

## 6 Ausgleichsprobleme, Methode der kleinsten Quadrate

In manchen Wissenschaftszweigen, wie etwa Experimentalphysik und Biologie, stellt sich die Aufgabe, unbekannte Parameter einer Funktion, die entweder auf Grund eines Naturgesetzes oder von Modellannahmen gegeben ist, durch eine Reihe von Messungen oder Beobachtungen zu bestimmen. Die Anzahl der vorgenommenen Messungen ist in der Regel bedeutend größer als die Zahl der Parameter, um dadurch den unvermeidbaren Beobachtungsfehlern Rechnung zu tragen. Die resultierenden, überbestimmten Systeme von linearen oder nichtlinearen Gleichungen für die unbekannten Parameter sind im Allgemeinen nicht exakt lösbar, sondern man kann nur verlangen, dass die in den einzelnen Gleichungen auftretenden Abweichungen oder Residuen in einem zu präzisierenden Sinn minimal sind. In der betrachteten Situation wird aus wahrscheinlichkeitstheoretischen Gründen nur die *Methode der kleinsten Quadrate von Gauß* der Annahme von statistisch normalverteilten Messfehlern gerecht [Lud 71]. Für die Approximation von Funktionen kommt auch noch die Minimierung der maximalen Abweichung nach Tschebyscheff in Betracht [Übe 95]. Das Gaußsche Ausgleichsprinzip führt allerdings auf einfacher durchführbare Rechenverfahren als das Tschebyscheffsche Prinzip.

### 6.1 Lineare Ausgleichsprobleme, Normalgleichungen

Wir betrachten ein überbestimmtes System von  $N$  linearen Gleichungen in  $n$  Unbekannten

$$\mathbf{C}\mathbf{x} = \mathbf{d}, \quad \mathbf{C} \in \mathbb{R}^{N,n}, \quad \mathbf{d} \in \mathbb{R}^N, \quad \mathbf{x} \in \mathbb{R}^n, \quad N > n. \quad (6.1)$$

Da wir dieses System im Allgemeinen nicht exakt lösen können, führen wir das Residuum  $\mathbf{r} \in \mathbb{R}^N$  ein, um die so genannten *Fehlergleichungen*

$$\boxed{\mathbf{C}\mathbf{x} - \mathbf{d} = \mathbf{r}, \quad \mathbf{C} \in \mathbb{R}^{N,n}, \quad \mathbf{x} \in \mathbb{R}^n, \quad \mathbf{d}, \mathbf{r} \in \mathbb{R}^N} \quad (6.2)$$

zu erhalten, die komponentenweise

$$\sum_{k=1}^n c_{ik}x_k - d_i = r_i, \quad i = 1, 2, \dots, N, \quad n < N, \quad (6.3)$$

lauten. Für das Folgende setzen wir voraus, dass die Matrix  $\mathbf{C}$  den *Maximalrang  $n$*  besitzt, d.h. dass ihre Spalten linear unabhängig sind. Die Unbekannten  $x_k$  der Fehlergleichungen sollen nach dem Gaußschen Ausgleichsprinzip so bestimmt werden, dass die *Summe der Quadrate der Residuen  $r_i$*  minimal ist. Diese Forderung ist äquivalent dazu, das Quadrat

der euklidischen Norm des Residuenvektors zu minimieren. Aus (6.2) ergibt sich dafür

$$\begin{aligned} \mathbf{r}^T \mathbf{r} &= (\mathbf{C}\mathbf{x} - \mathbf{d})^T (\mathbf{C}\mathbf{x} - \mathbf{d}) = \mathbf{x}^T \mathbf{C}^T \mathbf{C}\mathbf{x} - \mathbf{x}^T \mathbf{C}^T \mathbf{d} - \mathbf{d}^T \mathbf{C}\mathbf{x} + \mathbf{d}^T \mathbf{d} \\ &= \mathbf{x}^T \mathbf{C}^T \mathbf{C}\mathbf{x} - 2(\mathbf{C}^T \mathbf{d})^T \mathbf{x} + \mathbf{d}^T \mathbf{d}. \end{aligned} \quad (6.4)$$

Das Quadrat der euklidischen Länge von  $\mathbf{r}$  ist nach (6.4) darstellbar als quadratische Funktion  $F(\mathbf{x})$  der  $n$  Unbekannten  $x_k$ . Zur Vereinfachung der Schreibweise definieren wir

$$\boxed{\mathbf{A} := \mathbf{C}^T \mathbf{C}, \quad \mathbf{b} := \mathbf{C}^T \mathbf{d}, \quad \mathbf{A} \in \mathbb{R}^{n,n}, \quad \mathbf{b} \in \mathbb{R}^n.} \quad (6.5)$$

Weil  $\mathbf{C}$  Maximalrang hat, ist die symmetrische Matrix  $\mathbf{A}$  *positiv definit*, denn für die zugehörige quadratische Form gilt

$$\begin{aligned} Q(\mathbf{x}) &= \mathbf{x}^T \mathbf{A} \mathbf{x} = \mathbf{x}^T \mathbf{C}^T \mathbf{C} \mathbf{x} = (\mathbf{C}\mathbf{x})^T (\mathbf{C}\mathbf{x}) \geq 0 \text{ für alle } \mathbf{x} \in \mathbb{R}^n, \\ Q(\mathbf{x}) &= 0 \Leftrightarrow (\mathbf{C}\mathbf{x}) = \mathbf{0} \Leftrightarrow \mathbf{x} = \mathbf{0}. \end{aligned}$$

Mit (6.5) lautet die zu minimierende quadratische Funktion  $F(\mathbf{x})$

$$\boxed{F(\mathbf{x}) := \mathbf{r}^T \mathbf{r} = \mathbf{x}^T \mathbf{A} \mathbf{x} - 2\mathbf{b}^T \mathbf{x} + \mathbf{d}^T \mathbf{d}.} \quad (6.6)$$

Die notwendige Bedingung dafür, dass  $F(\mathbf{x})$  ein Minimum annimmt, besteht darin, dass ihr Gradient  $\nabla F(\mathbf{x})$  verschwindet. Die  $i$ -te Komponente des Gradienten  $\nabla F(\mathbf{x})$  berechnet sich aus der expliziten Darstellung von (6.6) zu

$$\frac{\partial F(\mathbf{x})}{\partial x_i} = 2 \sum_{k=1}^n a_{ik} x_k - 2b_i, \quad i = 1, 2, \dots, n. \quad (6.7)$$

Nach Division durch 2 ergibt sich somit aus (6.7) als notwendige Bedingung für ein Minimum von  $F(\mathbf{x})$  das lineare Gleichungssystem

$$\boxed{\mathbf{A}\mathbf{x} = \mathbf{b}} \quad (6.8)$$

für die Unbekannten  $x_1, x_2, \dots, x_n$ . Man nennt (6.8) die *Normalgleichungen* zu den Fehlergleichungen (6.2). Da die Matrix  $\mathbf{A}$  wegen der getroffenen Voraussetzung für  $\mathbf{C}$  positiv definit ist, sind die Unbekannten  $x_k$  durch die Normalgleichungen (6.8) eindeutig bestimmt und lassen sich mit der Methode von Cholesky berechnen. Die Funktion  $F(\mathbf{x})$  wird durch diese Werte auch tatsächlich minimiert, denn die Hessesche Matrix von  $F(\mathbf{x})$ , gebildet aus den zweiten partiellen Ableitungen, ist die positiv definite Matrix  $2\mathbf{A}$ .

Die klassische Behandlung der Fehlergleichungen (6.2) nach dem Gaußschen Ausgleichsprinzip besteht somit aus den folgenden, einfachen Lösungsschritten.

$$\boxed{\begin{array}{ll} 1. \mathbf{A} = \mathbf{C}^T \mathbf{C}, \quad \mathbf{b} = \mathbf{C}^T \mathbf{d} & (\text{Normalgleichungen } \mathbf{A}\mathbf{x} = \mathbf{b}) \\ 2. \mathbf{A} = \mathbf{L}\mathbf{L}^T & (\text{Cholesky-Zerlegung}) \\ \mathbf{L}\mathbf{y} = \mathbf{b}, \quad \mathbf{L}^T \mathbf{x} = \mathbf{y} & (\text{Vorwärts-/Rücksubstitution}) \\ [3. \mathbf{r} = \mathbf{C}\mathbf{x} - \mathbf{d}] & (\text{Residuenberechnung}) \end{array}} \quad (6.9)$$

Für die Berechnung der Matrixelemente  $a_{ik}$  und der Komponenten  $b_i$  der Normalgleichungen erhält man eine einprägsame Rechenvorschrift, falls die Spaltenvektoren  $\mathbf{c}_i$  der Matrix  $\mathbf{C}$

eingeführt werden. Dann gelten die Darstellungen

$$a_{ik} = \mathbf{c}_i^T \mathbf{c}_k, \quad b_i = \mathbf{c}_i^T \mathbf{d}, \quad i, k = 1, 2, \dots, n, \quad (6.10)$$

so dass sich  $a_{ik}$  als Skalarprodukt des  $i$ -ten und  $k$ -ten Spaltenvektors von  $\mathbf{C}$  und  $b_i$  als Skalarprodukt des  $i$ -ten Spaltenvektors  $\mathbf{c}_i$  und der rechten Seite  $\mathbf{d}$  der Fehlergleichungen bestimmt. Aus Symmetriegründen sind in  $\mathbf{A}$  nur die Elemente in und unterhalb der Diagonalen zu berechnen. Der Rechenaufwand zur Aufstellung der Normalgleichungen beträgt somit  $Z_{\text{Normgl}} = nN(n+3)/2$  Multiplikationen. Zur Lösung von  $N$  Fehlergleichungen (6.3) in  $n$  Unbekannten mit dem Algorithmus (6.9) einschließlich der Berechnung der Residuen sind wegen (2.102)

$$Z_{\text{Fehlergl}} = \frac{1}{2}nN(n+5) + \frac{1}{6}n^3 + \frac{3}{2}n^2 + \frac{1}{3}n \approx \frac{n^2 N}{2} + O(n^3) \quad (6.11)$$

multiplikative Operationen und  $n$  Quadratwurzeln erforderlich.

**Beispiel 6.1.** Zu bestimmten, nicht äquidistanten Zeitpunkten  $t_i$  wird eine physikalische Größe  $z$  gemäß (6.12) beobachtet.

$i =$	1	2	3	4	5	6	7
$t_i =$	0.04	0.32	0.51	0.73	1.03	1.42	1.60
$z_i =$	2.63	1.18	1.16	1.54	2.65	5.41	7.67

(6.12)

Es ist bekannt, dass  $z$  eine quadratische Funktion der Zeit  $t$  ist, und es sollen ihre Parameter nach der Methode der kleinsten Quadrate bestimmt werden. Mit dem Ansatz

$$z(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2 \quad (6.13)$$

lautet die  $i$ -te Fehlergleichung

$$\alpha_0 + \alpha_1 t_i + \alpha_2 t_i^2 - z_i = r_i, \quad i = 1, 2, \dots, 7,$$

und damit das Gleichungssystem (6.1)

$$\begin{pmatrix} 1 & 0.04 & 0.0016 \\ 1 & 0.32 & 0.1024 \\ 1 & 0.51 & 0.2601 \\ 1 & 0.73 & 0.5329 \\ 1 & 1.03 & 1.0609 \\ 1 & 1.42 & 2.0164 \\ 1 & 1.60 & 2.5600 \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} 2.63 \\ 1.18 \\ 1.16 \\ 1.54 \\ 2.65 \\ 5.41 \\ 7.67 \end{pmatrix} \quad (6.14)$$

Daraus ergeben sich bei sechsstelliger Rechnung die Normalgleichungen

$$\begin{pmatrix} 7.00000 & 5.65000 & 6.53430 \\ 5.65000 & 6.53430 & 8.60652 \\ 6.53430 & 8.60652 & 12.1071 \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} 22.2400 \\ 24.8823 \\ 34.6027 \end{pmatrix} \quad (6.15)$$

Die Cholesky-Zerlegung und die Vorwärts- und Rücksubstitution ergeben

$$\mathbf{L} = \begin{pmatrix} 2.64575 & & \\ 2.13550 & 1.40497 & \\ 2.46973 & 2.37187 & 0.617867 \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} 8.40593 \\ 4.93349 \\ 3.46466 \end{pmatrix},$$

$$\boldsymbol{\alpha} = \begin{pmatrix} 2.74928 \\ -5.95501 \\ 5.60745 \end{pmatrix}. \quad (6.16)$$

Die resultierende quadratische Funktion

$$z(t) = 2.74928 - 5.95501 t + 5.60745 t^2 \quad (6.17)$$

hat den Residuenvektor  $\mathbf{r} \doteq (-0.1099, 0.2379, 0.0107, -0.1497, -0.0854, 0.1901, -0.0936)^T$ . Die Messpunkte und der Graph der quadratischen Funktion sind in Abb. 6.1 dargestellt. Die Residuen  $r_i$  sind die Ordinatendifferenzen zwischen der Kurve  $z(t)$  und den Messpunkten und können als Korrekturen der Messwerte interpretiert werden, so dass die korrigierten Messpunkte auf die Kurve zu liegen kommen.  $\triangle$

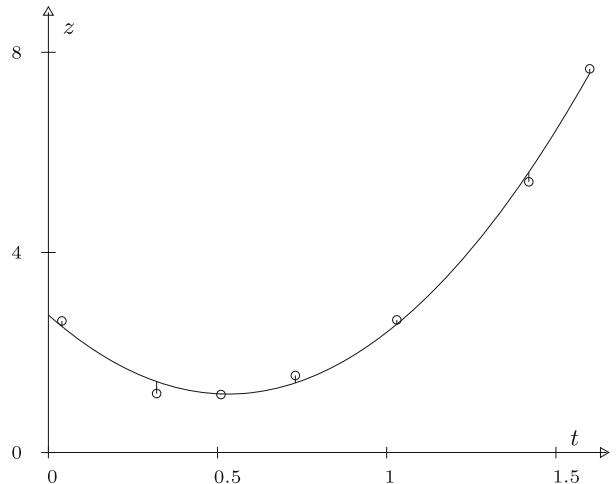


Abb. 6.1 Ausgleichung mit quadratischer Funktion.

Für die Lösungsmethode der Normalgleichungen besteht eine numerische Problematik darin, dass die Konditionszahl der Matrix  $\mathbf{A}$  der Normalgleichungen sehr groß sein kann. Die berechnete Lösung  $\tilde{\mathbf{x}}$  kann in diesem Fall einen entsprechend großen relativen Fehler aufweisen (vgl. Abschnitt 2.2.1). Da die Matrixelemente  $a_{ik}$  und die Komponenten  $b_i$  des Konstantenvektors als Skalarprodukte (6.10) zu berechnen sind, sind Rundungsfehler unvermeidlich. Die Matrix  $\mathbf{A}$  der Normalgleichungen (6.15) besitzt die Konditionszahl  $\kappa_2(\mathbf{A}) = \lambda_{\max}/\lambda_{\min} \doteq 23.00/0.09000 \doteq 256$ . Bei sechsstelliger Rechnung sind nach Abschnitt 2.2.1 in der Lösung  $\tilde{\boldsymbol{\alpha}}$  nur die drei ersten Ziffern garantiert richtig. Eine zwölfstellige Rechnung liefert in der Tat für  $z(t)$  mit den auf sieben Ziffern gerundeten Koeffizienten

$$z(t) \doteq 2.749198 - 5.954657t + 5.607247t^2. \quad (6.18)$$

**Beispiel 6.2.** Zur Illustration der möglichen schlechten Kondition von Normalgleichungen betrachten wir ein typisches Ausgleichsproblem. Zur analytischen Beschreibung der Kennlinie eines nichtlinearen Übertragungselementes  $y = f(x)$  sind für exakte Eingangsgrößen  $x_i$  die Ausgangsgrößen  $y_i$  beobachtet worden.

