) $A_{k+1} = Q_k^* A_k Q_k$

(b) $A_{k+1} = (Q_0 Q_1 \ldots Q_k)^* A (Q_0 Q_1 \ldots Q_k)$

(c) $\displaystyle\prod_{j=0}^{k} (A - \mu_j I) = (Q_0 Q_1 \ldots Q_k)(R_k R_{k-1} \ldots R_0)$

Beweis. *(a)* Aufgrund von (24.1) ist

$$A_{k+1} \;=\; R_k Q_k + \mu_k I \;=\; Q_k^* Q_k R_k Q_k + \mu_k Q_k^* Q_k \;=\; Q_k^* (Q_k R_k + \mu_k I) Q_k \;=\; Q_k^* A_k Q_k \,.$$

(b) folgt sofort aus *(a)* wegen $A_0 = A$.

(c) wird durch Induktion über k bewiesen: Die Aussage ist klar für $k = 0$;

<u>$k \mapsto k+1$:</u>

$$\begin{aligned} Q_{k+1} R_{k+1} &= A_{k+1} - \mu_{k+1} I \\ &= (Q_0 \ldots Q_k)^* A (Q_0 \ldots Q_k) - \mu_{k+1} (Q_0 \ldots Q_k)^* (Q_0 \ldots Q_k) \\ &= (Q_0 \ldots Q_k)^* (A - \mu_{k+1} I)(Q_0 \ldots Q_k) \,. \end{aligned}$$

Daraus folgt $Q_0 \ldots Q_k Q_{k+1} R_{k+1} = (A - \mu_{k+1} I)(Q_0 \ldots Q_k)$ und weiterhin

$$\begin{aligned} Q_0 \ldots Q_k Q_{k+1} R_{k+1} R_k \ldots R_0 &= (A - \mu_{k+1} I)(Q_0 \ldots Q_k) R_k \ldots R_0 \\ &= (A - \mu_{k+1} I) \prod_{j=0}^{k} (A - \mu_j I) \,, \end{aligned}$$

was zu beweisen war. □

Offensichtlich haben also alle A_k die gleichen Eigenwerte, da sie durch Ähnlichkeitstransformationen ineinander übergehen. Die Hoffnung besteht nun darin, daß die Iterierten A_k des QR-Verfahrens (24.1) im Verlauf der Iteration gegen eine obere Dreiecksmatrix konvergieren, so daß man dann die Eigenwerte von A_k (und damit die Eigenwerte von A) von der Diagonalen ablesen kann.

Im folgenden wollen wir versuchen, Zusammenhänge zwischen dem QR-Verfahren und der Potenzmethode sowie der gebrochenen Iteration herzustellen:

(i) Wir betrachten der Einfachheit halber den Fall ohne Shifts (d.h. $\mu_k = 0$ für alle k). Nach Lemma 24.1*(c)* ist dann

$$A^{k+1} = (Q_0 \ldots Q_k)(R_k \ldots R_0) \equiv \tilde{Q}_k \tilde{R}_k \tag{24.2}$$

eine QR-Zerlegung von A^{k+1}. Vergleicht man speziell die erste Spalte dieser Matrixgleichung, dann ergibt sich

$$A^{k+1} e_1 = \tilde{Q}_k \tilde{r}_{11}^{(k)} e_1 = \tilde{r}_{11}^{(k)} \tilde{q}^{(k)} \,,$$

wobei $\tilde{r}_{11}^{(k)}$ das $(1,1)$-Element von $\tilde{R}_k$ und $\tilde{q}^{(k)}$ die erste Spalte von $\tilde{Q}_k$ ist. Die Ergebnisse aus Abschnitt 23 besagen daher, daß $\tilde{q}^{(k)}$ eine gute Näherung an den Eigenvektor zum dominanten Eigenwert λ_1 von A ist. Wegen $\|e_1\|_2 = \|\tilde{q}^{(k)}\|_2 = 1$ ist $|\tilde{r}_{11}^{(k)}| \approx |\lambda_1|^{k+1}$.

Nach Lemma 24.1*(b)* ist $A_{k+1} = \tilde{Q}_k^* A \tilde{Q}_k$ und daher

$$A_{k+1} e_1 = \tilde{Q}_k^* A \tilde{q}^{(k)} \approx \lambda_1 \tilde{Q}_k^* \tilde{q}^{(k)} = \lambda_1 e_1,$$

d.h.

$$A_{k+1} \approx \left[\begin{array}{c|c} \lambda_1 & * \\ 0 & * \\ \vdots & * \\ 0 & * \end{array}\right].$$

(ii) Aus (24.2) folgt wegen der Orthogonalität von $\tilde{Q}_k$ die Gleichung $\tilde{Q}_k^* = \tilde{R}_k A^{-(k+1)}$ und Multiplikation mit e_n^* von links ergibt

$$\hat{q}^{(k)*} := e_n^* \tilde{Q}_k^* = e_n^* \tilde{R}_k A^{-(k+1)} = \tilde{r}_{nn}^{(k)} e_n^* A^{-(k+1)}.$$

Der Vektor $\hat{q}^{(k)}$ – die letzte Spalte von $\tilde{Q}_k$ – ist also nichts anderes wie das Resultat von $k+1$ Schritten der inversen Iteration mit A^* und somit eine Näherung für einen linken Eigenvektor zu dem betragskleinsten Eigenwert λ_n von A.

Aus Lemma 24.1*(b)* folgt daher

$$e_n^* A_{k+1} = e_n^* \tilde{Q}_k^* A \tilde{Q}_k = \hat{q}^{(k)*} A \tilde{Q}_k \approx \lambda_n \hat{q}^{(k)*} \tilde{Q}_k = \lambda_n e_n^*.$$

Somit ist die letzte Zeile von A_{k+1} näherungsweise ein Vielfaches von e_n^* und zusammen mit der Beobachtung (ii) ergibt sich für A_{k+1} näherungsweise die Gestalt

$$A_{k+1} \approx \left[\begin{array}{ccc|c} \lambda_1 & & & * \\ 0 & & & * \\ \vdots & & & * \\ \hline 0 & \dots & 0 & \lambda_n \end{array}\right].$$

Dieses Ergebnis soll als Motivation ausreichen, daß die Iterierten A_k von (24.1) gegen eine obere Dreiecksmatrix konvergieren.