$x =$	0.2	0.5	1.0	1.5	2.0	3.0	
$y =$	0.3	0.5	0.8	1.0	1.2	1.3	

(6.19)

Das Übertragungselement verhält sich für kleine  $x$  linear, und die Kennlinie besitzt für große  $x$  eine horizontale Asymptote. Um diesem Verhalten Rechnung zu tragen, soll für  $f(x)$  der Ansatz

$$f(x) = \alpha_1 \frac{x}{1+x} + \alpha_2 (1 - e^{-x}) \quad (6.20)$$

mit den beiden zu bestimmenden Parametern  $\alpha_1$  und  $\alpha_2$  verwendet werden. Bei sechsstelliger Rechnung lauten das überbestimmte System (6.1), die Normalgleichungen und die Linksdreiecksmatrix  $L$  der Cholesky-Zerlegung

$$\begin{pmatrix} 0.166667 & 0.181269 \\ 0.333333 & 0.393469 \\ 0.500000 & 0.632121 \\ 0.600000 & 0.776870 \\ 0.666667 & 0.864665 \\ 0.750000 & 0.950213 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} 0.3 \\ 0.5 \\ 0.8 \\ 1.0 \\ 1.2 \\ 1.3 \end{pmatrix},$$

$$\begin{pmatrix} 1.75583 & 2.23266 \\ 2.23266 & 2.84134 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} 2.99167 \\ 3.80656 \end{pmatrix}, \quad L = \begin{pmatrix} 1.32508 & \\ 1.68492 & 0.0487852 \end{pmatrix}.$$
(6.21)

Vorwärts- und Rücksubstitution liefern mit  $\alpha_1 = 0.384196$  und  $\alpha_2 = 1.03782$  die gesuchte Darstellung für die Kennlinie

$$f(x) = 0.384196 \frac{x}{1+x} + 1.03782 (1 - e^{-x}) \quad (6.22)$$

mit dem Residuenvektor  $r \doteq (-0.0478, 0.0364, 0.0481, 0.0368, -0.0465, -0.0257)^T$ . Aus den beiden Eigenwerten  $\lambda_1 \doteq 4.59627$  und  $\lambda_2 \doteq 0.0009006$  der Matrix der Normalgleichungen folgt die Konditionszahl  $\kappa_2(A) \doteq 5104$ . Da die berechneten Werte der Parameter  $\alpha_1$  und  $\alpha_2$  einen entsprechend großen relativen Fehler aufweisen können, wurde das Fehlergleichungssystem mit zwölfstelliger Rechnung behandelt. Sie lieferte die Werte  $\alpha_1 \doteq 0.382495$  und  $\alpha_2 \doteq 1.03915$ , für welche die Residuen aber mit den oben angegebenen Werten übereinstimmen. Das Resultat zeigt die große Empfindlichkeit der Parameter  $\alpha_1$  und  $\alpha_2$ .  $\triangle$

## 6.2 Methoden der Orthogonaltransformation

Die aufgezeigte Problematik der Normalgleichungen infolge möglicher schlechter Kondition verlangt nach numerisch sicheren Verfahren zur Lösung der Fehlergleichungen nach der Methode der kleinsten Quadrate. Die Berechnung der Normalgleichungen ist zu vermeiden, und die gesuchte Lösung ist durch eine direkte Behandlung der Fehlergleichungen zu bestimmen. Im Folgenden werden zwei Varianten beschrieben.

### 6.2.1 Givens-Transformation

Als wesentliche Grundlage für die Verfahren dient die Tatsache, dass die Länge eines Vektors unter orthogonalen Transformationen invariant bleibt, siehe Lemma 5.21. Zur Lösung der Fehlergleichungen  $\mathbf{C}\mathbf{x} - \mathbf{d} = \mathbf{r}$  nach dem Gaußschen Ausgleichsprinzip dürfen sie mit einer orthogonalen Matrix  $\mathbf{Q} \in \mathbb{R}^{N,N}$  transformiert werden, ohne dadurch die Summe der Quadrate der Residuen zu verändern. Somit wird das Fehlergleichungssystem (6.2) ersetzt durch das äquivalente System

$$\mathbf{Q}^T \mathbf{C} \mathbf{x} - \mathbf{Q}^T \mathbf{d} = \mathbf{Q}^T \mathbf{r} =: \hat{\mathbf{r}}. \quad (6.23)$$

Die orthogonale Matrix  $\mathbf{Q}$  wird in (6.23) so gewählt werden, dass die Matrix  $\mathbf{Q}^T \mathbf{C}$  eine spezielle Gestalt aufweist. In Verallgemeinerung des Satzes 5.7 gilt der

**Satz 6.1.** *Zu jeder Matrix  $\mathbf{C} \in \mathbb{R}^{N,n}$  mit Maximalrang  $n < N$  existiert eine orthogonale Matrix  $\mathbf{Q} \in \mathbb{R}^{N,N}$  derart, dass*

$$\mathbf{C} = \mathbf{Q} \hat{\mathbf{R}} \quad \text{mit } \hat{\mathbf{R}} = \begin{pmatrix} \mathbf{R} \\ \mathbf{0} \end{pmatrix}, \quad \mathbf{R} \in \mathbb{R}^{n,n}, \quad \mathbf{0} \in \mathbb{R}^{(N-n),n} \quad (6.24)$$

gilt, wo  $\mathbf{R}$  eine reguläre Rechtsdreiecksmatrix und  $\mathbf{0}$  eine Nullmatrix darstellen.

*Beweis.* Analog zur Beweisführung von Satz 5.7 erkennt man, dass die sukzessive Multiplikation der Matrix  $\mathbf{C}$  von links mit Rotationsmatrizen  $\mathbf{U}^T(p, q; \varphi)$  mit den Rotationsindexpaaren

$$(1, 2), (1, 3), \dots, (1, N), (2, 3), (2, 4), \dots, (2, N), (3, 4), \dots, (n, N) \quad (6.25)$$

und nach (5.89) gewählten Drehwinkeln die aktuellen Matrixelemente in der Reihenfolge

$$c_{21}, c_{31}, \dots, c_{N1}, c_{32}, c_{42}, \dots, c_{N2}, c_{43}, \dots, c_{Nn} \quad (6.26)$$

eliminiert. Nach  $N^* = n(2N - n - 1)/2$  Transformationsschritten gilt (6.24) mit

$$\mathbf{U}_{N^*}^T \dots \mathbf{U}_2^T \mathbf{U}_1^T \mathbf{C} = \mathbf{Q}^T \mathbf{C} = \hat{\mathbf{R}}, \quad \text{oder} \quad \mathbf{C} = \mathbf{Q} \hat{\mathbf{R}}. \quad (6.27)$$

Da die orthogonale Matrix  $\mathbf{Q}$  regulär ist, sind der Rang von  $\mathbf{C}$  und von  $\hat{\mathbf{R}}$  gleich  $n$ , und folglich ist die Rechtsdreiecksmatrix  $\mathbf{R}$  regulär.  $\square$

Mit der nach Satz 6.1 gewählten Matrix  $\mathbf{Q}$  lautet (6.23)

$$\hat{\mathbf{R}}\mathbf{x} - \hat{\mathbf{d}} = \hat{\mathbf{r}}, \quad \hat{\mathbf{d}} = \mathbf{Q}^T \mathbf{d}. \quad (6.28)$$

Das orthogonal transformierte Fehlergleichungssystem (6.28) hat wegen (6.24) die Form

$$\begin{aligned} r_{11}x_1 + r_{12}x_2 + \dots + r_{1n}x_n - \hat{d}_1 &= \hat{r}_1 \\ r_{22}x_2 + \dots + r_{2n}x_n - \hat{d}_2 &= \hat{r}_2 \\ \ddots &\quad \vdots \quad \vdots \quad \vdots \\ r_{nn}x_n - \hat{d}_n &= \hat{r}_n \\ -\hat{d}_{n+1} &= \hat{r}_{n+1} \\ &\quad \vdots \quad \vdots \\ -\hat{d}_N &= \hat{r}_N \end{aligned} \tag{6.29}$$

Die Methode der kleinsten Quadrate verlangt nun, dass die Summe der Quadrate der transformierten Residuen  $\hat{r}_i$  minimal sei. Die Werte der letzten  $(N - n)$  Residuen sind durch die zugehörigen  $\hat{d}_i$  unabhängig von den Unbekannten  $x_k$  vorgegeben. Die Summe der Residuenquadrate ist genau dann minimal, falls  $\hat{r}_1 = \hat{r}_2 = \dots = \hat{r}_n = 0$  gilt, und sie ist gleich der Summe der Quadrate der letzten  $(N - n)$  Residuen  $\hat{r}_j$ . Folglich sind die Unbekannten  $x_k$  gemäß (6.29) gegeben durch das lineare Gleichungssystem

$$\mathbf{R}\mathbf{x} = \hat{\mathbf{d}}_1, \tag{6.30}$$

worin  $\hat{\mathbf{d}}_1 \in \mathbb{R}^n$  den Vektor bedeutet, welcher aus den  $n$  ersten Komponenten von  $\hat{\mathbf{d}} = \mathbf{Q}^T \mathbf{d}$  gebildet wird. Den Lösungsvektor  $\mathbf{x}$  erhält man aus (6.30) durch Rücksubstitution.

Falls man sich nur für die Unbekannten  $x_k$  des Fehlergleichungssystems  $\mathbf{C}\mathbf{x} - \mathbf{d} = \mathbf{r}$  interessiert, ist der Algorithmus bereits vollständig beschrieben. Sollen auch die Residuen  $r_i$  berechnet werden, können dieselben im Prinzip durch Einsetzen in die gegebenen Fehlergleichungen ermittelt werden. Dieses Vorgehen erfordert, dass die Matrix  $\mathbf{C}$  und der Vektor  $\mathbf{d}$  noch verfügbar sind. Es ist in Bezug auf den Speicherbedarf ökonomischer, den Residuenvektor  $\mathbf{r}$  wegen (6.23) und (6.27) aus  $\hat{\mathbf{r}}$  gemäß

$$\mathbf{r} = \mathbf{Q}\hat{\mathbf{r}} = \mathbf{U}_1 \mathbf{U}_2 \dots \mathbf{U}_{N^*} \hat{\mathbf{r}} \tag{6.31}$$

zu berechnen. Die Information für die einzelnen Rotationsmatrizen  $\mathbf{U}_k$  kann in den  $\varrho$ -Werten (5.64) an der Stelle der eliminierten Matrixelemente  $c_{ij}$  gespeichert werden. Die ersten  $n$  Komponenten des Residuenvektors  $\hat{\mathbf{r}}$  sind gleich null, und die letzten  $(N - n)$  Komponenten sind durch die entsprechenden  $\hat{d}_j$  definiert. Der gesuchte Residuenvektor  $\mathbf{r}$  entsteht somit aus  $\hat{\mathbf{r}}$  durch sukzessive Multiplikation mit den Rotationsmatrizen  $\mathbf{U}_k$  in der umgekehrten Reihenfolge wie sie bei der Transformation von  $\mathbf{C}$  in  $\hat{\mathbf{R}}$  angewandt wurden. Zusammenfassend besteht die Behandlung von Fehlergleichungen (6.2) nach dem Gaußschen Ausgleichsprinzip mit Hilfe der Orthogonaltransformation mit Givens-Rotationen aus folgenden Schritten.

- |   |  |
|---|--|
| 1. $\mathbf{C} = \mathbf{Q}\hat{\mathbf{R}}$    | (QR-Zerlegung, Givens-Rotationen)              |
| 2. $\hat{\mathbf{d}} = \mathbf{Q}^T \mathbf{d}$ | (Transformation von $\mathbf{d}$ )             |
| 3. $\mathbf{R}\mathbf{x} = \hat{\mathbf{d}}_1$  | (Rücksubstitution)                             |
| [ 4. $\mathbf{r} = \mathbf{Q}\hat{\mathbf{r}}$  | (Rücktransformation von $\hat{\mathbf{r}}$ ) ] |

Der erste und der zweite Schritt von (6.32) werden im Allgemeinen gleichzeitig ausgeführt, falls nur ein Fehlergleichungssystem zu lösen ist. Sobald aber mehrere Systeme (6.2) mit derselben Matrix  $\mathbf{C}$ , aber verschiedenen Vektoren  $\mathbf{d}$  nacheinander zu behandeln sind, so

ist es zweckmäßig die beiden Schritte zu trennen. Für die Ausführung des zweiten Schrittes muss die Information über die Rotationen verfügbar sein. Der Rechenaufwand für den Rechenprozess (6.32) beträgt

$$Z_{\text{FGIGivens}} = 2nN(n+6) - \frac{2}{3}n^3 - \frac{13}{2}n^2 - \frac{35}{6}n \approx 2n^2N + O(n^3) \quad (6.33)$$

multiplikative Operationen und  $n(2N - n - 1)$  Quadratwurzeln. Im Vergleich zu (6.11) ist er um einen Faktor zwischen 2 und 4 größer, abhängig vom Verhältnis von  $N$  zu  $n$ . Der Mehraufwand rechtfertigt sich dadurch, dass die berechnete Lösung  $\tilde{x}$  der Fehlgleichungen bei gleicher Rechengenauigkeit einen bedeutend kleineren relativen Fehler aufweist.

*Ausgleichslösung der Fehlgleichungen (6.3) mit Givens-Rotationen*

Für  $j = 1, 2, \dots, n$  :

für  $i = j+1, j+2, \dots, N$  :

falls  $c_{ij} \neq 0$  :

falls  $|c_{jj}| < \tau \times |c_{ij}|$  :

$$w = -c_{ij}; \gamma = 0; \sigma = 1; \varrho = 1,$$

sonst

$$w = \text{sgn}(c_{jj}) \times \sqrt{c_{jj}^2 + c_{ij}^2}$$

$$\gamma = c_{jj}/w; \sigma = -c_{ij}/w$$

$$\text{falls } |\sigma| < \gamma : \varrho = \sigma, \text{ sonst } \varrho = \text{sgn}(\sigma)/\gamma$$

$$c_{jj} = w; c_{ij} = \varrho$$

für  $k = j+1, j+2, \dots, n$  :

$$h = \gamma \times c_{jk} - \sigma \times c_{ik}$$

$$c_{ik} = \sigma \times c_{jk} + \gamma \times c_{ik}; c_{jk} = h$$

$$h = \gamma \times d_j - \sigma \times d_i; d_i = \sigma \times d_j + \gamma \times d_i; d_j = h$$

Für  $i = n, n-1, \dots, 1$  :

$$s = d_i; r_i = 0$$

für  $k = i+1, i+2, \dots, n$  :

$$s = s - c_{ik} \times x_k$$

$$x_i = s/c_{ii}$$

Für  $i = n+1, n+2, \dots, N$  :

$$r_i = -d_i$$

Für  $j = n, n-1, \dots, 1$  :

für  $i = N, N-1, \dots, j+1$  :

$$\varrho = c_{ij}$$

$$\text{falls } \varrho = 1 : \gamma = 0; \sigma = 1,$$

sonst

$$\text{falls } |\varrho| < 1 : \sigma = \varrho; \gamma = \sqrt{1 - \sigma^2},$$

$$\text{sonst } \gamma = 1/|\varrho|; \sigma = \text{sgn}(\varrho) \times \sqrt{1 - \gamma^2}$$

$$h = \gamma \times r_j + \sigma \times r_i; r_i = -\sigma \times r_j + \gamma \times r_i; r_j = h$$

(6.34)

Die Berechnung der Lösung  $\mathbf{x}$  eines Fehlergleichungssystems  $\mathbf{C}\mathbf{x} - \mathbf{d} = \mathbf{r}$  mit dem Verfahren (6.32) besitzt die algorithmische Beschreibung (6.34). Dabei wird angenommen, dass die Matrix  $\hat{\mathbf{R}}$  an der Stelle von  $\mathbf{C}$  aufgebaut wird und die Werte  $\varrho$  (5.64) am Platz der eliminierten Matrixelemente  $c_{ij}$  gespeichert werden. Die beiden Schritte 1 und 2 von (6.32) werden gleichzeitig ausgeführt. Zur Vermeidung eines Namenskonfliktes bedeuten  $\gamma = \cos \varphi$  und  $\sigma = \sin \varphi$ . Schließlich ist  $\tau$  die Maschinengenauigkeit, also die kleinste positive Zahl des Rechners mit  $1 + \tau \neq 1$ .

**Beispiel 6.3.** Das Fehlergleichungssystem (6.14) von Beispiel 6.1 wird mit der Methode der Orthogonaltransformation (6.32) behandelt. Bei sechsstelliger Rechnung lauten die Matrix  $\hat{\mathbf{R}}$  der QR-Zerlegung von  $\mathbf{C}$  mit den  $\varrho$ -Werten anstelle der Nullen, der transformierte Vektor  $\hat{\mathbf{d}}$  und der Residuenvektor  $\mathbf{r}$

$$\hat{\mathbf{R}} = \left( \begin{array}{ccc|c} 2.64575 & 2.13549 & 2.46973 & 8.40594 \\ -1.41421 & 1.40497 & 2.37187 & 4.93353 \\ -0.577351 & -1.68878 & 0.617881 & 3.46460 \\ \hline -0.500000 & -1.51616 & -2.53186 & -0.128686 \\ -0.447213 & -1.49516 & -2.19433 & -0.234145 \\ -0.408248 & -1.46945 & -1.98963 & -0.211350 \\ -0.377965 & -0.609538 & -0.635138 & 0.165290 \end{array} \right), \quad \hat{\mathbf{d}} = \left( \begin{array}{c} -0.110017 \\ 0.237881 \\ 0.0108260 \\ \hline -0.149594 \\ -0.0853911 \\ 0.190043 \\ -0.0937032 \end{array} \right), \quad \mathbf{r} = \left( \begin{array}{c} -0.110017 \\ 0.237881 \\ 0.0108260 \\ \hline -0.149594 \\ -0.0853911 \\ 0.190043 \\ -0.0937032 \end{array} \right)$$

Der Lösungsvektor  $\alpha = (2.74920, -5.95463, 5.60723)^T$  ergibt sich aus der Rücksubstitution mit der Matrix  $\mathbf{R}$  und den ersten drei Komponenten von  $\hat{\mathbf{d}}$ . Er stimmt mit der exakten Lösung (6.18) bis auf höchstens drei Einheiten der letzten Ziffer überein und weist einen um den Faktor 9 kleineren relativen Fehler auf als (6.17).  $\triangle$

**Beispiel 6.4.** Die Methode der Orthogonaltransformation liefert auch für das Fehlergleichungssystem (6.21) von Beispiel 6.2 eine Näherungslösung mit kleinerem Fehler. Die wesentlichen Ergebnisse sind bei sechsstelliger Rechnung

$$\hat{\mathbf{R}} = \left( \begin{array}{cc|c} 1.32508 & 1.68492 & 2.25773 \\ -2.23607 & 0.0486849 & 0.0505901 \\ -1.67332 & -2.47237 & 0.0456739 \\ \hline -0.693334 & -0.653865 & 0.0314087 \\ -0.610277 & -0.403536 & 0.0778043 \\ -0.566004 & 0.0862074 & 0.0313078 \end{array} \right), \quad \hat{\mathbf{d}} = \left( \begin{array}{c} -0.0478834 \\ 0.0363745 \\ 0.0481185 \\ \hline 0.0367843 \\ -0.0464831 \\ -0.0257141 \end{array} \right), \quad \mathbf{r} = \left( \begin{array}{c} -0.0478834 \\ 0.0363745 \\ 0.0481185 \\ \hline 0.0367843 \\ -0.0464831 \\ -0.0257141 \end{array} \right).$$

Die resultierenden Parameter  $\alpha_1 = 0.382528$  und  $\alpha_2 = 1.03913$  sind jetzt auf vier Dezimalstellen nach dem Komma richtig.  $\triangle$

Wie die Zahlenbeispiele vermuten lassen, besteht zwischen der klassischen Methode der Normalgleichungen und der Methode der Orthogonaltransformation ein Zusammenhang. So wird die Matrix  $\mathbf{C}$  der Fehlergleichungen nach (6.24) zerlegt in  $\mathbf{C} = \mathbf{Q}\hat{\mathbf{R}}$ , und somit gilt für die Matrix  $\mathbf{A}$  der Normalgleichungen wegen der Orthogonalität von  $\mathbf{Q}$  und der Struktur von  $\hat{\mathbf{R}}$

$$\mathbf{A} = \mathbf{C}^T \mathbf{C} = \hat{\mathbf{R}}^T \mathbf{Q}^T \mathbf{Q} \hat{\mathbf{R}} = \hat{\mathbf{R}}^T \hat{\mathbf{R}} = \mathbf{R}^T \mathbf{R}. \quad (6.35)$$

Die Cholesky-Zerlegung  $\mathbf{A} = \mathbf{L}\mathbf{L}^T$  einer symmetrischen, positiv definiten Matrix ist eindeutig, falls die Diagonalelemente von  $\mathbf{L}$  positiv gewählt werden. Folglich muss wegen (6.35) die Matrix  $\mathbf{R}$  im Wesentlichen mit  $\mathbf{L}^T$  übereinstimmen, d.h. bis auf eventuell verschiedene Vorzeichen von Zeilen in  $\mathbf{R}$ . Obwohl theoretisch die beiden Verfahren im Wesentlichen die gleichen Dreiecksmatrizen liefern, besteht numerisch doch ein entscheidender Unterschied für deren Berechnung.

Um eine plausible Erklärung dafür zu erhalten, betrachten wir zuerst die Entstehung der Diagonalelemente von  $\mathbf{R}$ . Die Folge von orthogonalen Givens-Transformationen lässt die euklidischen Normen der Spaltenvektoren  $\mathbf{c}_j$  von  $\mathbf{C}$  invariant. Nach Elimination der Elemente der ersten Spalte gilt somit  $|r_{11}| = \|\mathbf{c}_1\|_2$ . Weil das geänderte Element  $c'_{12}$  während der Elimination der Elemente der zweiten Spalte unverändert stehen bleibt, folgt für das zweite Diagonalelement  $|r_{22}| = \|\mathbf{c}'_2 - c'_{12}\mathbf{e}_1\|_2 \leq \|\mathbf{c}_2\|_2$ . Allgemein gilt  $|r_{jj}| \leq \|\mathbf{c}_j\|_2$ ,  $j = 1, 2, \dots, n$ . Wichtig ist nun die Tatsache, dass die Diagonalelemente  $r_{jj}$  aus den Vektoren  $\mathbf{c}_j$  als euklidische Normen von Teilvektoren nach orthogonalen Transformationen entstehen.

Die Diagonalelemente  $l_{jj}$  von  $\mathbf{L}$  der Cholesky-Zerlegung von  $\mathbf{A} = \mathbf{C}^T \mathbf{C}$  entstehen aus den Diagonalelementen  $a_{jj}$ , nachdem von  $a_{jj}$  Quadrate von Matrixelementen  $l_{jk}$  subtrahiert worden sind. Nach (6.10) ist aber  $a_{jj} = \mathbf{c}_j^T \mathbf{c}_j = \|\mathbf{c}_j\|_2^2$  gleich dem Quadrat der euklidischen Norm des Spaltenvektors  $\mathbf{c}_j$ . Im Verlauf der Cholesky-Zerlegung von  $\mathbf{A}$  wird folglich solange mit den Quadraten der Norm gerechnet, bis mit dem Ziehen der Quadratwurzel mit  $l_{jj}$  die Norm eines Vektors erscheint. Da nun das *Quadrat* der Norm im Verfahren von Cholesky verkleinert wird, kann bei endlicher Rechengenauigkeit im Fall einer starken Reduktion des Wertes der relative Fehler infolge Auslöschung bedeutend größer sein als im Fall der Orthogonaltransformation, bei der mit den Vektoren  $\mathbf{c}_j$  selbst gearbeitet wird. Deshalb ist die Matrix  $\mathbf{R}$  numerisch genauer als die Matrix  $\mathbf{L}$  der Cholesky-Zerlegung.

Schließlich berechnet sich die Lösung  $\mathbf{x}$  der Fehlgleichungen aus dem Gleichungssystem (6.30) mit der genaueren Rechtsdreiecksmatrix  $\mathbf{R}$  und dem Vektor  $\hat{\mathbf{d}}_1$ . Dieser ist aus  $\mathbf{d}$  durch eine Folge von Orthogonaltransformationen entstanden, die sich durch gute numerische Eigenschaften auszeichnen. Der Vektor  $\hat{\mathbf{d}}_1$  ist theoretisch im Wesentlichen gleich dem Vektor  $\mathbf{y}$ , der sich durch Vorwärtssubstitution aus  $\mathbf{Ly} = \mathbf{b}$  ergibt, der aber mit  $\mathbf{L}$  bereits ungenauer ist als  $\hat{\mathbf{d}}_1$ .

Der Rechenaufwand des Verfahrens (6.32) kann reduziert werden, falls die schnelle Givens-Transformation von Abschnitt 5.4.3 angewandt wird. Nach der dort ausführlich dargestellten Methode erhalten wir wegen (6.28) sowohl die transformierte Matrix  $\hat{\mathbf{R}}$  als auch die transformierten Vektoren  $\hat{\mathbf{d}}$  und  $\hat{\mathbf{r}}$  in faktorisierter Form

$$\hat{\mathbf{R}} = \mathbf{D}\hat{\tilde{\mathbf{R}}}, \quad \hat{\mathbf{d}} = \mathbf{D}\hat{\tilde{\mathbf{d}}}, \quad \hat{\mathbf{r}} = \mathbf{D}\hat{\tilde{\mathbf{r}}}, \quad \mathbf{D} \in \mathbb{R}^{N,N}. \quad (6.36)$$

Da die Matrix  $\mathbf{D}$  regulär ist, ist sie für die Berechnung der Lösung  $\mathbf{x}$  aus dem zu (6.29) analogen Fehlgleichungssystem irrelevant, so dass die explizite Bildung von  $\hat{\mathbf{R}}$  und  $\hat{\mathbf{d}}$  nicht nötig ist. Ist  $\tilde{\mathbf{R}}$  die Rechtsdreiecksmatrix in  $\hat{\mathbf{R}}$  und  $\hat{\tilde{\mathbf{d}}}_1$  der Vektor, gebildet aus den  $n$  ersten Komponenten von  $\hat{\tilde{\mathbf{d}}}$ , so berechnet sich  $\mathbf{x}$  aus

$$\tilde{\mathbf{R}}\mathbf{x} = \hat{\tilde{\mathbf{d}}}_1. \quad (6.37)$$

Zur Bestimmung des Residuenvektors  $\mathbf{r}$  aus  $\hat{\mathbf{r}}$  in Analogie zu (6.31) wäre die Information über die Transformationen erforderlich. Da aber pro Schritt zwei Zahlenwerte nötig sind,

ist es wohl sinnvoller,  $\mathbf{r}$  aus den gegebenen Fehlergleichungen  $\mathbf{Cx} - \mathbf{d} = \mathbf{r}$  zu berechnen. Falls man sich nur für die Summe der Quadrate der Residuen interessiert, kann sie direkt auf den letzten  $(N - n)$  Komponenten von  $\hat{\mathbf{d}}$  und den zugehörigen Diagonalelementen von  $\mathbf{D}$  bestimmt werden.

Der Rechenaufwand zur Elimination der aktuellen Matrixelemente  $c_{ij}$ ,  $i = j+1, j+2, \dots, N$ , der  $j$ -ten Spalte beträgt nach Abschnitt 5.4.3 unter Einschluss der Transformation der beiden Komponenten im Vektor  $\mathbf{d}$  insgesamt  $(N - j)(2n - 2j + 12)$  Operationen. Nach Summation über  $j$  von 1 bis  $n$  ergibt sich der totale Rechenaufwand zur Bestimmung der Lösung  $\mathbf{x}$  und des Residuenvektors  $\mathbf{r}$  zu

$$Z_{\text{FGISG}} = nN(n + 12) - \frac{1}{3}n^3 - \frac{11}{2}n^2 - \frac{31}{6}n. \quad (6.38)$$

Im Vergleich zu (6.33) reduziert sich die Zahl der multiplikativen Operationen für große Werte von  $n$  und  $N$  tatsächlich auf die Hälfte. Zudem entfallen im Fall der schnellen Givens-Transformationen alle Quadratwurzeln. Für kleine  $n$  und  $N$  überwiegt der Term  $12nN$  so stark, dass die schnelle Version sogar aufwändiger sein kann. Dies trifft zu in den Beispielen 6.3 und 6.4.

### 6.2.2 Spezielle Rechentechniken

Fehlergleichungssysteme aus der Landes- oder Erdvermessung haben die Eigenschaft, dass die Matrix  $\mathbf{C}$  schwach besetzt ist, denn jede Fehlergleichung enthält nur wenige der Unbekannten, und ihre Anzahl ist klein im Verhältnis zu  $n$ . Die Methode der Orthogonalsubstitution in der normalen oder schnellen Version der Givens-Transformation nutzt die schwache Besetzung von  $\mathbf{C}$  sicher einmal dadurch aus, dass Rotationen zu verschwindenden Matrixelementen  $c_{ij}$  unterlassen werden. Dann ist aber auf Grund der Reihenfolge (6.26) offensichtlich, dass die Matrixelemente  $c_{ij}$  der  $i$ -ten Zeile mit  $i > j$ , welche in der gegebenen Matrix  $\mathbf{C}$  gleich null sind und links vom ersten, von null verschiedenen Matrixelement liegen, im Verlauf des Eliminationsprozesses unverändert bleiben. Um diese Tatsache zu verwerten, bezeichnen wir mit

$$f_i(\mathbf{C}) := \min\{j \mid c_{ij} \neq 0, j = 1, 2, \dots, n\}, \quad i = 1, 2, \dots, N, \quad (6.39)$$

den Index des ersten von null verschiedenen Matrixelementes von  $\mathbf{C}$  in der  $i$ -ten Zeile. Da in den ersten  $n$  Zeilen von  $\mathbf{C}$  die reguläre Rechtsdreiecksmatrix  $\mathbf{R}$  entsteht, setzen wir voraus, dass

$$f_i(\mathbf{C}) \leq i \quad \text{für } i = 1, 2, \dots, n \quad (6.40)$$

gilt. Durch geeignete Vertauschungen der Fehlergleichungen und allenfalls durch eine Ummumerierung der Unbekannten kann (6.40) stets erfüllt werden. Unter dieser Voraussetzung ist die orthogonale Transformation von  $\mathbf{C}$  mit denjenigen Matrixelementen  $c_{ij}$  ausführbar, deren Indexpaare  $(i, j)$  der Hülle von  $\mathbf{C}$  angehören, welche wie folgt definiert ist,

$$\text{Env}(\mathbf{C}) := \{(i, j) \mid f_i(\mathbf{C}) \leq j \leq n; i = 1, 2, \dots, N\}. \quad (6.41)$$

Zur ökonomischen Speicherung der relevanten Matrixelemente ist es naheliegend, dieselben zeilenweise in einem eindimensionalen Feld anzugeben, wobei die einzelnen Zeilen je mit

dem ersten, von null verschiedenen Element beginnen. Für die Matrix  $\mathbf{C} \in \mathbb{R}^{6,4}$

$$\mathbf{C} = \left( \begin{array}{cccc} c_{11} & 0 & c_{13} & c_{14} \\ 0 & c_{22} & 0 & c_{24} \\ 0 & c_{32} & c_{33} & 0 \\ 0 & 0 & c_{43} & c_{44} \\ 0 & 0 & c_{53} & c_{54} \\ c_{61} & 0 & 0 & c_{64} \end{array} \right) \quad \text{mit} \quad \begin{aligned} f_1 &= 1 \\ f_2 &= 2 \\ f_3 &= 2 \\ f_4 &= 3 \\ f_5 &= 3 \\ f_6 &= 1 \end{aligned} \quad (6.42)$$

sieht diese Anordnung konkret so aus:

$$\mathbf{C} : \quad \begin{array}{|c|c|c|c|} \hline c_{11} & c_{12} & c_{13} & c_{14} \\ \hline c_{22} & c_{23} & c_{24} & \\ \hline c_{32} & c_{33} & c_{34} & \\ \hline c_{43} & c_{44} & & \\ \hline c_{53} & c_{54} & & \\ \hline c_{61} & c_{62} & c_{63} & c_{64} \\ \hline \end{array} \quad (6.43)$$

Für die in  $\mathbf{C}$  verschwindenden Matrixelemente, deren Indizes der Hülle angehören, sind Plätze vorzusehen, da sie im Verlauf des Rechenprozesses ungleich null werden können. Um den Zugriff zu den Matrixelementen  $c_{ij}$  in der Anordnung (6.43) zu ermöglichen, ist ein Zeigervektor  $\mathbf{z} \in \mathbb{R}^N$  nötig, dessen  $i$ -te Komponente den Platz des letzten Matrixelements  $c_{in}$  der  $i$ -ten Zeile angibt. Für (6.43) lautet er

$$\mathbf{z} = (4, 7, 10, 12, 14, 18)^T.$$

Das Matrixelement  $c_{ij}$  mit  $(i, j) \in \text{Env}(\mathbf{C})$  steht in einer allgemeinen Anordnung der Art (6.43) am Platz  $k$  mit der Zuordnung

$$(i, j) \in \text{Env}(\mathbf{C}) \rightarrow k = z_i + j - n. \quad (6.44)$$

Die Transformation von  $\mathbf{C}$  in die Matrix  $\hat{\mathbf{R}}$  ist ohne Test durchführbar, falls die Matrixelemente  $c_{ij}$  mit  $(i, j) \in \text{Env}(\mathbf{C})$  zeilenweise anstatt spaltenweise eliminiert werden. Man verifiziert analog zum Beweis von Satz 5.7, dass die entsprechende Folge von Rotationen zur zeilenweisen Elimination die gewünschte Transformation leistet. Der benötigte Index des ersten von null verschiedenen Matrixelements der  $i$ -ten Zeile ist gegeben durch

$$f_i(\mathbf{C}) = n - z_i + z_{i-1} + 1, \quad i = 2, 3, \dots, N. \quad (6.45)$$

Mit diesen Angaben kann der Algorithmus (6.34) leicht der neuen Situation angepasst werden. Die Schleifenanweisungen für  $i$  und  $j$  sind zu vertauschen, der Startwert für  $j$  ist durch  $f_i(\mathbf{C})$  (6.45) und die Indizes der Matrixelemente  $c_{ij}$  sind gemäß (6.44) zu ersetzen.