(iii) Stellen wir uns nun vor, wir wollten die Konvergenz der in (ii) beobachteten inversen Iteration zum kleinsten Eigenwert λ_n von A (und damit von A_k) beschleunigen. Als linken Näherungs-Eigenvektor für A_k haben wir den n-ten Einheitsvektor identifiziert. Wegen der lokal schnellen Konvergenz der Rayleigh-Quotient Iteration ist es nun naheliegend, Algorithmus 23.4 anzuwenden, also den Rayleigh-Quotienten $\mu_k := e_n^* A_k e_n$ zu bilden (das ist gerade das rechte untere Eckelement von A_k) und dann einen Schritt der inversen Iteration bezüglich des linken Eigenvektors auszuführen: Wegen (24.1a) ergibt das

$$z := e_n^*(A_k - \mu_k I)^{-1} = e_n^* R_k^{-1} Q_k^* = \frac{1}{r_{nn}^{(k)}} e_n^* Q_k^* = \frac{1}{r_{nn}^{(k)}} q^{(k)*} ,$$

wobei $r_{nn}^{(k)}$ das rechte untere Eckelement von R_k und $q^{(k)}$ die hinterste Spalte von Q_k bezeichnen.

Mit anderen Worten: Ein Schritt der inversen Rayleigh-Quotienten Iteration ergibt gerade die hinterste Spalte von Q_k als neue Näherung an den linken Eigenvektor von A_k zu λ_n; darüberhinaus sieht man mit Hilfe von Lemma 24.1*(a)* sofort, daß das nächste rechte untere Eckelement von A_{k+1} gerade der zugehörige Rayleigh-Quotient ist:

$$\mu_{k+1} = q^{(k)*} A_k q^{(k)} = e_n^* Q_k^* A_k Q_k e_n = e_n^* A_{k+1} e_n .$$

Wählt man also als Shift μ_k in (24.1) jeweils das (n,n)-Element von A_k, dann darf man sehr schnelle (quadratische oder gar kubische) Konvergenz dieser Eckelemente gegen den kleinsten Eigenwert λ_n von A erwarten.

Shifts in (24.1) dienen also der Konvergenzbeschleunigung des QR-Verfahrens.

Soviel zur Motivation des QR-Verfahrens. Der Vollständigkeit halber beweisen wir die lineare Konvergenz des Verfahrens ohne Shifts für den einfachsten und wichtigsten Spezialfall:

$A = X\Lambda X^{-1} \in \mathbb{R}^{n\times n}$ sei diagonalisierbar und Λ sei die entsprechende Diagonalmatrix mit den absteigend sortierten Eigenwerten von A: $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n| > 0$. Ferner sei S die Diagonalmatrix

$$S = \begin{bmatrix} \operatorname{sign}\lambda_1 & & 0 \\ & \ddots & \\ 0 & & \operatorname{sign}\lambda_n \end{bmatrix} .$$

Ist $|\Lambda|$ die Betragsmatrix von Λ (Beträge komponentenweise genommen), dann gilt also $\Lambda = S|\Lambda|$. Für gegebenes $R \in \mathbb{K}^{n\times n}$ bezeichnet im folgenden $\operatorname{diag} R \in \mathbb{K}^n$ immer einen Vektor, der die Diagonaleinträge von R enthält, während $D_R \in \mathbb{K}^{n\times n}$ der Diagonalanteil von R ist, d.h., D_R ist eine Diagonalmatrix mit $\operatorname{diag} R = \operatorname{diag} D_R$.

Satz 24.2 *$A \in \mathbb{R}^{n\times n}$ erfülle die oben genannten Voraussetzungen und zusätzlich existiere eine LR-Zerlegung von X^{-1}. Dann gilt:*

$$Q_k \to S, \quad D_{R_k} \to |\Lambda|, \quad D_{A_k} \to \Lambda, \qquad k \to \infty .$$

Wir benötigen zunächst ein Hilfsresultat. Dazu erinnern wir uns an die Bemerkung am Schluß von Abschnitt **??**, daß zu jeder nichtsingulären Matrix eine eindeutige QR-Zerlegung mit positiven Diagonalelementen in R existiert.

Lemma 24.3 *Sei $(B_k) \subset \mathbb{R}^{n\times n}$ eine Folge von $n \times n$-Matrizen mit QR-Zerlegung $B_k = Q_k R_k$, $\operatorname{diag} R_k > 0$. Ferner konvergiere B_k gegen B für $k \to \infty$. Ist B nichtsingulär, dann gilt $Q_k \to Q$ und $R_k \to R$ für $k \to \infty$, und $QR = B$ ist die QR-Zerlegung von B mit $\operatorname{diag} R > 0$.*

Beweis. Wegen $\|B_k\|_2 = \|Q_k R_k\|_2 = \|R_k\|_2$ sind (Q_k), (R_k) beschränkte Folgen mit konvergenten Teilfolgen

$$Q_{k_n} \longrightarrow Q, \quad R_{k_n} \longrightarrow R, \qquad n \to \infty.$$

Die Grenzmatrizen erben die jeweiligen Eigenschaften, d. h. Q ist eine orthogonale Matrix und R ist eine obere Dreiecksmatrix mit nichtnegativen Diagonalelementen. Wegen

$$B \longleftarrow B_{k_n} = Q_{k_n} R_{k_n} \longrightarrow QR, \qquad n \to \infty,$$

ist $B = QR$, d. h. R ist nichtsingulär und hat damit sogar positive Diagonalelemente. QR ist dadurch als QR-Zerlegung von B eindeutig festgelegt. Also konvergieren alle konvergenten Teilfolgen von (Q_k) gegen Q und alle konvergenten Teilfolgen von (R_k) gegen R, d. h. die beiden gesamten Folgen konvergieren:

$$Q_k \longrightarrow Q, \quad R_k \longrightarrow R, \qquad k \to \infty.$$

□

Nun zum Beweis von Satz 24.2:

Beweis. Da $\mu_k = 0$ angenommen wird, ergibt sich aus Lemma 24.1*(c)*:

$$A^k = (Q_0 \ldots Q_{k-1})(R_{k-1} \ldots R_0). \tag{24.3}$$

Andererseits ist

$$A^k = (X \Lambda X^{-1})^k = X \Lambda^k X^{-1}.$$

Wir ersetzen $X = QR$ durch seine QR-Zerlegung (mit $\operatorname{diag} R > 0$) und $X^{-1} = LU$ durch die vorausgesetzte LR-Zerlegung; dabei wählen wir $\operatorname{diag} U > 0$ und L so, daß alle Diagonaleinträge von L jeweils ± 1 sind. Es folgt

$$A^k = QR\Lambda^k LU = QR\Lambda^k L\Lambda^{-k}\Lambda^k U\,.$$

Schließlich fügen wir noch die QR-Zerlegung von $R\Lambda^k L\Lambda^{-k} = \hat{Q}_k \hat{R}_k$ ein (wieder mit $\operatorname{diag} \hat{R}_k > 0$) und erhalten

$$A^k = Q\hat{Q}_k \hat{R}_k \Lambda^k U = \underbrace{(Q\hat{Q}_k S^k)}_{\text{orthogonal}} \underbrace{(S^k \hat{R}_k S^k}_{\text{diag}(\cdot)>0} |\Lambda|^k U). \tag{24.4}$$

Aufgrund der Eindeutigkeit der QR-Zerlegung ergibt sich durch Vergleich von (24.3) und (24.4):

$$Q_0 \dots Q_{k-1} = Q\hat{Q}_k S^k, \quad R_{k-1} \dots R_0 = S^k \hat{R}_k \Lambda^k U. \tag{24.5}$$

Für $k \to \infty$ gilt nun mit $L = (l_{ij})$:

$$\Lambda^k L \Lambda^{-k} = (\lambda_i^k l_{ij} \lambda_j^{-k})_{ij} = \begin{cases} 0 & i < j \\ l_{ii} & i = j \\ O(q^k) & i > j \quad \text{mit } 0 < q < 1\,, \end{cases}$$

also

$$\Lambda^k L \Lambda^{-k} = D_L + E_k \quad \text{mit} \quad \|E_k\|_2 \le C q^k$$

für ein $C \ge 0$. Folglich gilt für $k \to \infty$:

$$\hat{Q}_k \hat{R}_k = R(D_L + E_k) \longrightarrow RD_L = D_L \underbrace{D_L R D_L}_{\text{diag}(\cdot)>0}\,.$$

Da D_L aufgrund der Konstruktion eine orthogonale Matrix ist, folgt aus Lemma 24.3, daß

$$\hat{Q}_k \longrightarrow D_L, \quad \hat{R}_k \longrightarrow D_L R D_L\,, \qquad k \to \infty\,. \tag{24.6}$$

Wegen (24.5) ist

$$Q_k = (Q_0 \dots Q_{k-1})^{-1}(Q_0 \dots Q_k) = S^k \hat{Q}_k^* Q^* Q \hat{Q}_{k+1} S^{k+1},$$

und (24.6) ergibt somit

$$S^k Q_k S^{k+1} = \hat{Q}_k^* Q^* Q \hat{Q}_{k+1} = \hat{Q}_k^* \hat{Q}_{k+1} \longrightarrow D_L^2 = I\,,$$

bzw.

$$Q_k \longrightarrow S\,, \qquad k \to \infty\,.$$

Außerdem folgt aus (24.5) und (24.6), daß

$$\begin{aligned} S^{k+1} R_k S^k &= S^{k+1}(R_k \dots R_0)(R_{k-1} \dots R_0)^{-1} S^k = \hat{R}_{k+1} \Lambda^{k+1} U U^{-1} \Lambda^{-k} \hat{R}_k^{-1} \\ &= \hat{R}_{k+1} \Lambda \hat{R}_k^{-1} \longrightarrow D_L R \Lambda R^{-1} D_L\,, \end{aligned}$$

d.h. die Diagonalelemente $r_{ii}^{(k)}$ von R_k streben gegen $\operatorname{sign}(\lambda_i)\lambda_i = |\lambda_i|$ für $k \to \infty$.

Wegen Lemma 24.1*(b)* ist $\|A_k\|_2 = \|A\|_2$ für alle k. Folglich ist (A_k) eine beschränkte Folge und zerfällt in lauter konvergente Teilfolgen für $k \to \infty$. Bezeichne A_∞ einen Häufungspunkt dieser Folge: Dann gilt nach (24.1a) für eine geeignete Teilfolge

$$R_{k_n} = Q_{k_n}^* A_{k_n} \longrightarrow S A_\infty =: R_\infty\,, \qquad n \to \infty\,,$$

und wegen der schon bewiesen Konvergenz von $\operatorname{diag} R_{k_n}$ ist $\operatorname{diag} R_\infty = |\Lambda|$. Andererseits ist nun aber $A_\infty = S R_\infty$ und daher folgt

$$\operatorname{diag} A_\infty = S \operatorname{diag} R_\infty = S|\Lambda| = \Lambda\,.$$

Da dies für jeden Häufungspunkt A_∞ von (A_k) gilt, streben also die Diagonalelemente von A_k gegen λ_i für $k \to \infty$, was noch zu zeigen war. □

25 Implementierung des QR-Verfahrens

A. Reduktion auf Hessenbergform

Für beliebige Matrizen $A \in \mathbb{K}^{n\times n}$ wäre das QR-Verfahren viel zu aufwendig ($O(n^3)$ Operationen pro Iteration). Statt dessen transformiert man A zunächst auf obere Hessenbergform:

Definition 25.1 *Eine Matrix $H = (h_{ij})$ hat* ***obere Hessenbergform****, wenn $h_{ij} = 0$ für $j < i - 1$, d. h.*

$$H = \begin{bmatrix} h_{11} & h_{12} & \dots & \dots & h_{1n} \\ h_{21} & h_{22} & & & h_{2n} \\ 0 & h_{32} & h_{33} & & h_{3n} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & h_{n,n-1} & h_{nn} \end{bmatrix}.$$

Ziel ist es also zunächst, A durch Ähnlichkeitstransformationen auf obere Hessenbergform zu bringen. Dazu gehen wir analog zu Abschnitt **??** bei der QR-Zerlegung vor: Wir wählen Householder-Matrizen $P_1, \dots, P_{n-2}$ und transformieren

$$A \;\mapsto\; A_0 = P^*AP = P_{n-2}^* \dots P_1^* A P_1 \dots P_{n-2}$$

derart, daß ("$*$" bezeichnet neu erzeugte Einträge der Matrix, "$\times$"-Einträge bleiben fest)

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{bmatrix} \xrightarrow[P_1^* \cdot]{} \begin{bmatrix} \times & \times & \times & \times & \times \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \end{bmatrix} \xrightarrow[\cdot P_1]{} \begin{bmatrix} \times & * & * & * & * \\ \times & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \end{bmatrix}$$

$$\xrightarrow[P_2^* \cdot]{} \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & * & * & * \end{bmatrix} \xrightarrow[\cdot P_2]{} \begin{bmatrix} \times & \times & * & * & * \\ \times & \times & * & * & * \\ 0 & \times & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & * & * & * \end{bmatrix} \xrightarrow[P_3^* \cdot]{} \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{bmatrix}$$

$$\xrightarrow[\cdot P_3]{} \begin{bmatrix} \times & \times & \times & * & * \\ \times & \times & \times & * & * \\ 0 & \times & \times & * & * \\ 0 & 0 & \times & * & * \\ 0 & 0 & 0 & * & * \end{bmatrix}.$$

Durch Vergleich mit der Aufwandsabschätzung aus Abschnitt **??** macht man sich nun leicht klar, daß für jeden "$*$" in der obigen Umformung etwa 2 Multiplikationen benötigt werden; daher ist der Aufwand für diese Transformation in Hessenbergform ungefähr

$$\sum_{k=2}^{n-1} 2(k^2 + nk) \approx \frac{2}{3}n^3 + n^3 = \frac{5}{3}n^3 \quad \text{Multiplikationen}.$$

Die Householder-Matrizen wählt man genau in der selben Weise wie in Abschnitt **??**.

B. QR-Zerlegung einer Hessenberg-Matrix
Für Hessenberg-Matrizen läßt sich die QR-Zerlegung besonders effizient mit sogenannten **Givens-Rotationen** berechnen.

Definition 25.2 *Eine Matrix $G = G(i,j,\theta)$ mit*

$$G(i,j,\theta) = \begin{bmatrix} 1 & & & & & & & & 0 \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ & & & c & & & s & & \leftarrow i \\ & & & & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & 1 & & \\ & & & -s & & & c & & \leftarrow j \\ & & & & & & & 1 & \\ & & & & & & & & \ddots \\ 0 & & & & & & & & 1 \end{bmatrix} \qquad \begin{array}{l} c = \cos(\theta) \\ s = \sin(\theta) \end{array}$$

heißt ***Givens-Rotation.***

Givens-Rotationen sind orthogonale Matrizen (alle Spalten sind paarweise zueinander senkrecht und haben Euklidnorm 1) und die Operation GA ersetzt die Zeilen i und j von A (a_i^* bzw. a_j^*) durch Linearkombinationen $ca_i^* + sa_j^*$ bzw. $-sa_i^* + ca_j^*$, während AG die Spalten i und j von A durch entsprechende Linearkombinationen ersetzt.

Man kann daher eine Givens-Rotation so wählen, daß ein beliebiges Element von A zu 0 transformiert wird, etwa a_{jk}:

$$-sa_{ik} + ca_{jk} \stackrel{!}{=} 0 \quad \Longrightarrow \quad c = \frac{a_{ik}}{\sqrt{a_{ik}^2 + a_{jk}^2}}\,, \; s = \frac{a_{jk}}{\sqrt{a_{ik}^2 + a_{jk}^2}}\,.$$

Um bei der Berechnung von s und c overflow zu vermeiden, geht man in der Regel folgendermaßen vor: Ist $|a_{ik}| > |a_{jk}|$, dann berechnet man $t = a_{jk}/a_{ik}$ und setzt

$$c = \frac{1}{\sqrt{1+t^2}}\,, \quad s = \frac{t}{\sqrt{1+t^2}}\,.$$

Wir wenden nun sukzessive Givens-Rotationen $G(i, i+1, \theta_i)$, $i = 1, \ldots, n-1$, an, um jeweils das $(i+1, i)$-Element zu Null zu machen:

$$R = G(n-1, n, \theta_{n-1}) \ldots G(1,2,\theta_1) A\,.$$

Schema:

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{bmatrix} \underset{i=1}{\longrightarrow} \begin{bmatrix} * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{bmatrix} \underset{i=2}{\longrightarrow} \begin{bmatrix} \times & \times & \times & \times & \times \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{bmatrix}$$

$$\underset{i=3}{\longrightarrow} \begin{bmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & \times & \times \end{bmatrix} \underset{i=4}{\longrightarrow} \begin{bmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 0 & * \end{bmatrix}$$

Im Teilschritt (24.1b) muß das Produkt der Givens-Rotationen von rechts an R_k heranmultipliziert werden:

$$Q_k = \big(G(n-1,n,\theta_{n-1}) \ldots G(1,2,\theta_1)\big)^* \\ \Longrightarrow \quad A_{k+1} = R_k G(1,2,\theta_1)^* \ldots G(n-1,n,\theta_{n-1})^* .$$

Eine Multiplikation von rechts mit $G(i,i+1)$ kombiniert die Spalten i und $i+1$; daher hat A_{k+1} wieder Hessenberg-Gestalt.

Aufwand: (Multiplikationen der $2(n-1)$ Givens-Rotationen)

$$2\left(\sum_{i=1}^{n-1} 4(n-i+1)\right) = 2\left(\sum_{i=2}^{n} 4i\right) \sim 4n^2.$$

Zusammenfassung:

Algorithmus 25.3 (QR-Algorithmus)

Schritt 1: Transformiere A in obere Hessenbergform $A_0 = P^*AP$ durch Householdertransformationen. Aufwand $\approx \frac{5}{3}n^3$ Multiplikationen.