Wir wollen noch eine Variante der Methode der Orthogonaltransformation [Geo 80] beschreiben, bei der nicht die ganze Matrix  $\mathbf{C}$  gespeichert werden muss. Sie minimiert insoweit den Speicherplatzbedarf für Fälle, bei denen entweder die Fehlgleichungen sukzessive gebildet oder von einem externen Speichermedium abgerufen werden.

Um das Vorgehen zu erklären, soll die Transformation von  $\mathbf{C}$  in  $\hat{\mathbf{R}}$  zeilenweise erfolgen, und der Vektor  $\hat{\mathbf{d}}$  soll gleichzeitig aus  $\mathbf{d}$  berechnet werden. Wir untersuchen die Behandlung der  $i$ -ten Fehlgleichung und ihren Beitrag zum schließlich resultierenden Gleichungssystem (6.30). Die erste Fehlgleichung liefert die Startwerte der ersten Zeile von  $\mathbf{R}$  und der ersten Komponente von  $\hat{\mathbf{d}}_1$ . Für  $2 \leq i \leq n$  sind höchstens  $(i-1)$  Rotationen zur Elimination der ersten  $(i-1)$  Elemente  $c_{ij}$ ,  $j = 1, 2, \dots, i-1$ , auszuführen. Die verbleibende transformierte Gleichung ergibt die Startwerte der  $i$ -ten Zeile von  $\mathbf{R}$  und der  $i$ -ten Komponente von  $\hat{\mathbf{d}}_1$ . In den weiteren Fehlgleichungen ( $i > n$ ) sind alle  $n$  Koeffizienten  $c_{ij}$  durch entsprechende Rotationen zu eliminieren mit Hilfe der  $n$  Zeilen der entstehenden Matrix  $\mathbf{R}$ . Da eine solche

$i$ -te Fehlgleichung nach ihrer Bearbeitung unverändert bleibt, wird sie nicht mehr benötigt. Der (transformierte) konstante Term  $\hat{d}_i$  liefert gemäß (6.29) höchstens seinen Beitrag zur Summe der Residuenquadrate. Somit lassen sich die einzelnen Fehlgleichungen unabhängig voneinander bearbeiten, und das Gleichungssystem (6.30) wird sukzessive aufgebaut. Für die Realisierung benötigt man nur den Speicherplatz für die Rechtsdreiecksmatrix  $\mathbf{R}$ , den Vektor  $\hat{\mathbf{d}}_1$  und eine Fehlgleichung. Bei entsprechender Speicherung von  $\mathbf{R}$  beträgt der Speicherbedarf also nur etwa  $S \approx \frac{1}{2}n(n+1) + 2n$  Plätze.

Der Algorithmus (6.46) für diese Methode legt fest, dass für die Matrix  $\mathbf{R} = (r_{ij})$  zur besseren Verständlichkeit die übliche Indizierung verwendet wird, die Komponenten des Vektors  $\hat{\mathbf{d}}$  mit  $d_i$  bezeichnet werden und in der  $i$ -ten Fehlgleichung

$$c_1x_1 + c_2x_2 + \dots + c_nx_n - \tilde{d} = r$$

der Index  $i$  weggelassen wird, da ihre Koeffizienten sukzessive zu definieren sind. Die Rücksubstitution zur Lösung von  $\mathbf{R}\mathbf{x} = \hat{\mathbf{d}}_1$  kann aus (6.34) übernommen werden.

Für  $i = 1, 2, \dots, N$ :

Eingabe von  $c_1, c_2, \dots, c_n, \tilde{d}$

für  $j = 1, 2, \dots, \min(i-1, n)$ :

falls  $c_j \neq 0$ :

falls  $|r_{jj}| < \tau \times |c_j|$ :

$$w = -c_j; \gamma = 0; \sigma = 1$$

sonst

$$w = \operatorname{sgn}(r_{jj}) \times \sqrt{r_{jj}^2 + c_j^2}$$

$$\gamma = r_{jj}/w; \sigma = -c_j/w$$

$$r_{jj} = w$$

für  $k = j+1, j+2, \dots, n$ :

$$h = \gamma \times r_{jk} - \sigma \times c_k$$

$$c_k = \sigma \times r_{jk} + \gamma \times c_k; r_{jk} = h$$

$$h = \gamma \times d_j - \sigma \times \tilde{d}; \tilde{d} = \sigma \times d_j + \gamma \times \tilde{d}; d_j = h$$

falls  $i \leq n$ :

für  $k = i, i+1, \dots, n$ :

$$r_{ik} = c_k$$

$$d_i = \tilde{d}$$

(6.46)

### 6.2.3 Householder-Transformation

Zur orthogonalen Transformation des Fehlgleichungssystems  $\mathbf{C}\mathbf{x} - \mathbf{d} = \mathbf{r}$  in das dazu äquivalente System (6.28)  $\hat{\mathbf{R}}\mathbf{x} - \hat{\mathbf{d}} = \hat{\mathbf{r}}$  werden anstelle der Rotationsmatrizen auch so genannte *Householder-Matrizen* [Hou 58]

$$\mathbf{U} := \mathbf{I} - 2\mathbf{w}\mathbf{w}^T \quad \text{mit} \quad \mathbf{w}^T \mathbf{w} = 1, \quad \mathbf{w} \in \mathbb{R}^N, \quad \mathbf{U} \in \mathbb{R}^{N,N} \quad (6.47)$$

verwendet. Die in (6.47) definierte Matrix  $\mathbf{U}$  ist symmetrisch und *involutorisches*, denn es gilt unter Benutzung der Normierungseigenschaft von  $\mathbf{w}$

$$\mathbf{U}\mathbf{U} = (\mathbf{I} - 2\mathbf{w}\mathbf{w}^T)(\mathbf{I} - 2\mathbf{w}\mathbf{w}^T) = \mathbf{I} - 2\mathbf{w}\mathbf{w}^T - 2\mathbf{w}\mathbf{w}^T + 4\mathbf{w}\mathbf{w}^T\mathbf{w}\mathbf{w}^T = \mathbf{I}.$$

Daraus folgt mit  $\mathbf{U}^T = \mathbf{U}$ , dass  $\mathbf{U}$  orthogonal ist. Die Householder-Matrix  $\mathbf{U}$ , aufgefasst als Darstellung einer linearen Abbildung im  $\mathbb{R}^N$ , entspricht einer *Spiegelung* an einer bestimmten Hyperebene. Um dies einzusehen, sei  $\mathbf{s} \in \mathbb{R}^N$  ein beliebiger Vektor, der orthogonal zu  $\mathbf{w}$  ist. Sein Bildvektor

$$\mathbf{s}' := \mathbf{U}\mathbf{s} = (\mathbf{I} - 2\mathbf{w}\mathbf{w}^T)\mathbf{s} = \mathbf{s} - 2\mathbf{w}(\mathbf{w}^T\mathbf{s})\mathbf{s} = \mathbf{s}$$

ist identisch mit  $\mathbf{s}$ . Der Bildvektor eines Vektors  $\mathbf{z} \in \mathbb{R}^N$ , der mit  $\mathbf{z} = c\mathbf{w}$  proportional zu  $\mathbf{w}$  ist,

$$\mathbf{z}' := \mathbf{U}\mathbf{z} = c\mathbf{U}\mathbf{w} = c(\mathbf{I} - 2\mathbf{w}\mathbf{w}^T)\mathbf{w} = c(\mathbf{w} - 2\mathbf{w}(\mathbf{w}^T\mathbf{w})) = -c\mathbf{w} = -\mathbf{z}$$

ist entgegengesetzt zu  $\mathbf{z}$ . Ein beliebiger Vektor  $\mathbf{x} \in \mathbb{R}^N$  ist eindeutig als Summe  $\mathbf{x} = \mathbf{s} + \mathbf{z}$  eines Vektors  $\mathbf{s}$  und eines Vektors  $\mathbf{z}$  mit den genannten Eigenschaften darstellbar. Für sein Bild gilt deshalb

$$\mathbf{x}' := \mathbf{U}\mathbf{x} = \mathbf{U}(\mathbf{s} + \mathbf{z}) = \mathbf{s} - \mathbf{z},$$

d.h. der Vektor  $\mathbf{x}$  wird an der zu  $\mathbf{w}$  orthogonalen Hyperebene durch den Nullpunkt gespiegelt.

Mit Householder-Matrizen (6.47) können bei entsprechender Wahl des normierten Vektors  $\mathbf{w}$  im Bildvektor  $\mathbf{x}' = \mathbf{U}\mathbf{x}$  eines beliebigen Vektors  $\mathbf{x} \in \mathbb{R}^N$  bestimmte Komponenten gleichzeitig gleich null gemacht werden. Selbstverständlich ist das Quadrat der euklidischen Länge der beiden Vektoren  $\mathbf{x}$  und  $\mathbf{x}'$  gleich. Auf Grund dieser Tatsache wird es möglich sein, mittels einer Folge von  $n$  Transformationsschritten die Matrix  $\mathbf{C} \in \mathbb{R}^{N,n}$  in die gewünschte Form  $\hat{\mathbf{R}}$  (6.24) zu überführen, wobei in jedem Schritt mit Hilfe einer Householder-Matrix eine ganze Spalte behandelt wird.

Im ersten Transformationsschritt soll eine Householder-Matrix  $\mathbf{U}_1 := \mathbf{I} - 2\mathbf{w}_1\mathbf{w}_1^T$  so angewandt werden, dass in der Matrix  $\mathbf{C}' = \mathbf{U}_1\mathbf{C}$  die erste Spalte gleich einem Vielfachen des ersten Einheitsvektors  $\mathbf{e}_1 \in \mathbb{R}^N$  wird. Bezeichnen wir den ersten Spaltenvektor von  $\mathbf{C}$  mit  $\mathbf{c}_1$ , dann soll

$$\mathbf{U}_1\mathbf{c}_1 = \gamma\mathbf{e}_1 \tag{6.48}$$

gelten, wobei für die Konstante  $\gamma$  infolge der Invarianz der euklidischen Länge der Vektoren

$$|\gamma| = \|\mathbf{c}_1\|_2 \tag{6.49}$$

gilt. Da der Bildvektor  $\mathbf{c}'_1 = \gamma\mathbf{e}_1$  aus  $\mathbf{c}_1$  durch Spiegelung an der zu  $\mathbf{w}_1$  orthogonalen Hyperebene hervorgeht, muss der Vektor  $\mathbf{w}_1$  die Richtung der Winkelhalbierenden der Vektoren  $\mathbf{c}_1$  und  $-\mathbf{c}'_1 = -\gamma\mathbf{e}_1$  aufweisen, und somit muss  $\mathbf{w}_1$  proportional zum Vektor  $\mathbf{c}_1 - \gamma\mathbf{e}_1$  sein. Das Vorzeichen von  $\gamma$  ist aber durch (6.49) nicht festgelegt, und so kann als Richtung von  $\mathbf{w}_1$  ebenso gut der Vektor  $\mathbf{c}_1 + \gamma\mathbf{e}_1$  verwendet werden, der zum erstgenannten orthogonal ist.

Der gespiegelte Bildvektor von  $\mathbf{c}_1$  ist in diesem Fall entgegengesetzt zu demjenigen des ersten Falles. Die Freiheit in der Wahl der Richtung von  $\mathbf{w}_1$  wird so ausgenutzt, dass bei

der Berechnung der ersten Komponente des Richtungsvektors

$$\mathbf{h} := \mathbf{c}_1 + \gamma \mathbf{e}_1 \quad (6.50)$$

keine Auslöschung stattfindet. Dies führt zu folgender Festsetzung von  $\gamma$  in (6.50)

$$\gamma := \begin{cases} \|\mathbf{c}_1\|_2 & \text{falls } c_{11} \geq 0 \\ -\|\mathbf{c}_1\|_2 & \text{falls } c_{11} < 0 \end{cases} \quad (6.51)$$

Zur Normierung von  $\mathbf{h}$  zum Vektor  $\mathbf{w}_1$  benötigen wir sein Längenquadrat. Dafür gilt nach (6.50) und wegen (6.49) und (6.51)

$$\begin{aligned} \mathbf{h}^T \mathbf{h} &= \mathbf{c}_1^T \mathbf{c}_1 + 2\gamma \mathbf{c}_1^T \mathbf{e}_1 + \gamma^2 \mathbf{e}_1^T \mathbf{e}_1 = \gamma^2 + 2\gamma c_{11} + \gamma^2 \\ &= 2\gamma(\gamma + c_{11}) =: \beta^2 > 0. \end{aligned} \quad (6.52)$$

Mit der Normierungskonstanten  $\beta > 0$  ergibt sich so der Vektor

$$\mathbf{w}_1 = \mathbf{h}/\beta, \quad (6.53)$$

wobei zu beachten sein wird, dass sich  $\mathbf{h}$  von  $\mathbf{c}_1$  nur in der ersten Komponente unterscheidet. Die Berechnung der Komponenten von  $\mathbf{w}_1$  fassen wir wie folgt zusammen:

$$\begin{aligned} \gamma &= \sqrt{\sum_{i=1}^N c_{i1}^2}; \quad \text{falls } c_{11} < 0 : \gamma = -\gamma \\ \beta &= \sqrt{2\gamma(\gamma + c_{11})} \\ w_1 &= (c_{11} + \gamma)/\beta \\ \text{für } k = 2, 3, \dots, N : \quad w_k &= c_{k1}/\beta \end{aligned} \quad (6.54)$$