Schritt 2: Eigentliches QR-Verfahren gemäß (24.1): Die QR-Zerlegung von A_k erfolgt durch $(n-1)$ Givens-Rotationen, so daß die Hessenbergform von A_k in A_{k+1} erhalten bleibt. Aufwand $\approx 4n^2$ Multiplikationen je Iteration.

Wegen der quadratischen Konvergenz der Eigenwerte kann man mit höchstens $O(n)$ Iterationen rechnen. Der Gesamtaufwand ist daher $O(n^3)$.

C. Bestimmung der Shifts und "Deflation"

Wie wir in Abschnitt 24 gesehen haben, bietet sich das (n,n)-Element von A_k für den Shift μ_k an, um die Konvergenz zu beschleunigen. Als noch erfolgreicher erweist sich eine andere Wahl, bei der man μ_k aus dem rechten unteren (2×2)-Block von A_k wie folgt bestimmt:

$$\mu_k \text{ sei der Eigenwert } \lambda \text{ von } \begin{bmatrix} a^{(k)}_{n-1,n-1} & a^{(k)}_{n-1,n} \\ a^{(k)}_{n,n-1} & a^{(k)}_{n,n} \end{bmatrix}, \text{ der am nächsten an } a^{(k)}_{n,n} \text{ liegt.} \quad (25.1)$$

In beiden Fällen konvergiert das (n,n)-Element von A_k sehr schnell gegen den exakten Eigenwert, und das $(n,n-1)$-Element von A_k gegen 0, also

$$A_k \quad \longrightarrow \quad \begin{bmatrix} * & \cdots & \cdots & \cdots & * \\ * & \ddots & & & \vdots \\ & \ddots & \ddots & & \vdots \\ & & * & * & * \\ 0 & & & 0 & \lambda_n \end{bmatrix} = \begin{bmatrix} B_{k-1} & * \\ 0 & \lambda_n \end{bmatrix}.$$

Wir können dann das kleinere Teilproblem mit der Hessenberg-Matrix B_{k-1} weiter betrachten ("Deflation"). Das Problem zerfällt auch ansonsten gelegentlich in Teilprobleme, wenn $a_{i,i-1}^{(k)}$ – also ein Nebendiagonalelement – numerisch Null wird.

D. Komplexe Eigenwerte
Wir haben bislang vorausgesetzt, daß $A \in \mathbb{R}^{n \times n}$ und alle Eigenwerte von A reell sind. Liegen konjugiert komplexe Eigenwerte vor, dann entwickeln sich rechts unten in A_k (2×2)-Blöcke mit diesen beiden Eigenwerten, wie bei der reellen Jordan-Normalform. Eigentlich müßte man dann "komplex shiften", aber komplexe Shifts lassen sich umgehen, indem man zwei reelle Schritte geeignet zusammenfaßt (vgl. Stoer/Bulirsch).

E. Symmetrische Matrizen
Ist $A = A^T$, dann ist die Konvergenz lokal kubisch (Satz 23.5). Zudem bleiben nach Lemma 24.1 *(a)* alle A_k symmetrisch. A_k ist also eine symmetrische Hessenberg-Matrix und damit eine Tridiagonalmatrix. Dadurch werden die einzelnen Iterationsschritte noch billiger; sie erfordern nur noch $O(n)$ Opertionen. Die Grenzmatrix R von Satz 24.2 ist dann übrigens eine Diagonalmatrix.

F. Bestimmung von Eigenvektoren
Prinzipiell gibt es zwei Möglichkeiten zur Berechnung der zugehörigen Eigenvektoren:

(a) Akkumuliert man alle orthogonalen Transformationen, dann ist $R = Q^*AQ$ und es gilt, Kernvektoren von $R - r_{ii}I$ zu bestimmen. Man kann beispielsweise das LGS $(R - r_{ii}I)x = 0$ durch Rücksubstitution lösen, indem man $x_i = 1$ und $x_j = 0$, $j > i$, initialisiert. Dann ergibt $v = Qx$ einen Eigenvektor von A zu $\lambda = r_{ii}$. Diese Vorgehensweise ist allerdings wegen der expliziten Berechnung von Q sehr teuer.

(b) Alternativ kann man die inverse (bzw. gebrochene) Iteration mit der Hessenberg-Matrix A_0 zur Berechnung von Eigenvektoren verwenden. Dies ergibt in der Regel nach nur wenigen Schritten sehr gute Näherungen x an Eigenvektoren von $A_0 = P^*AP$. Die gesuchten Eigenvektoren von A erhält man dann als $v = Px$. Die inverse Iteration ist unter Verwendung der Givens-QR-Zerlegung von A_0 sehr effizient implementierbar.

Beispiel. Das *QR–Verfahren* wird auf die symmetrische Tridiagonalmatrix

$$A_0 = \begin{pmatrix} 12 & 1 & & & 0 \\ 1 & 9 & 1 & & \\ & 1 & 6 & 1 & \\ & & 1 & 3 & 1 \\ 0 & & & 1 & 0 \end{pmatrix} \qquad (n = 5)$$

angewendet. Die Shifts μ_k werden dabei wie in (25.1) als jeweiliger Eigenwert λ des 2×2 Eckblocks $\begin{bmatrix} a^{(k)}_{n-1,n-1} & a^{(k)}_{n-1,n} \\ a^{(k)}_{n,n-1} & a^{(k)}_{n,n} \end{bmatrix}$ von A_k gewählt, für den $|a^{(k)}_{nn} - \lambda|$ am kleinsten ist. Nachfolgend werden für jedes k die Elemente $a^{(k)}_{n,n-1}, a^{(k)}_{n,n}$, sowie die Shifts μ_k angegeben:

k	$a^{(k)}_{5,4}$	$a^{(k)}_{5,5}$	μ_k
1	1	0	-0.302775637732
2	$-0.454544295102_{10^{-2}}$	-0.316869782391	-0.316875874226
3	$0.106774452090_{10^{-9}}$	-0.316875952616	-0.316875952619
4	$0.918983519419_{10^{-22}}$	-0.316875952617	

Nach vier Schritten wird also $\lambda_5 = -0.316875\ldots$ erkannt.