Damit ist die Householder-Matrix  $\mathbf{U}_1 = \mathbf{I} - 2\mathbf{w}_1 \mathbf{w}_1^T$  festgelegt, und die Berechnung der im ersten Schritt transformierten Matrix  $\mathbf{C}' = \mathbf{U}_1 \mathbf{C}$  kann erfolgen. Für die erste Spalte von  $\mathbf{C}'$  ergibt sich

$$\begin{aligned} \mathbf{U}_1 \mathbf{c}_1 &= (\mathbf{I} - 2\mathbf{w}_1 \mathbf{w}_1^T) \mathbf{c}_1 = \mathbf{c}_1 - 2\mathbf{w}_1 \mathbf{w}_1^T \mathbf{c}_1 \\ &= \mathbf{c}_1 - 2(\mathbf{c}_1 + \gamma \mathbf{e}_1)(\mathbf{c}_1 + \gamma \mathbf{e}_1)^T \mathbf{c}_1 / \beta^2 \\ &= \mathbf{c}_1 - 2(\mathbf{c}_1 + \gamma \mathbf{e}_1)(\gamma^2 + \gamma c_{11}) / \beta^2 = -\gamma \mathbf{e}_1. \end{aligned} \quad (6.55)$$

Somit gilt für die erste Komponente  $c'_{11} = -\gamma$  und für die übrigen der ersten Spalte  $c'_{k1} = 0$  für  $k = 2, 3, \dots, N$ . Die anderen Elemente der transformierten Matrix  $\mathbf{C}'$  sind gegeben durch

$$\begin{aligned} c'_{ij} &= \sum_{k=1}^N (\delta_{ik} - 2w_i w_k) c_{kj} = c_{ij} - 2w_i \sum_{k=1}^N w_k c_{kj} =: c_{ij} - w_i p_j, \\ i &= 1, 2, \dots, N; \quad j = 2, 3, \dots, n, \end{aligned} \quad (6.56)$$

mit den nur von  $j$  abhängigen Hilfsgrößen

$$p_j := 2 \sum_{k=1}^N w_k c_{kj}, \quad j = 2, 3, \dots, n. \quad (6.57)$$

Die Elemente ungleich null von  $\mathbf{C}'$  berechnen sich damit gemäß

$$\boxed{\begin{aligned} c'_{11} &= -\gamma \\ p_j &= 2 \sum_{k=1}^N w_k c_{kj} \\ c'_{ij} &= c_{ij} - w_i p_j, \quad i = 1, 2, \dots, N, \end{aligned} \quad j = 2, 3, \dots, n.} \quad (6.58)$$

Die Transformation der Matrix  $\mathbf{C}'$  wird mit einer Householder-Matrix  $\mathbf{U}_2 := \mathbf{I} - 2\mathbf{w}_2\mathbf{w}_2^T$  fortgesetzt mit dem Ziel, die Matrixelemente  $c'_{k2}$  mit  $k \geq 3$  zu null zu machen und dabei die erste Spalte unverändert zu lassen. Beide Forderungen erreicht man mit einem Vektor  $\mathbf{w}_2 = (0, w_2, w_3, \dots, w_N)^T$ , dessen erste Komponente gleich null ist. Die Matrix  $\mathbf{U}_2$  hat dann die Gestalt

$$\mathbf{U}_2 = \mathbf{I} - 2\mathbf{w}_2\mathbf{w}_2^T = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 - 2w_2^2 & -2w_2w_3 & \dots & -2w_2w_N \\ 0 & -2w_2w_3 & 1 - 2w_3^2 & \dots & -2w_3w_N \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & -2w_2w_N & -2w_3w_N & \dots & 1 - 2w_N^2 \end{pmatrix}, \quad (6.59)$$

so dass in der Matrix  $\mathbf{C}'' = \mathbf{U}_2\mathbf{C}'$  nicht nur die erste Spalte von  $\mathbf{C}'$  unverändert bleibt sondern auch die erste Zeile. Die letzten  $(N-1)$  Komponenten von  $\mathbf{w}_2$  werden analog zum ersten Schritt aus der Bedingung bestimmt, dass der Teilvektor  $(c'_{22}, c'_{32}, \dots, c'_{N2})^T \in \mathbb{R}^{N-1}$  in ein Vielfaches des Einheitsvektors  $\mathbf{e}_1 \in \mathbb{R}^{N-1}$  transformiert wird. Die sich ändernden Matrixelemente ergeben sich mit entsprechenden Modifikationen analog zu (6.58).

Sind allgemein die ersten  $(l-1)$  Spalten von  $\mathbf{C}$  mit Hilfe von Householder-Matrizen  $\mathbf{U}_1, \mathbf{U}_2, \dots, \mathbf{U}_{l-1}$  auf die gewünschte Form transformiert worden, erfolgt die Behandlung der  $l$ -ten Spalte mit  $\mathbf{U}_l = \mathbf{I} - 2\mathbf{w}_l\mathbf{w}_l^T$ , wo in  $\mathbf{w}_l = (0, 0, \dots, 0, w_l, w_{l+1}, \dots, w_N)^T$  die ersten  $(l-1)$  Komponenten gleich null sind. Im zugehörigen Transformationsschritt bleiben die ersten  $(l-1)$  Spalten und Zeilen der momentanen Matrix  $\mathbf{C}$  deshalb unverändert.

Nach  $n$  Transformationsschritten ist das Ziel (6.28) mit

$$\mathbf{U}_n \mathbf{U}_{n-1} \dots \mathbf{U}_2 \mathbf{U}_1 \mathbf{C} = \hat{\mathbf{R}}, \quad \mathbf{U}_n \mathbf{U}_{n-1} \dots \mathbf{U}_2 \mathbf{U}_1 \mathbf{d} = \hat{\mathbf{d}} \quad (6.60)$$

erreicht. Der gesuchte Lösungsvektor  $\mathbf{x}$  ergibt sich aus dem System (6.30) durch Rücksubstitution.

Ist auch der Residuenvektor  $\mathbf{r}$  gewünscht, so gilt zunächst

$$\mathbf{U}_n \mathbf{U}_{n-1} \dots \mathbf{U}_2 \mathbf{U}_1 \mathbf{r} = \hat{\mathbf{r}} = (0, 0, \dots, 0, \hat{d}_{n+1}, \hat{d}_{n+2}, \dots, \hat{d}_N)^T. \quad (6.61)$$

Infolge der Symmetrie und Orthogonalität der Householder-Matrizen  $\mathbf{U}_l$  folgt daraus

$$\mathbf{r} = \mathbf{U}_1 \mathbf{U}_2 \dots \mathbf{U}_{n-1} \mathbf{U}_n \hat{\mathbf{r}}. \quad (6.62)$$

Der bekannte Residuenvektor  $\hat{\mathbf{r}}$  ist sukzessive mit den Matrizen  $\mathbf{U}_n, \mathbf{U}_{n-1}, \dots, \mathbf{U}_1$  zu multiplizieren. Die typische Multiplikation eines Vektors  $\mathbf{y} \in \mathbb{R}^N$  mit  $\mathbf{U}_l$  erfordert gemäß

$$\mathbf{y}' = \mathbf{U}_l \mathbf{y} = (\mathbf{I} - 2\mathbf{w}_l \mathbf{w}_l^T) \mathbf{y} = \mathbf{y} - 2(\mathbf{w}_l^T \mathbf{y}) \mathbf{w}_l \quad (6.63)$$

die Bildung des Skalarproduktes  $\mathbf{w}_l^T \mathbf{y} =: a$  und die anschließende Subtraktion des  $(2a)$ -fachen des Vektors  $\mathbf{w}_l$  von  $\mathbf{y}$ . Da in  $\mathbf{w}_l$  die ersten  $(l-1)$  Komponenten gleich null sind,

erfordert dieser Schritt  $2(N - l + 1)$  Multiplikationen. Die Rücktransformation von  $\hat{\mathbf{r}}$  in  $\mathbf{r}$  nach (6.62) benötigt also insgesamt  $Z_{\text{Rück}} = n(2N - n + 1)$  Operationen.

Zur tatsächlichen Ausführung der Berechnung von  $\mathbf{r}$  sind die Vektoren  $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_n$  nötig, welche die Householder-Matrizen definieren. Die Komponenten von  $\mathbf{w}_l$  können in der Matrix  $\mathbf{C}$  in der  $l$ -ten Spalte an die Stelle von  $c_{ll}$  und an Stelle der eliminierten Matrixelemente  $c_{il}$  gesetzt werden. Da die Diagonalelemente  $r_{ii} = c_{ii}$  der Rechtsdreiecksmatrix  $\mathbf{R}$  ohnehin bei der Rücksubstitution eine spezielle Rolle spielen, werden sie in einem Vektor  $\mathbf{t} \in \mathbb{R}^n$  gespeichert.

In der algorithmischen Formulierung (6.65) der Householder-Transformation zur Lösung eines Fehlergleichungssystems  $\mathbf{Cx} - \mathbf{d} = \mathbf{r}$  sind die orthogonale Transformation von  $\mathbf{C}$  und dann die sukzessive Berechnung von  $\hat{\mathbf{d}}$ , des Lösungsvektors  $\mathbf{x}$  und des Residuenvektors  $\mathbf{r}$  getrennt nebeneinander dargestellt. Das erlaubt, nacheinander verschiedene Fehlergleichungen mit derselben Matrix  $\mathbf{C}$ , aber unterschiedlichen Vektoren  $\mathbf{d}$  zu lösen. Die von null verschiedenen Komponenten der Vektoren  $\mathbf{w}_l$  werden in den entsprechenden Spalten von  $\mathbf{C}$  gespeichert. Deshalb ist einerseits kein Vektor  $\mathbf{w}$  nötig, andererseits erfahren die Formeln (6.58) eine Modifikation. Weiter ist es nicht nötig, die Werte  $p_j$  in (6.58) zu indizieren. Die Berechnung des Vektors  $\hat{\mathbf{d}}$  aus  $\mathbf{d}$  gemäß (6.60) erfolgt nach den Formeln (6.63). Dasselbe gilt natürlich für den Residuenvektor  $\mathbf{r}$ .

Die Methode der Householder-Transformation löst die Aufgabe mit dem kleinsten Rechenaufwand. Im Algorithmus (6.65) links erfordert der  $l$ -te Schritt  $(N - l + 1) + 3 + (N - l) + 2(n - l)(N - l + 1) = 2(N - l + 1)(n - l + 1) + 2$  multiplikative Operationen und 2 Quadratwurzeln. Summation über  $l$  von 1 bis  $n$  ergibt als Rechenaufwand für die orthogonale Transformation von  $\mathbf{C}$

$$Z_{\text{Householder}} = Nn(n + 1) - \frac{1}{3}n(n^2 - 7)$$

multiplikative Operationen und  $2n$  Quadratwurzeln. Der Aufwand zur Berechnung von  $\hat{\mathbf{d}}$  ist gleich groß wie zur Rücktransformation des Residuenvektors. Deshalb werden im Algorithmus (6.65) rechts

$$2n(2N - n + 1) + \frac{1}{2}n(n + 1) = 4Nn - \frac{3}{2}n^2 + \frac{5}{2}n$$

Multiplikationen benötigt, und der Gesamtaufwand beträgt

$$Z_{\text{FGIHouseholder}} = nN(n + 5) - \frac{1}{3}n^3 - \frac{3}{2}n^2 + \frac{35}{6}n$$

(6.64)

multiplikative Operationen und  $2n$  Quadratwurzeln. Im Vergleich zu (6.33) ist der Aufwand an wesentlichen Operationen tatsächlich nur etwa halb so groß, und die Zahl der Quadratwurzeln ist bedeutend kleiner. Auch im Vergleich zur Methode der schnellen Givens-Transformation ist die Zahl der Operationen günstiger.

Die praktische Durchführung der Householder-Transformation setzt voraus, dass die Matrix  $\mathbf{C}$  gespeichert ist. Die Behandlung von schwach besetzten Fehlergleichungssystemen erfordert aber zusätzliche Überlegungen, um das Auffüllen der Matrix mit Elementen ungleich null, das sog. fill-in, gering zu halten [Duf 76, Gil 76, Gol 80, Gol 96b]. Schließlich ist es mit dieser Methode – im Gegensatz zur Givens-Transformation – nicht möglich die Matrix  $\mathbf{R}$

<i>Die Methode der Householder-Transformation</i>	
Für $l = 1, 2, \dots, n$ :	Für $l = 1, 2, \dots, n$ :
$\gamma = 0$	$s = 0$
für $i = l, l+1, \dots, N$ :	für $k = l, l+1, \dots, N$ :
$\gamma = \gamma + c_{il}^2$	$s = s + c_{kl} \times d_k$
$\gamma = \sqrt{\gamma}$	$s = s + s$
falls $c_{ll} < 0$ : $\gamma = -\gamma$	für $k = l, l+1, \dots, N$ :
$\beta = \sqrt{2 \times \gamma \times (\gamma + c_{ll})}$	$d_k = d_k - s \times c_{kl}$
$t_l = -\gamma$	für $i = n, n-1, \dots, 1$ :
$c_{ll} = (c_{ll} + \gamma) / \beta$	$s = d_i; r_i = 0$
für $k = l+1, l+2, \dots, N$ :	für $k = i+1, i+2, \dots, n$ :
$c_{kl} = c_{kl} / \beta$	$s = s - c_{ik} \times x_k$
für $j = l+1, l+2, \dots, n$ :	$x_i = s / t_i$
$p = 0$	für $i = n+1, n+2, \dots, N$ :
für $k = l, l+1, \dots, N$ :	$r_i = d_i$
$p = p + c_{kl} \times c_{kj}$	für $l = n, n-1, \dots, 1$ :
$p = p + p$	$s = 0$
für $i = l, l+1, \dots, N$ :	für $k = l, l+1, \dots, N$ :
$c_{ij} = c_{ij} - p \times c_{il}$	$s = s + c_{kl} \times r_k$
	$s = s + s$
	für $k = l, l+1, \dots, N$ :
	$r_k = r_k - s \times c_{kl}$

sukzessive durch Bearbeitung der einzelnen Fehlergleichungen aufzubauen.

**Beispiel 6.5.** Die Householder-Transformation liefert für das Fehlergleichungssystem (6.21) bei sechsstelliger Rechnung die transformierte Matrix  $\tilde{\mathbf{C}}$ , welche die Vektoren  $\mathbf{w}_i$  enthält und das Nebendiagonalelement von  $\tilde{\mathbf{R}}$ , den orthogonal transformierten Vektor  $\hat{\mathbf{d}}$  und den Vektor  $\mathbf{t}$  mit den Diagonalelementen von  $\mathbf{R}$ .

$$\tilde{\mathbf{C}} = \left( \begin{array}{cc|c} 0.750260 & -1.68491 & \\ 0.167646 & -0.861177 & \\ 0.251470 & 0.0789376 & \\ 0.301764 & 0.313223 & \\ 0.335293 & 0.365641 & \\ 0.377205 & 0.142624 & \end{array} \right), \quad \hat{\mathbf{d}} = \left( \begin{array}{c} -2.25773 \\ 0.0505840 \\ \hline -0.0684838 \\ -0.0731628 \\ 0.00510470 \\ -0.00616310 \end{array} \right), \quad \mathbf{t} = \left( \begin{array}{c} -1.32508 \\ 0.0486913 \end{array} \right)$$

Rücksubstitution ergibt die Parameterwerte  $\alpha_1 = 0.382867$  und  $\alpha_2 = 1.03887$ , welche nur auf drei Dezimalstellen nach dem Komma richtig sind. Die Abweichungen haben eine Größe, die auf Grund einer Fehleranalyse zu erwarten ist [Kau 79, Law 66].  $\triangle$

### 6.3 Singulärwertzerlegung

Im Satz 6.1 wurde mit (6.24) eine Zerlegung der Matrix  $\mathbf{C} \in \mathbb{R}^{N,n}$  unter der Voraussetzung, dass  $\mathbf{C}$  den Maximalrang  $n < N$  hat, eingeführt, um sie zur Lösung der Fehlergleichungen anzuwenden. Wir werden jetzt eine allgemeinere orthogonale Zerlegung einer Matrix kennenlernen, wobei die Voraussetzung über den Maximalrang fallengelassen wird. Diese Zerlegung gestattet, die Lösungsmenge des Fehlergleichungssystems in dieser allgemeineren Situation zu beschreiben und insbesondere eine Lösung mit einer speziellen Eigenschaft zu charakterisieren und festzulegen. Als Vorbereitung verallgemeinern wir die Aussage des Satzes 6.1, die wir im Hinblick auf ihre Anwendung nur für die betreffende Situation  $\text{Rang}(\mathbf{C}) = r < n < N$  formulieren. Die Aussagen der folgenden Sätze gelten auch für  $r = n$  oder  $n = N$ , wobei dann Matrizenblöcke dementsprechend entfallen können.