Weiterbehandlung der (4×4)–Matrix ergibt dann nach weiteren drei Schritten $\lambda_4 = 2.98386\ldots$:

k	$a^{(k)}_{4,3}$	$a^{(k)}_{4,4}$	μ_k
4	0.143723850633	2.99069135875	2.98389967722
5	$-0.171156231712_{10^{-5}}$	2.98386369683	2.98386369682
6	$-0.111277687663_{10^{-17}}$	2.98386369682	

Weiterbehandlung der (3×3)–Matrix ergibt schließlich $\lambda_3 = 5.999999\ldots$:

k	$a^{(k)}_{3,2}$	$a^{(k)}_{3,3}$	μ_k
6	$0.780088052879_{10^{-1}}$	6.00201597254	6.00000324468
7	$-0.838854980961_{10^{-7}}$	5.99999999996	5.99999999995
8	$0.12781135623_{10^{-19}}$	5.99999999995	

Die verbleibende (2×2)–Matrix hat die Eigenwerte

$$\lambda_2 = 9.016136303414\,, \qquad \lambda_1 = 12.3168759526\,.$$

Zum Vergleich noch das Ergebnis des *QR–Verfahrens ohne Shift* nach 11 Iterationsschritten: Die Elemente der Matrix A_{11} (wobei die gesuchten Näherungen für die Eigenwerte in der Diagonalen stehen) lauten

i	$a_{i,i-1}^{(11)}$	$a_{i,i}^{(11)}$
1		12.3165309125
2	$0.337457586637_{10^{-1}}$	9.01643819611
3	$0.114079951421_{10^{-1}}$	6.00004307566
4	$0.463086759853_{10^{-3}}$	2.98386376789
5	$0.202188244733_{10^{-10}}$	−0.316875952617

Bei der Weiterbehandlung der (4×4)–Matrix sind sogar weitere 23 Iterationen erforderlich um $a_{4,3}^{(34)} \approx 0.5_{10^{-10}}$ zu erreichen.

26 Das Jacobi-Verfahren

Im weiteren betrachten wir das Eigenwertproblem ausschließlich für symmetrische Matrizen $A = A^{\mathsf{T}} \in \mathbb{R}^{n \times n}$. Wir zerlegen

$$A = D - L - L^{\mathsf{T}}$$

in eine Diagonalmatrix D und eine echte untere Dreiecksmatrix L und bezeichnen mit

$$S(A) := \|L + L^{\mathsf{T}}\|_F^2 = \sum_{i,j=1,\, i \neq j}^{n} |a_{ij}|^2. \tag{26.1}$$

$S(A)$ ist ein Maß dafür, wie gut die Diagonalelemente von A die Eigenwerte von A approximieren (vgl. Satz 21.1 von Gerschgorin):

Proposition 26.1 *Ist d_{ii} ein beliebiges Diagonalelement von A, dann existiert ein $\lambda \in \sigma(A)$ mit*

$$|\lambda - d_{ii}| \leq \sqrt{S(A)}\,.$$

Beweis. Die Aussage folgt sofort aus dem Satz 22.4 von Wielandt-Hoffman, oder aus Korollar 22.3 zum Satz von Bauer-Fike: Demnach existiert ein $\lambda \in \sigma(A)$ mit

$$|\lambda - d_{ii}| \leq \|L + L^{\mathsf{T}}\|_2 \leq \|L + L^{\mathsf{T}}\|_F = \sqrt{S(A)}\,.$$

Die zweite Ungleichung folgt dabei aus der Tatsache, daß die Spektralnorm von der Euklidnorm induziert ist, während die Frobeniusnorm mit der Euklidnorm verträglich ist (vgl. Lemma **??**). □

Das **Jacobi-Verfahren** ist ein iteratives Verfahren, bei dem orthogonale Ähnlichkeitstransformationen

$$A_{k+1} = Q_k^* A_k Q_k$$

mit dem Ziel angewendet werden, (26.1) sukzessive zu verkleinern. Die Matrizen Q_k sind hierbei wieder Givens-Rotationen $G(i,j,\theta_k)$, wobei i, j und θ_k so gewählt sind, daß ein $a_{ij}^{(k+1)}$ zu Null rotiert wird.

Seien also i und j entsprechend gewählt: Dann gilt (mit $b_{ij} = a_{ij}^{(k+1)}$ und $a_{ij} = a_{ij}^k$)

$$\begin{bmatrix} b_{ii} & b_{ij} \\ b_{ji} & b_{jj} \end{bmatrix} = \begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} a_{ii} & a_{ij} \\ a_{ji} & a_{jj} \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \tag{26.2}$$

und folglich

$$\begin{aligned} 0 \stackrel{!}{=} b_{ij} &= \begin{bmatrix} ca_{ii} - sa_{ji} & ca_{ij} - sa_{jj} \end{bmatrix} \begin{bmatrix} s \\ c \end{bmatrix} \\ &= sca_{ii} - s^2 a_{ji} + c^2 a_{ij} - sca_{jj} = sc(a_{ii} - a_{jj}) + (c^2 - s^2)a_{ij} \,. \end{aligned} \tag{26.3}$$

Daher kann $G(i,j,\theta_k) = I$ gewählt werden, falls $a_{ij}^{(k)}$ bereits Null ist; andernfalls setzen wir $t = s/c$ und fordern, daß

$$t^2 - 2rt - 1 = 0\,, \qquad r = \frac{a_{ii} - a_{jj}}{2a_{ij}}\,.$$

Hierzu existieren zwei reelle Wurzeln $t_{1/2}$, deren Produkt -1 ergibt. Wir wählen die (betragsmäßig) kleinere Wurzel

$$t = \begin{cases} r - \sqrt{r^2+1} = -1/(r + \sqrt{r^2+1}) & r \geq 0\,, \\ r + \sqrt{r^2+1} = +1/(|r| + \sqrt{r^2+1}) & r < 0\,. \end{cases}$$

Dies ist für die Konvergenzgeschwindigkeit entscheidend, vgl. (26.8).