**Satz 6.2.** Zu jeder Matrix  $\mathbf{C} \in \mathbb{R}^{N,n}$  mit Rang  $r < n < N$  existieren zwei orthogonale Matrizen  $\mathbf{Q} \in \mathbb{R}^{N,N}$  und  $\mathbf{W} \in \mathbb{R}^{n,n}$  derart, dass

$$\mathbf{Q}^T \mathbf{C} \mathbf{W} = \hat{\mathbf{R}} \quad \text{mit } \hat{\mathbf{R}} = \left( \begin{array}{c|c} \mathbf{R} & \mathbf{0}_1 \\ \hline \mathbf{0}_2 & \mathbf{0}_3 \end{array} \right), \quad \hat{\mathbf{R}} \in \mathbb{R}^{N,n}, \mathbf{R} \in \mathbb{R}^{r,r} \quad (6.66)$$

gilt, wo  $\mathbf{R}$  eine reguläre Rechtsdreiecksmatrix der Ordnung  $r$  und die  $\mathbf{0}_i$ ,  $i = 1, 2, 3$ , Nullmatrizen darstellen mit  $\mathbf{0}_1 \in \mathbb{R}^{r,n-r}$ ,  $\mathbf{0}_2 \in \mathbb{R}^{N-r,r}$ ,  $\mathbf{0}_3 \in \mathbb{R}^{N-r,n-r}$ .

*Beweis.* Es sei  $\mathbf{P} \in \mathbb{R}^{N,N}$  eine Permutationsmatrix, so dass in  $\mathbf{C}' = \mathbf{PC}$  die ersten  $r$  Zeilenvektoren linear unabhängig sind. Wird die Matrix  $\mathbf{C}'$  sukzessive von rechts mit Rotationsmatrizen  $\mathbf{U}(p, q; \varphi) \in \mathbb{R}^{n,n}$  mit den Rotationsindexpaaren

$$(1, 2), (1, 3), \dots, (1, n), (2, 3), (2, 4), \dots, (2, n), \dots, (r, r+1), \dots, (r, n)$$

mit geeignet, analog zu (5.89) bestimmten Drehwinkeln multipliziert, so werden die aktuellen Matrixelemente der ersten  $r$  Zeilen in der Reihenfolge

$$c'_{12}, c'_{13}, \dots, c'_{1n}, c'_{23}, c'_{24}, \dots, c'_{2n}, \dots, c'_{r,r+1}, \dots, c'_{rn}$$

eliminiert. Bezeichnen wir das Produkt dieser Rotationsmatrizen mit  $\mathbf{W} \in \mathbb{R}^{n,n}$ , so besitzt die transformierte Matrix die Gestalt

$$\begin{aligned} \mathbf{C}' \mathbf{W} = \mathbf{PC} \mathbf{W} = \mathbf{C}'' &= \left( \begin{array}{c|c} \mathbf{L} & \mathbf{0}_1 \\ \hline \mathbf{X} & \mathbf{0}_2 \end{array} \right), \\ \mathbf{0}_1 &\in \mathbb{R}^{r,(n-r)}, \quad \mathbf{0}_2 \in \mathbb{R}^{(N-r),(n-r)}, \end{aligned} \quad (6.67)$$

wo  $\mathbf{L} \in \mathbb{R}^{r,r}$  eine reguläre Linksdreiecksmatrix,  $\mathbf{X} \in \mathbb{R}^{(N-r),r}$  eine im Allgemeinen von null verschiedene Matrix und  $\mathbf{0}_1, \mathbf{0}_2$  Nullmatrizen sind. Auf Grund der getroffenen Annahme für  $\mathbf{C}'$  müssen auch in  $\mathbf{C}''$  die ersten  $r$  Zeilen linear unabhängig sein, und folglich ist  $\mathbf{L}$  regulär. Da der Rang von  $\mathbf{C}''$  gleich  $r$  ist, muss notwendigerweise  $\mathbf{0}_2$  eine Nullmatrix sein.

Nach Satz 6.1 existiert zu  $\mathbf{C}''$  eine orthogonale Matrix  $\mathbf{Q}_1 \in \mathbb{R}^{N,N}$ , so dass  $\mathbf{Q}_1^T \mathbf{C}'' = \hat{\mathbf{R}}$  die Eigenschaft (6.66) besitzt. Auf Grund jener konstruktiven Beweisführung genügt es, die ersten  $r$  Spalten zu behandeln, und die Nullelemente in den letzten  $(n-r)$  Spalten von  $\mathbf{C}''$  werden nicht zerstört. Die orthogonale Matrix  $\mathbf{Q}$  ist gegeben durch  $\mathbf{Q}^T = \mathbf{Q}_1^T \mathbf{P}$ .  $\square$

**Satz 6.3.** Zu jeder Matrix  $\mathbf{C} \in \mathbb{R}^{N,n}$  mit Rang  $r \leq n < N$  existieren zwei orthogonale Matrizen  $\mathbf{U} \in \mathbb{R}^{N,N}$  und  $\mathbf{V} \in \mathbb{R}^{n,n}$  derart, dass die Singulärwertzerlegung

$$\mathbf{C} = \mathbf{U} \hat{\mathbf{S}} \mathbf{V}^T \quad \text{mit } \hat{\mathbf{S}} = \begin{pmatrix} \mathbf{S} \\ \mathbf{0} \end{pmatrix}, \quad \hat{\mathbf{S}} \in \mathbb{R}^{N,n}, \mathbf{S} \in \mathbb{R}^{n,n} \quad (6.68)$$

gilt, wo  $\mathbf{S}$  eine Diagonalmatrix mit nichtnegativen Diagonalelementen  $s_i$  ist, die eine nicht-zunehmende Folge mit  $s_1 \geq s_2 \geq \dots \geq s_r > s_{r+1} = \dots = s_n = 0$  bilden, und  $\mathbf{0}$  eine Nullmatrix darstellt.

*Beweis.* Wir betrachten zuerst den Fall  $r = n$ , um anschließend die allgemeinere Situation  $r < n$  zu behandeln. Für  $r = n$  ist die Matrix  $\mathbf{A} := \mathbf{C}^T \mathbf{C} \in \mathbb{R}^{n,n}$  symmetrisch und positiv definit. Ihre reellen und positiven Eigenwerte  $s_i^2$  seien in nicht zunehmender Reihenfolge  $s_1^2 \geq s_2^2 \geq \dots \geq s_n^2 > 0$  indiziert. Nach dem Hauptachsensatz existiert eine orthogonale Matrix  $\mathbf{V} \in \mathbb{R}^{n,n}$ , so dass

$$\mathbf{V}^T \mathbf{A} \mathbf{V} = \mathbf{V}^T \mathbf{C}^T \mathbf{C} \mathbf{V} = \mathbf{D} \text{ mit } \mathbf{D} = \text{diag}(s_1^2, s_2^2, \dots, s_n^2) \quad (6.69)$$

gilt. Weiter sei  $\mathbf{S}$  die reguläre Diagonalmatrix mit den positiven Werten  $s_i$  in der Diagonale. Dann definieren wir die Matrix

$$\hat{\mathbf{U}} := \mathbf{C} \mathbf{V} \mathbf{S}^{-1} \in \mathbb{R}^{N,n}, \quad (6.70)$$

die unter Berücksichtigung von (6.69) wegen  $\hat{\mathbf{U}}^T \hat{\mathbf{U}} = \mathbf{S}^{-1} \mathbf{V}^T \mathbf{C}^T \mathbf{C} \mathbf{V} \mathbf{S}^{-1} = \mathbf{I}_n$  orthonormierte Spaltenvektoren enthält. Im  $\mathbb{R}^N$  lassen sie sich zu einer orthonormierten Basis ergänzen, und so können wir  $\hat{\mathbf{U}}$  zu einer orthogonalen Matrix  $\mathbf{U} := (\hat{\mathbf{U}}, \mathbf{Y}) \in \mathbb{R}^{N,N}$  erweitern, wobei  $\mathbf{Y}^T \hat{\mathbf{U}} = \mathbf{Y}^T \mathbf{C} \mathbf{V} \mathbf{S}^{-1} = \mathbf{0}$  ist. In diesem Fall erhalten wir mit

$$\mathbf{U}^T \mathbf{C} \mathbf{V} = \left( \frac{\mathbf{S}^{-1} \mathbf{V}^T \mathbf{C}^T}{\mathbf{Y}^T} \right) \mathbf{C} \mathbf{V} = \left( \frac{\mathbf{S}^{-1} \mathbf{V}^T \mathbf{C}^T \mathbf{C} \mathbf{V}}{\mathbf{Y}^T \mathbf{C} \mathbf{V}} \right) = \left( \frac{\mathbf{S}}{\mathbf{0}} \right) = \hat{\mathbf{S}} \quad (6.71)$$

die Aussage (6.68). Dieses Teilresultat wenden wir an, um (6.68) im Fall  $r < n$  zu zeigen. Nach Satz 6.2 existieren orthogonale Matrizen  $\mathbf{Q}$  und  $\mathbf{W}$ , so dass  $\mathbf{Q}^T \mathbf{C} \mathbf{W} = \hat{\mathbf{R}}$  gilt mit der Matrix  $\hat{\mathbf{R}}$  gemäß (6.66). Die Teilmatrix  $\mathbf{R}$  von  $\hat{\mathbf{R}}$ , gebildet aus den ersten  $r$  Spalten, hat den Maximalrang  $r$ , und folglich existieren zwei orthogonale Matrizen  $\tilde{\mathbf{U}}$  und  $\tilde{\mathbf{V}}$ , so dass

$$\begin{aligned} \left( \frac{\mathbf{R}}{\mathbf{0}} \right) &= \tilde{\mathbf{U}} \tilde{\mathbf{S}} \tilde{\mathbf{V}}^T = \tilde{\mathbf{U}} \left( \frac{\mathbf{S}_1}{\mathbf{0}} \right) \tilde{\mathbf{V}}^T, \\ \tilde{\mathbf{U}} \in \mathbb{R}^{N,N}, \tilde{\mathbf{V}} \in \mathbb{R}^{r,r}, \tilde{\mathbf{S}} \in \mathbb{R}^{N,r}, \mathbf{S}_1 \in \mathbb{R}^{r,r} \end{aligned}$$

gilt. Die Matrix  $\tilde{\mathbf{S}}$  erweitern wir durch  $(n - r)$  Nullvektoren und die Matrix  $\tilde{\mathbf{V}}$  zu einer orthogonalen Matrix gemäß

$$\hat{\mathbf{S}} := \left( \begin{array}{c|c} \mathbf{S}_1 & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array} \right) \in \mathbb{R}^{N,n}, \quad \hat{\mathbf{V}} := \left( \begin{array}{c|c} \tilde{\mathbf{V}} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I}_{n-r} \end{array} \right) \in \mathbb{R}^{n,n},$$

wo  $\mathbf{I}_{n-r}$  die  $(n - r)$ -reihige Einheitsmatrix darstellt. Mit den orthogonalen Matrizen

$$\mathbf{U} := \mathbf{Q}\tilde{\mathbf{U}} \in \mathbb{R}^{N,N} \quad \text{und} \quad \mathbf{V} := \mathbf{W}\hat{\mathbf{V}} \in \mathbb{R}^{n,n}$$

ergibt sich

$$\begin{aligned} \mathbf{U}^T \mathbf{C} \mathbf{V} = \tilde{\mathbf{U}}^T \mathbf{Q}^T \mathbf{C} \mathbf{W} \hat{\mathbf{V}} &= \tilde{\mathbf{U}}^T \left( \begin{array}{c|c} \mathbf{R} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array} \right) \left( \begin{array}{c|c} \tilde{\mathbf{V}} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array} \right) \\ &= \left( \begin{array}{c|c} \mathbf{S}_1 & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array} \right) = \hat{\mathbf{S}}. \end{aligned} \tag{6.72}$$

Dies ist im Wesentlichen die Behauptung (6.68), die aus (6.72) durch eine entsprechende Partitionierung von  $\hat{\mathbf{S}}$  hervorgeht.  $\square$

Die  $s_i$  heißen die *singulären Werte* der Matrix  $\mathbf{C}$ . Bezeichnen wir weiter mit  $\mathbf{u}_i \in \mathbb{R}^N$  und  $\mathbf{v}_i \in \mathbb{R}^n$  die Spaltenvektoren von  $\mathbf{U}$  und  $\mathbf{V}$ , so folgen aus (6.68) die Relationen

$$\mathbf{C}\mathbf{v}_i = s_i \mathbf{u}_i \quad \text{und} \quad \mathbf{C}^T \mathbf{u}_i = s_i \mathbf{v}_i, \quad i = 1, 2, \dots, n. \tag{6.73}$$

Die  $\mathbf{v}_i$  heißen *Rechtssingulärvektoren* und die  $\mathbf{u}_i$  *Linkssingulärvektoren* der Matrix  $\mathbf{C}$ .

Die Singulärwertzerlegung (6.68) eröffnet eine weitere Möglichkeit, ein System von Fehlergleichungen  $\mathbf{C}\mathbf{x} - \mathbf{d} = \mathbf{r}$  durch ein orthogonal transformiertes, äquivalentes System zu ersetzen. Mit  $\mathbf{V}\mathbf{V}^T = \mathbf{I}$  können wir schreiben

$$\mathbf{U}^T \mathbf{C} \mathbf{V} \mathbf{V}^T \mathbf{x} - \mathbf{U}^T \mathbf{d} = \mathbf{U}^T \mathbf{r} = \hat{\mathbf{r}}. \tag{6.74}$$

Dann führen wir die Hilfsvektoren

$$\mathbf{y} := \mathbf{V}^T \mathbf{x} \in \mathbb{R}^n, \quad \mathbf{b} := \mathbf{U}^T \mathbf{d} \in \mathbb{R}^N \quad \text{mit } b_i = \mathbf{u}_i^T \mathbf{d} \tag{6.75}$$

ein. Dann lautet aber (6.74) auf Grund der Singulärwertzerlegung sehr speziell

$$\begin{aligned} s_i y_i - b_i &= \hat{r}_i, \quad i = 1, 2, \dots, r, \\ -b_i &= \hat{r}_i, \quad i = r + 1, r + 2, \dots, N. \end{aligned} \tag{6.76}$$

Da die letzten  $(N - r)$  Residuen  $\hat{r}_i$  unabhängig von den (neuen) Unbekannten sind, ist die Summe der Quadrate der Residuen genau dann minimal, falls  $\hat{r}_i = 0$ ,  $i = 1, 2, \dots, r$ , gilt, und sie hat folglich den eindeutigen Wert

$$\varrho_{\min} := \mathbf{r}^T \mathbf{r} = \sum_{i=r+1}^N \hat{r}_i^2 = \sum_{i=r+1}^N b_i^2 = \sum_{i=r+1}^N (\mathbf{u}_i^T \mathbf{d})^2. \tag{6.77}$$

Die ersten  $r$  Unbekannten  $y_i$  sind nach (6.76) gegeben durch

$$y_i = b_i/s_i, \quad i = 1, 2, \dots, r, \quad (6.78)$$

während die restlichen  $(n - r)$  Unbekannten frei wählbar sind. Der Lösungsvektor  $\mathbf{x}$  der Fehlerequationen besitzt nach (6.75) somit die Darstellung

$$\mathbf{x} = \sum_{i=1}^r \frac{\mathbf{u}_i^T \mathbf{d}}{s_i} \mathbf{v}_i + \sum_{i=r+1}^n y_i \mathbf{v}_i \quad (6.79)$$

mit den  $(n - r)$  freien Parametern  $y_{r+1}, y_{r+2}, \dots, y_n$ . Hat die Matrix  $\mathbf{C}$  nicht Maximalrang, ist die *allgemeine Lösung* als Summe einer partikulären Lösung im Unterraum der  $r$  Rechts-singulärvektoren  $\mathbf{v}_i$  zu den *positiven* singulären Werten  $s_i$  und einem beliebigen Vektor aus dem Nullraum der Matrix  $\mathbf{C}$  darstellbar. Denn nach (6.73) gilt für die verschwindenden singulären Werte  $\mathbf{C}\mathbf{v}_i = \mathbf{0}$ ,  $i = r+1, r+2, \dots, n$ .

In der Lösungsmenge des Fehlerequationssystems existiert eine spezielle Lösung  $\mathbf{x}^*$  mit minimaler euklidischer Norm. Infolge der Orthogonalität der Rechtssingulärvektoren  $\mathbf{v}_i$  ist sie durch  $y_{r+1} = y_{r+2} = \dots = y_n = 0$  gekennzeichnet und ist gegeben durch

$$\boxed{\mathbf{x}^* = \sum_{i=1}^r \frac{\mathbf{u}_i^T \mathbf{d}}{s_i} \mathbf{v}_i, \quad \|\mathbf{x}^*\|_2 = \min_{\mathbf{C}\mathbf{x}-\mathbf{d}=r} \|\mathbf{x}\|_2.} \quad (6.80)$$

Die Singulärwertzerlegung der Matrix  $\mathbf{C}$  liefert einen wesentlichen Einblick in den Aufbau der allgemeinen Lösung  $\mathbf{x}$  (6.79) oder der speziellen Lösung  $\mathbf{x}^*$ , der für die problemgerechte Behandlung von heiklen Fehlerequationen wegweisend sein kann. In statistischen Anwendungen treten häufig Fehlerequationen auf, deren Normalgleichungen eine extrem schlechte Kondition haben. Da ja im Fall  $r = n$  die singulären Werte  $s_i$  die Quadratwurzeln der Eigenwerte der Normalgleichungsmatrix  $\mathbf{A} = \mathbf{C}^T \mathbf{C}$  sind, existieren sehr kleine singuläre Werte. Aber auch im Fall  $r < n$  stellt man oft fest, dass sehr kleine positive singuläre Werte auftreten. Da sie in (6.80) im Nenner stehen, können die kleinsten der positiven singulären Werte sehr große und eventuell unerwünschte Beiträge zum Lösungsvektor  $\mathbf{x}^*$  bewirken. Anstelle von (6.80) kann es deshalb sinnvoll sein, die Folge von Vektoren

$$\boxed{\mathbf{x}^{(k)} := \sum_{i=1}^k \frac{\mathbf{u}_i^T \mathbf{d}}{s_i} \mathbf{v}_i, \quad k = 1, 2, \dots, r,} \quad (6.81)$$

zu betrachten. Man erhält sie formal mit  $y_{k+1} = \dots = y_r = 0$ , so dass wegen (6.76) die zugehörigen Summen der Quadrate der Residuen

$$\varrho^{(k)} := \sum_{i=k+1}^N \hat{r}_i^2 = \varrho_{\min} + \sum_{i=k+1}^r b_i^2 = \varrho_{\min} + \sum_{i=k+1}^r (\mathbf{u}_i^T \mathbf{d})^2 \quad (6.82)$$

mit zunehmendem  $k$  eine monotone, nichtzunehmende Folge bildet mit  $\varrho^{(r)} = \varrho_{\min}$ . Die euklidische Norm  $\|\mathbf{x}^{(k)}\|_2$  hingegen nimmt mit wachsendem  $k$  zu. Die Vektoren  $\mathbf{x}^{(k)}$  können deshalb als Näherungen für  $\mathbf{x}^*$  betrachtet werden. Je nach Aufgabenstellung oder Zielsetzung ist entweder jene Näherung  $\mathbf{x}^{(k)}$  problemgerecht, für welche  $\varrho^{(k)} - \varrho_{\min}$  eine vorgegebene Schranke nicht übersteigt, oder in welcher alle Anteile weggelassen sind, die zu singulären

Werten gehören, die eine Schranke unterschreiten [Gol 96b, Law 95]. Dies stellt auch eine Möglichkeit zur Regularisierung inkorrekt gestellter Probleme dar. Eine andere Möglichkeit haben wir in (4.67) kennen gelernt.

Das Rechenverfahren ist bei bekannter Singulärwertzerlegung von  $\mathbf{C} = \mathbf{U}\hat{\mathbf{S}}\mathbf{V}^T$  (6.68) durch (6.80) oder (6.81) vorgezeichnet. Die algorithmische Durchführung der Singulärwertzerlegung kann hier nicht im Detail entwickelt werden. Sie besteht im Wesentlichen aus zwei Schritten. Zuerst wird die Matrix  $\mathbf{C}$  in Analogie zu (6.66) mit zwei orthogonalen Matrizen  $\mathbf{Q}$  und  $\mathbf{W}$  so transformiert, dass  $\mathbf{R}$  nur in der Diagonale und in der Nebendiagonale von null verschiedene Elemente aufweist, also eine *bidiagonale Matrix* ist. Die Matrix  $\hat{\mathbf{R}}$  wird dann mit einer Variante des QR-Algorithmus weiter iterativ in die Matrix  $\hat{\mathbf{S}}$  übergeführt [Cha 82, Gol 65, Gol 96b, Kie 88, Law 95].

**Beispiel 6.6.** Die Matrix  $\mathbf{C} \in \mathbb{R}^{7,3}$  in (6.14) besitzt eine Singulärwertzerlegung in den Matrizen

$$\mathbf{U} \doteq \begin{pmatrix} 0.103519 & -0.528021 & -0.705006 & 0.089676 & 0.307578 & 0.171692 & -0.285166 \\ 0.149237 & -0.485300 & -0.074075 & -0.259334 & -0.694518 & -0.430083 & 0.046303 \\ 0.193350 & -0.426606 & 0.213222 & -0.003650 & 0.430267 & -0.101781 & 0.734614 \\ 0.257649 & -0.328640 & 0.403629 & 0.513837 & -0.295124 & 0.544000 & -0.125034 \\ 0.368194 & -0.143158 & 0.417248 & -0.597155 & 0.297387 & 0.046476 & -0.471858 \\ 0.551347 & 0.187483 & 0.010556 & 0.496511 & 0.152325 & -0.589797 & -0.207770 \\ 0.650918 & 0.374216 & -0.338957 & -0.239885 & -0.197914 & 0.359493 & 0.308912 \end{pmatrix}$$

und

$$\mathbf{V} \doteq \begin{pmatrix} 0.474170 & -0.845773 & -0.244605 \\ 0.530047 & 0.052392 & 0.846348 \\ 0.703003 & 0.530965 & -0.473142 \end{pmatrix}$$

und den singulären Werten  $s_1 \doteq 4.796200$ ,  $s_2 \doteq 1.596202$ ,  $s_3 \doteq 0.300009$ . Der Vektor  $\mathbf{b}$  (6.75) ist  $\mathbf{b} \doteq (-10.02046, -0.542841, 2.509634, 0.019345, -0.128170, -0.353416, 0.040857)^T$ , so dass sich mit  $y_1 \doteq 2.089251$ ,  $y_2 \doteq 0.340083$ ,  $y_3 \doteq -8.365203$  die Komponenten des Lösungsvektors aus den Spalten von  $\mathbf{V}$  berechnen lassen. Ihre Werte sind  $\alpha_0 \doteq 2.749198$ ,  $\alpha_1 \doteq -5.954657$  und  $\alpha_2 \doteq 5.607247$ .  $\triangle$

## 6.4 Nichtlineare Ausgleichsprobleme

Zur Behandlung von überbestimmten Systemen nichtlinearer Gleichungen nach der Methode der kleinsten Quadrate existieren zwei grundlegend verschiedene Verfahren, deren Prinzip dargestellt werden wird. Zahlreiche Varianten sind für spezielle Problemstellungen daraus entwickelt worden.

### 6.4.1 Gauß-Newton-Methode

Wir betrachten das überbestimmte System von  $N$  nichtlinearen Gleichungen zur Bestimmung der  $n$  Unbekannten  $x_1, x_2, \dots, x_n$  aus den beobachteten  $N$  Messwerten  $l_1, l_2, \dots, l_N$

$$\boxed{f_i(x_1, x_2, \dots, x_n) - l_i = r_i, \quad i = 1, 2, \dots, N,} \quad (6.83)$$

wo  $r_i$  wieder die Residuen-Werte sind. In (6.83) ist angenommen, dass die in den Unbekannten  $x_j$  nichtlinearen Funktionen  $f_i$  vom Index  $i$  der Fehlergleichung abhängig seien, obwohl dies in vielen Fällen nicht zutrifft.

Die notwendigen Bedingungen zur Minimierung der Funktion

$$F(\mathbf{x}) := \mathbf{r}^T \mathbf{r} = \sum_{i=1}^N [f_i(x_1, x_2, \dots, x_n) - l_i]^2 \quad (6.84)$$

sind für  $j = 1, 2, \dots, n$

$$\frac{1}{2} \frac{\partial F(\mathbf{x})}{\partial x_j} = \sum_{i=1}^N [f_i(x_1, x_2, \dots, x_n) - l_i] \frac{\partial f_i(x_1, x_2, \dots, x_n)}{\partial x_j} = 0. \quad (6.85)$$

Sie ergeben ein System von  $n$  nichtlinearen Gleichungen für die Unbekannten  $x_1, x_2, \dots, x_n$ . Seine analytische Lösung ist meistens unmöglich und seine numerische Lösung aufwändig.

Deshalb werden die nichtlinearen Fehlergleichungen (6.83) zuerst *linearisiert*. Wir nehmen an, für die gesuchten Werte der Unbekannten seien Näherungen  $x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}$  geeignet vorgegeben. Dann verwenden wir den *Korrekturansatz*

$$x_j = x_j^{(0)} + \xi_j, \quad j = 1, 2, \dots, n, \quad (6.86)$$

so dass die  $i$ -te Fehlergleichung von (6.83) im Sinn einer Approximation ersetzt werden kann durch

$$\sum_{j=1}^n \frac{\partial f_i(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})}{\partial x_j} \xi_j + f_i(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}) - l_i = \varrho_i^{(0)}. \quad (6.87)$$

Da in den linearisierten Fehlergleichungen andere Residuenwerte auftreten, bezeichnen wir sie mit  $\varrho_i$ . Wir definieren die Größen

$$c_{ij}^{(0)} := \frac{\partial f_i(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})}{\partial x_j}, \quad d_i^{(0)} := l_i - f_i(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}), \quad (6.88)$$

$$i = 1, 2, \dots, N; \quad j = 1, 2, \dots, n,$$

so dass (6.87) mit  $\mathbf{C}^{(0)} = (c_{ij}^{(0)}) \in \mathbb{R}^{N,n}$ ,  $\mathbf{d}^{(0)} = (d_1^{(0)}, d_2^{(0)}, \dots, d_N^{(0)})^T$  ein lineares Fehlergleichungssystem  $\mathbf{C}^{(0)} \boldsymbol{\xi} - \mathbf{d}^{(0)} = \boldsymbol{\varrho}^{(0)}$  für den Korrekturvektor  $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_n)^T$  darstellt. Dieser kann mit den Verfahren von Abschnitt 6.2 oder 6.3 bestimmt werden. Der Korrekturvektor  $\boldsymbol{\xi}^{(1)}$  als kleinste-Quadrat-Lösung des linearisierten Fehlergleichungssystems (6.87) kann im Allgemeinen nach (6.86) nicht zu der Lösung des nichtlinearen Fehlergleichungssystems (6.83) führen. Vielmehr stellen die Werte

$$x_j^{(1)} := x_j^{(0)} + \xi_j^{(1)}, \quad j = 1, 2, \dots, n, \quad (6.89)$$

im günstigen Fall bessere Näherungen für die Unbekannten  $x_j$  dar, die iterativ weiter verbessert werden können. Das Iterationsverfahren bezeichnet man als *Gauß-Newton-Methode*, da die Korrektur  $\xi^{(1)}$  aus (6.87) nach dem Gaußschen Prinzip ermittelt wird und sich die Fehlergleichungen (6.87) im Sonderfall  $N = n$  auf die linearen Gleichungen reduzieren, die in der Methode von Newton zur Lösung von nichtlinearen Gleichungen auftreten. Die Matrix  $\mathbf{C}^{(0)}$ , deren Elemente in (6.88) erklärt sind, ist die *Jacobi-Matrix* der Funktionen  $f_i(x_1, x_2, \dots, x_n)$  am Punkt  $\mathbf{x}^{(0)} \in \mathbb{R}^n$ .

**Beispiel 6.7.** Zur Bestimmung der Abmessungen einer Pyramide mit quadratischem Grundriss sind die Seite  $a$  der Grundfläche, ihre Diagonale  $d$ , die Höhe  $H$ , die Pyramidenkante  $s$  und die Höhe  $h$  einer Seitenfläche gemessen worden, siehe Abb. 6.2. Die Messwerte sind (in Längeneinheiten)  $a = 2.8$ ,  $d = 4.0$ ,  $H = 4.5$ ,  $s = 5.0$  und  $h = 4.7$ . Die Unbekannten des Problems sind die Länge  $x_1$  der Grundkante und die Höhe  $x_2$  der Pyramide. Das System von fünf teilweise nichtlinearen Fehlergleichungen lautet hier

$$\begin{aligned} x_1 - a &= r_1, & f_1(x_1, x_2) &:= x_1 \\ \sqrt{2}x_1 - d &= r_2, & f_2(x_1, x_2) &:= \sqrt{2}x_1 \\ x_2 - H &= r_3, & f_3(x_1, x_2) &:= x_2 \\ \sqrt{\frac{1}{2}x_1^2 + x_2^2} - s &= r_4, & f_4(x_1, x_2) &:= \sqrt{\frac{1}{2}x_1^2 + x_2^2} \\ \sqrt{\frac{1}{4}x_1^2 + x_2^2} - h &= r_5, & f_5(x_1, x_2) &:= \sqrt{\frac{1}{4}x_1^2 + x_2^2} \end{aligned} \quad (6.90)$$

Die Messwerte  $a$  und  $H$  stellen brauchbare Näherungen der Unbekannten dar, und wir setzen  $x_1^{(0)} = 2.8$ ,  $x_2^{(0)} = 4.5$ . Mit diesen Startwerten erhalten wir bei sechsstelliger Rechnung

$$\mathbf{C}^{(0)} \doteq \begin{pmatrix} 1.00000 & 0 \\ 1.41421 & 0 \\ 0 & 1.00000 \\ 0.284767 & 0.915322 \\ 0.148533 & 0.954857 \end{pmatrix}, \quad \mathbf{d}^{(0)} \doteq \begin{pmatrix} 0 \\ 0.04021 \\ 0 \\ 0.08370 \\ -0.01275 \end{pmatrix} = -\mathbf{r}^{(0)},$$

daraus mit der Methode von Householder den Korrekturvektor  $\xi^{(1)} \doteq (0.0227890, 0.0201000)^T$  und die Näherungen  $x_1^{(1)} \doteq 2.82279$ ,  $x_2^{(1)} \doteq 4.52010$ . Mit diesen Werten resultieren die Matrix  $\mathbf{C}^{(1)}$  und

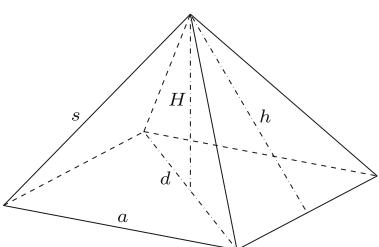


Abb. 6.2  
Pyramide.

der Konstantenvektor  $\mathbf{d}^{(1)}$

$$\mathbf{C}^{(1)} \doteq \begin{pmatrix} 1.00000 & 0 \\ 1.41421 & 0 \\ 0 & 1.00000 \\ 0.285640 & 0.914780 \\ 0.149028 & 0.954548 \end{pmatrix}, \quad \mathbf{d}^{(1)} \doteq \begin{pmatrix} -0.02279 \\ 0.00798 \\ -0.02010 \\ 0.05881 \\ -0.03533 \end{pmatrix} = -\mathbf{r}^{(1)}.$$

Der resultierende Korrekturvektor ist  $\boldsymbol{\xi}^{(2)} \doteq (0.00001073, -0.00001090)^T$ , so dass  $x_1^{(2)} \doteq 2.82280$ ,  $x_2^{(2)} \doteq 4.52009$  sind. Die begonnene Iteration wird solange fortgesetzt, bis eine Norm des Korrekturvektors  $\boldsymbol{\xi}^{(k)}$  genügend klein ist. Wegen der raschen Konvergenz bringt der nächste Schritt in diesem Beispiel bei der verwendeten Rechengenauigkeit keine Änderung der Näherungslösung. Die Vektoren  $-\mathbf{d}^{(k)}$  sind gleich den Residuenvektoren des nichtlinearen Fehlergleichungssystems (6.90). Ihre euklidischen Normen  $\|\mathbf{r}^{(0)}\| \doteq 9.373 \cdot 10^{-2}$ ,  $\|\mathbf{r}^{(1)}\| \doteq 7.546 \cdot 10^{-2}$  und  $\|\mathbf{r}^{(2)}\| \doteq 7.546 \cdot 10^{-2}$  nehmen nur innerhalb der angegebenen Ziffern monoton ab.  $\triangle$

**Beispiel 6.8.** Die Standortbestimmung eines Schiffes kann beispielsweise durch Radiopeilung erfolgen, indem die Richtungen zu bekannten Sendestationen ermittelt werden. Zur Vereinfachung der Aufgabenstellung wollen wir annehmen, dass die Erdkrümmung nicht zu berücksichtigen und dass eine feste Richtung bekannt sei. In einem rechtwinkligen Koordinatensystem sind somit die unbekannten Koordinaten  $x$  und  $y$  eines Punktes  $P$  zu bestimmen, falls mehrere Winkel  $\alpha_i$  gemessen werden, unter denen Sender  $S_i$  mit bekannten Koordinaten  $(x_i, y_i)$  angepeilt werden (vgl. Abb. 6.3).