Wegen $t = \tan\theta_k$ und $|t| < 1$ ergibt sich somit

$$|\theta_k| \leq \frac{\pi}{4}\,. \tag{26.4}$$

Die beiden Einträge s und c von $G(i,j,\theta_k)$ ergeben sich aus t durch

$$c = (1+t^2)^{-1/2}\,, \qquad s = tc\,.$$

Schließlich ist noch

$$\begin{aligned} b_{ii} &= \begin{bmatrix} ca_{ii} - sa_{ji} & ca_{ij} - sa_{jj} \end{bmatrix} \begin{bmatrix} c \\ -s \end{bmatrix} &&= c^2 a_{ii} - 2sca_{ij} + s^2 a_{jj}\,, \\ b_{jj} &= \begin{bmatrix} sa_{ii} + ca_{ji} & sa_{ij} + ca_{jj} \end{bmatrix} \begin{bmatrix} s \\ c \end{bmatrix} &&= s^2 a_{ii} + 2sca_{ij} + c^2 a_{jj}\,. \end{aligned} \tag{26.5}$$

Um zu untersuchen, wie sich diese Transformation auf $S(A_k)$ auswirkt, wird das folgende Lemma benötigt.

Lemma 26.2 *Sind $Q, A \in \mathbb{K}^{n\times n}$ mit Q unitär, dann ist*

$$\|QA\|_F = \|A\|_F = \|AQ\|_F .$$

Beweis. Es reicht, die erste Gleichung nachzuweisen, da $\|AQ\|_F = \|Q^*A^*\|_F$. Sind $a_1, \ldots, a_n$ die Spalten von A, dann ist

$$\|A\|_F^2 = \sum_{i=1}^{n} \|a_i\|_2^2 .$$

Die Spalten von QA lauten $Qa_1, \ldots, Qa_n$. Daher gilt entsprechend

$$\|QA\|_F^2 = \sum_{i=1}^{n} \|Qa_i\|_2^2 = \sum_{i=1}^{n} \|a_i\|_2^2 = \|A\|_F^2 .$$

□

Daher folgt $\|A_{k+1}\|_F = \|A_k\|_F$, und wegen

$$S(A_k) = \|A_k\|_F^2 - \sum_{\nu=1}^{n} \left(a_{\nu\nu}^{(k)}\right)^2$$

ergibt sich

$$\begin{aligned} S(A_{k+1}) &= \|A_{k+1}\|_F^2 - \sum_{\nu=1}^{n} \left(a_{\nu\nu}^{(k+1)}\right)^2 = \|A_k\|_F^2 - \sum_{\nu=1}^{n} \left(a_{\nu\nu}^{(k+1)}\right)^2 \\ &= S(A_k) + \sum_{\nu=1}^{n} \left(\left(a_{\nu\nu}^{(k)}\right)^2 - \left(a_{\nu\nu}^{(k+1)}\right)^2 \right) \\ &= S(A_k) + \left(a_{ii}^{(k)}\right)^2 + \left(a_{jj}^{(k)}\right)^2 - \left(a_{ii}^{(k+1)}\right)^2 - \left(a_{jj}^{(k+1)}\right)^2 . \end{aligned}$$

Entsprechend kann man in (26.2) argumentieren und erhält:

$$a_{ii}^2 + a_{jj}^2 - b_{ii}^2 - b_{jj}^2 = S(B) - S(A) = 0 - 2a_{ij}^2 .$$

Da $a_{ij} = a_{ij}^{(k)}$ und $b_{ij} = a_{ij}^{(k+1)}$ folgt insgesamt

$$S(A_{k+1}) = S(A_k) - 2\left(a_{ij}^{(k)}\right)^2 . \tag{26.6}$$

Zusammenfassung: Bei einer Transformation des Jacobi-Verfahrens wird $S(A_k)$ kleiner.

Man unterscheidet zwei praktische Realisierungen des Jacobi-Verfahrens:

1. Beim **klassischen Jacobi-Verfahren** wird - im Hinblick auf (26.6) - (i,j) so ausgewählt, daß $|a_{ij}^{(k)}| = \max_{\nu\neq\mu} |a_{\nu\mu}^{(k)}|$. Die Maximumssuche hat jedoch den Aufwand $O(n^2)$ und ist damit recht teuer.

2. Billiger ist daher das **zyklische Jacobi-Verfahren**, in dem (i,j) die Nebendiagonalelemente zyklisch (zeilenweise) durchläuft:

$$(i,j) = (1,2),(1,3),\ldots,(1,n),(2,3),\ldots,(2,n),\ldots,(n-1,n)\,.$$

In beiden Fällen konvergiert $S(A_k) \to 0$, $k \to \infty$. Wir beweisen dies jedoch nur für das klassische Verfahren; der Beweis für die zyklische Variante ist erheblich schwieriger.

Satz 26.3 *Beim klassischen Jacobi-Verfahren gilt $S(A_k) \to 0$, $k \to \infty$, d. h. die Diagonaleinträge von A_k konvergieren gegen die Eigenwerte von A.*

Beweis. Da bei dieser Variante des Verfahrens im k-ten Schritt $|a_{ij}^{(k)}|$ maximal ist (unter allen $i \neq j$), gilt

$$S(A_k) \leq (n-1)n|a_{ij}^{(k)}|^2\,.$$

Eingesetzt in (26.6) folgt

$$S(A_{k+1}) \leq S(A_k) - \frac{2}{(n-1)n}S(A_k) = \Big(1 - \frac{2}{(n-1)n}\Big)S(A_k)\,. \tag{26.7}$$

Da $1 - 2/\big((n-1)n\big) \in [0,1)$, konvergiert $S(A_k)$ linear gegen 0, und die zweite Behauptung des Satzes folgt aus Propostion 26.1. □

Je Iterationsschritt sind $8n$ Multiplikationen notwendig. Hinreichend für $S(A_k) < \varepsilon$ ist nach (26.7) die Abschätzung

$$S(A_k) \leq \Big(1 - \frac{2}{(n-1)n}\Big)^k S(A_k) \overset{!}{<} \varepsilon\,.$$

Da $(1 - \frac{2}{(n-1)n})^k \approx 1 - \frac{2k}{(n-1)n}$ (für $n^2 \gg k$) erwarten wir etwa $O(n^2)$ Iterationen, und daher einen Gesamtaufwand $O(n^3)$.

Tatsächlich ist die Konvergenz beider Varianten wesentlich schneller als (26.7) suggeriert, nämlich lokal quadratisch. Der an und für sich nicht schwierige Beweis dieser Aussage ist jedoch sehr technisch, und wir wollen daher nur die Beweisidee für den Fall skizzieren, in dem die Eigenwerte von A paarweise verschieden sind, d. h.

$$|\lambda_i - \lambda_j| > 2\delta \quad \text{für} \quad \lambda_i \neq \lambda_j, \quad \lambda_i, \lambda_j \in \sigma(A)\,.$$

Nehmen wir ferner an, daß bereits $S(A^{(k)}) < \delta^2/4$ ist, dann folgt aus Proposition 26.1 für beliebige i und j und entsprechende Eigenwerte λ_i, λ_j:

$$\begin{aligned} |a_{ii}^{(k)} - a_{jj}^{(k)}| &= |a_{ii}^{(k)} + \lambda_i - \lambda_i - \lambda_j + \lambda_j - a_{jj}^{(k)}| > |\lambda_i - \lambda_j| - |\lambda_i - a_{ii}^{(k)}| - |\lambda_j - a_{jj}^{(k)}| \\ &> 2\delta - \delta/2 - \delta/2 = \delta\,. \end{aligned}$$

Ist $S(A^{(k)}) = \varepsilon^2 \ll \delta^2/4$, dann ist auch $\max_{i,j} |a_{ij}^{(k)}| \le \varepsilon \ll \delta$. Wir betrachten nun den Effekt von $n(n-1)/2$ Transformationen. Wegen (26.4) gilt

$$|\sin\theta_k| \le |\theta_k| = \frac{1}{2}|2\theta_k| \le \frac{1}{2}|\tan(2\theta_k)|\,, \tag{26.8}$$

und wegen (26.3) ist daher

$$|\tan 2\theta_k| = \left|\frac{\sin 2\theta_k}{\cos 2\theta_k}\right| = \left|\frac{2sc}{c^2 - s^2}\right| \underset{(26.3)}{=} \left|\frac{2a_{ij}}{a_{ii} - a_{jj}}\right|\,.$$

Eingesetzt in (26.8) ergibt sich

$$|s| \le \frac{|a_{ij}|}{|a_{ii} - a_{jj}|} < \frac{\varepsilon}{\delta} \qquad (\delta \text{ ist fest!})$$

und wegen $\varepsilon/\delta < 1$ ist

$$1 \ge |c| \ge \left(1 - \frac{\varepsilon^2}{\delta^2}\right)^{1/2} \ge 1 - \frac{\varepsilon}{\delta}\,.$$

In der $(k+1)$-ten Rotation gibt es nun drei Fälle zu unterscheiden, wobei immer $i \neq j$ ist:

(a) $a_{ij}^{(k)}$ wird nicht verändert,

(b) $a_{ij}^{(k)} \mapsto 0$,

(c) $a_{ij}^{(k)} \mapsto c a_{ij}^{(k)} + O(\varepsilon \max|a_{ij}^{(k)}|/\delta) = a_{ij}^{(k)} + O(\varepsilon^2)$.

In jedem Schritt wird also ein Element a_{ij} auf Null rotiert und fällt dadurch unter die Schranke $|a_{ij}| = O(\varepsilon^2)$; ist andererseits $|a_{ij}|$ erst einmal unterhalb dieser Schranke, dann bleibt es auch im weiteren Verlauf darunter. Beim zyklischen Verfahren wird so in $(n-1)n/2$ Schritten jedes Nebendiagonalelement unter die Schranke $O(\varepsilon^2)$ gedrückt; beim klassischen Jacobi-Verfahren ist das ebenfalls richtig, da $\max|a_{ij}^{(k)}| \gg \varepsilon^2$ höchstens $(n-1)n/2$ Schritte lang gelten kann. Mit $N = (n-1)n/2$ ist also

$$S(A^{(k+N)}) \le n(n-1)C\varepsilon^4 = n(n-1)CS(A^{(k)})^2$$

für ein geeignetes $C > 0$.

Ein praktischer Vergleich zwischen Jacobi- und QR-Verfahren ergibt dennoch eine Überlegenheit des QR-Verfahrens um etwa den Faktor 4 bis 10. Dafür ist das Jacobi-Verfahren einfacher zu programmieren und zu parallelisieren!

Beispiel. Die Eigenwerte der Matrix

$$A_0 = \begin{pmatrix} 20 & -7 & 3 & -2 \\ -7 & 5 & 1 & 4 \\ 3 & 1 & 3 & 1 \\ -2 & 4 & 1 & 2 \end{pmatrix}, \qquad S(A_0) = 160\,,$$

werden mit dem *zyklischen Jacobi–Verfahren* berechnet. Nach einem vollen Zyklus von sechs Rotationen lautet die resultierende Matrix

$$A_6 = \begin{pmatrix} \mathbf{23.523089} & -0.009053 & -0.238471 & 0.151640 \\ -0.009053 & \mathbf{-0.437554} & -1.397689 & 0.931475 \\ -0.238471 & -1.397689 & \mathbf{6.174371} & 0 \\ 0.151640 & 0.931475 & 0 & \mathbf{0.740095} \end{pmatrix}.$$

Die folgende Tabelle gibt die Größe $S(A_k)$ nach den ersten vier Zyklen an:

k	$S(A_k)$
6	5.802252
12	$1.387334_{10^{-2}}$
18	$1.094265_{10^{-9}}$
24	$3.7645_{10^{-31}}$

Die sehr rasche Konvergenz ist deutlich erkennbar, und setzt früh ein, weil die Eigenwerte von A_0 gut getrennt sind ($|\lambda_i - \lambda_j| \geq 2.334$ für $i \neq j$). Die Eigenwertnäherungen ergeben sich aus der Diagonalen von A_{24}:

$$\begin{aligned} \lambda_1 &= 23.527386\,, \\ \lambda_2 &= -1.160950\,, \\ \lambda_3 &= 6.460515\,, \\ \lambda_4 &= 1.173049\,. \end{aligned}$$