Die  $i$ -te Fehlergleichung für die Unbekannten  $x$  und  $y$  lautet

$$\arctan\left(\frac{y - y_i}{x - x_i}\right) - \alpha_i = r_i.$$

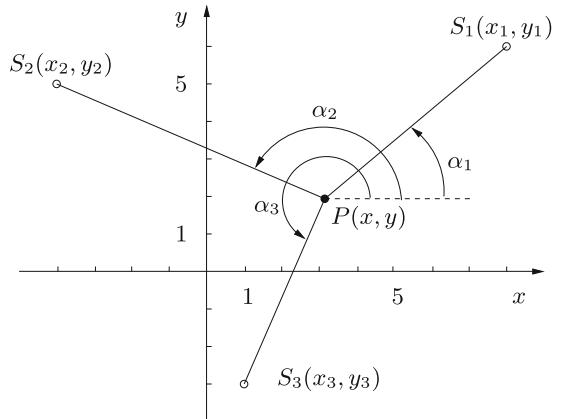


Abb. 6.3 Ortsbestimmung durch Radiopeilung.

Für die Linearisierung ist zu beachten, dass die Winkel im Bogenmaß zu verstehen sind, und dass sie zudem auf das Intervall des Hauptwertes der arctan-Funktion reduziert werden. Mit den Näherungen  $x^{(k)}, y^{(k)}$  lautet die  $i$ -te linearisierte Fehlergleichung für die Korrekturen  $\xi$  und  $\eta$

$$\frac{-(y^{(k)} - y_i)}{(x^{(k)} - x_i)^2 + (y^{(k)} - y_i)^2} \xi + \frac{x^{(k)} - x_i}{(x^{(k)} - x_i)^2 + (y^{(k)} - y_i)^2} \eta + \arctan\left(\frac{y^{(k)} - y_i}{x^{(k)} - x_i}\right) - \alpha_i = \varrho_i$$

Die Daten, die der Abb. 6.3 zu Grunde liegen, sind

$i$	$x_i$	$y_i$	$\alpha_1$	Hauptwert
1	8	6	$42^\circ$	0.733038
2	-4	5	$158^\circ$	-0.383972
3	1	-3	$248^\circ$	1.18682

Für die geschätzten Standortkoordinaten  $x^{(0)} = 3, y^{(0)} = 2$  resultieren die Matrix  $C^{(0)}$  und der Vektor  $d^{(0)}$  bei sechsstelliger Rechnung

$$C^{(0)} \doteq \begin{pmatrix} 0.0975610 & -0.121951 \\ 0.0517241 & 0.120690 \\ -0.172414 & 0.0689655 \end{pmatrix}, \quad d^{(0)} \doteq \begin{pmatrix} 0.058297 \\ 0.020919 \\ -0.003470 \end{pmatrix}.$$

Der weitere Verlauf der Iteration ist in der folgenden Tabelle dargestellt.

$k$	$x^{(k)}$	$y^{(k)}$	$r^{(k)T} r^{(k)}$	$\xi^{(k+1)}$	$\eta^{(k+1)}$
0	3.00000	2.00000	$3.84819 \cdot 10^{-3}$	0.148633	-0.0648092
1	3.14863	1.93519	$2.42227 \cdot 10^{-3}$	0.00721119	0.00834256
2	3.15584	1.94353	$2.42007 \cdot 10^{-3}$	0.00021494	-0.00014632
3	3.15605	1.94338	$2.42029 \cdot 10^{-3}$	0.00000367	0.00001563
4	3.15605	1.94340			

△

## 6.4.2 Minimierungsverfahren

Die mit der Gauß-Newton-Methode gebildete Folge der Vektoren  $\mathbf{x}^{(k)}$  braucht bei ungeeigneter Wahl des Startvektors  $\mathbf{x}^{(0)}$  oder bei kritischen Ausgleichsproblemen nicht gegen die gesuchte Lösung  $\mathbf{x}$  des nichtlinearen Fehlergleichungssystems zu konvergieren. Um stets eine gegen  $\mathbf{x}$  konvergente Folge von Vektoren  $\mathbf{x}^{(k)}$  zu konstruieren, soll sie, geleitet durch das Gaußsche Prinzip, die Eigenschaft haben, dass die Summe der Quadrate der Residuen  $F(\mathbf{x}) = \mathbf{r}^T \mathbf{r}$  (6.84) die Bedingung

$$F(\mathbf{x}^{(k)}) < F(\mathbf{x}^{(k-1)}), \quad (k = 1, 2, \dots) \quad (6.91)$$

erfüllt. Dies ist die Forderung eines Minimierungsverfahrens und bedeutet, das Minimum einer Funktion  $F(\mathbf{x})$  aufzufinden. Dazu muss eine so genannte *Abstiegsrichtung*  $\mathbf{v}^{(k)}$  bekannt sein, für welche positive Werte  $t$  existieren, so dass mit

$$\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} + t\mathbf{v}^{(k)}, \quad t > 0 \quad (6.92)$$

die Bedingung (6.91) erfüllt ist. Eine Abstiegsrichtung  $\mathbf{v}^{(k)}$  stellt der negative Gradient der Funktion  $F(\mathbf{x})$  im Punkt  $\mathbf{x}^{(k-1)}$  dar. Nach (6.85), (6.83) und (6.88) ist diese Richtung berechenbar als

$$\mathbf{v}^{(k)} = -\mathbf{C}^{(k-1)T} \mathbf{r}^{(k-1)}, \quad (6.93)$$

wo  $\mathbf{C}^{(k-1)}$  die Jacobi-Matrix und  $\mathbf{r}^{(k-1)}$  den Residuenvektor für  $\mathbf{x}^{(k-1)}$  darstellen. Wird der Parameter  $t$  so bestimmt, dass

$$F(\mathbf{x}^{(k)}) = \min_t F(\mathbf{x}^{(k-1)} + t\mathbf{v}^{(k)}) \quad (6.94)$$

gilt, spricht man von der *Methode des steilsten Abstiegs*. Die in (6.94) zu minimierende Funktion ist nichtlinear in der Unbekannten  $t$ . Der Wert von  $t$  wird aus Gründen des Rechenaufwandes in der Regel nur näherungsweise mit Hilfe eines Suchverfahrens ermittelt [Bre 02, Jac 72]. Die lokal optimale Suchrichtung (6.93) liefert eine Folge von Näherungen  $\mathbf{x}^{(k)}$ , welche in der Regel sehr langsam gegen die Lösung  $\mathbf{x}$  konvergiert. Aus diesem Grund erweist sich die Methode des steilsten Abstiegs oft als sehr ineffizient.

**Satz 6.4.** Der Korrekturvektor  $\boldsymbol{\xi}^{(k+1)}$  der Gauß-Newton-Methode für die Näherung  $\mathbf{x}^{(k)}$  stellt stets eine Abstiegsrichtung dar, solange  $\nabla F(\mathbf{x}^{(k)}) \neq \mathbf{0}$  ist.

*Beweis.* Es ist zu zeigen, dass der Korrekturvektor  $\boldsymbol{\xi}^{(k+1)}$  als Lösung der linearisierten Fehlergleichungen  $\mathbf{C}^{(k)}\boldsymbol{\xi}^{(k+1)} - \mathbf{d}^{(k)} = \boldsymbol{\varrho}^{(k)}$  mit dem Gradienten  $\nabla F(\mathbf{x}^{(k)})$  einen stumpfen Winkel bildet. Zur Vereinfachung der Schreibweise lassen wir im Folgenden die oberen Indizes weg. Somit ist zu zeigen, dass  $(\nabla F)^T \boldsymbol{\xi} < 0$  gilt. Wir verwenden die Singulärwertzerlegung (6.68) der Matrix  $\mathbf{C} = \mathbf{U}\hat{\mathbf{S}}\mathbf{V}^T$ , um den allgemeinen Fall zu erfassen. Da  $\mathbf{d} = -\mathbf{r}$  ist, hat der Gradient  $\nabla F$  die Darstellung

$$\nabla F = -2\mathbf{C}^T \mathbf{d} = -2\mathbf{V}(\hat{\mathbf{S}}^T \mathbf{U}^T \mathbf{d}) = -2 \sum_{j=1}^r s_j (\mathbf{u}_j^T \mathbf{d}) \mathbf{v}_j. \quad (6.95)$$

Der Korrekturvektor  $\boldsymbol{\xi}^*$  mit minimaler euklidischer Länge ist nach (6.80)

$$\boldsymbol{\xi}^* = \sum_{i=1}^r \frac{(\mathbf{u}_i^T \mathbf{d})}{s_i} \mathbf{v}_i,$$

und somit ist wegen der Orthonormierung der Vektoren  $\mathbf{v}_i$

$$(\nabla F)^T \boldsymbol{\xi}^* = -2 \sum_{j=1}^r (\mathbf{u}_j^T \mathbf{d})^2 < 0,$$

da in (6.95) nicht alle Skalare  $\mathbf{u}_j^T \mathbf{d}$  verschwinden können, solange  $\nabla F \neq \mathbf{0}$  ist.  $\square$

Mit der Gauß-Newton-Methode ergibt sich auf Grund von Satz 6.4 ein Minimierungsalgorithmus. Nach Wahl eines Startvektors  $\mathbf{x}^{(0)}$  führen wir für  $k = 0, 1, \dots$  die folgenden Schritte durch: Aus den linearisierten Fehlergleichungen  $\mathbf{C}^{(k)}\boldsymbol{\xi}^{(k+1)} - \mathbf{d}^{(k)} = \boldsymbol{\varrho}^{(k)}$  wird der Korrekturvektor  $\boldsymbol{\xi}^{(k+1)}$  als Abstiegsrichtung berechnet. Um eine Abnahme des Funktionswertes  $F(\mathbf{x}^{(k+1)})$  gegenüber  $F(\mathbf{x}^{(k)})$  zu erzielen, prüft man für die Folge der Parameterwerte  $t = 1, 1/2, 1/4, \dots$  mit den Vektoren  $\mathbf{y} := \mathbf{x}^{(k)} + t\boldsymbol{\xi}^{(k+1)}$ , ob die Bedingung  $F(\mathbf{y}) < F(\mathbf{x}^{(k)})$  erfüllt ist. Ist dies der Fall, dann setzen wir  $\mathbf{x}^{(k+1)} = \mathbf{y}$ , und es folgt ein Test auf Konvergenz. Man beachte, dass zur Berechnung von  $F(\mathbf{y})$  der zu  $\mathbf{y}$  gehörende Residuenvektor  $\mathbf{r}$  berechnet werden muss. Sobald  $\mathbf{y}$  ein akzeptabler Vektor ist, ist für die Fehlergleichungen des nächsten Iterationsschrittes bereits  $\mathbf{d}^{(k+1)} = -\mathbf{r}^{(k+1)}$  bekannt.

Durch die garantierte Abnahme der nach unten beschränkten Summe der Quadrate der Residuen ist die Konvergenz der Folge  $\mathbf{x}^{(k)}$  sichergestellt. Bei ungünstiger Wahl des Startvektors  $\mathbf{x}^{(0)}$  kann die Konvergenz zu Beginn der Iteration sehr langsam sein. In der Nähe des Lösungsvektors  $\mathbf{x}$  ist die Konvergenz annähernd quadratisch [Fle 00].

Eine effizientere Methode stammt von *Marquardt* [Mar 63]. Um eine günstigere Abstiegsrichtung zu bestimmen, betrachtet er die Aufgabe, mit  $\mathbf{C} = \mathbf{C}^{(k)}$  und  $\mathbf{d} = \mathbf{d}^{(k)}$  den Vektor  $\mathbf{v}$  als Lösung des Extremalproblems

$$\|\mathbf{C}\mathbf{v} - \mathbf{d}\|_2^2 + \lambda^2\|\mathbf{v}\|_2^2 = \text{Min!}, \quad \lambda > 0, \quad (6.96)$$

zu bestimmen. Bei gegebenem Wert des Parameters  $\lambda$  ist  $\mathbf{v}$  die Lösung des Systems von Fehlgleichungen nach der Methode der kleinsten Quadrate

$$\begin{aligned} \tilde{\mathbf{C}}\mathbf{v} - \tilde{\mathbf{d}} &= \tilde{\boldsymbol{\varrho}} \text{ mit } \tilde{\mathbf{C}} := \left( \frac{\mathbf{C}}{\lambda \mathbf{I}} \right) \in \mathbb{R}^{(N+n), n}, \\ \tilde{\mathbf{d}} &:= \left( \frac{\mathbf{d}}{\mathbf{0}} \right) \in \mathbb{R}^{N+n}, \quad \tilde{\boldsymbol{\varrho}} \in \mathbb{R}^{N+n}. \end{aligned} \quad (6.97)$$

Für jedes  $\lambda > 0$  hat die Matrix  $\tilde{\mathbf{C}}$  den Maximalrang  $n$  unabhängig von Rang  $\mathbf{C}$ . Im Vergleich zur Gauß-Newton-Methode wurde jenes Fehlgleichungssystem (6.87) um  $n$  Gleichungen erweitert und auf diese Weise regularisiert. Der Lösungsvektor  $\mathbf{v}$  besitzt folgende Eigenschaften.

**Satz 6.5.** *Der Vektor  $\mathbf{v} = \mathbf{v}^{(k+1)}$  als Lösung von (6.96) ist eine Abstiegsrichtung, solange  $\nabla F(\mathbf{x}^{(k)}) \neq \mathbf{0}$  ist.*

*Beweis.* Da  $\tilde{\mathbf{C}}$  Maximalrang hat, kann  $\mathbf{v}$  als Lösung von (6.97) formal mit Hilfe der zugehörigen Normalgleichungen dargestellt werden in der Form

$$\mathbf{v} = (\tilde{\mathbf{C}}^T \tilde{\mathbf{C}})^{-1} (\tilde{\mathbf{C}}^T \tilde{\mathbf{d}}) = (\tilde{\mathbf{C}}^T \tilde{\mathbf{C}})^{-1} (\mathbf{C}^T \mathbf{d}) = -\frac{1}{2} (\tilde{\mathbf{C}}^T \tilde{\mathbf{C}})^{-1} (\nabla F). \quad (6.98)$$

Folglich ist

$$(\nabla F)^T \mathbf{v} = -\frac{1}{2} (\nabla F)^T (\tilde{\mathbf{C}}^T \tilde{\mathbf{C}})^{-1} (\nabla F) < 0, \quad \text{falls } \nabla F \neq \mathbf{0},$$

denn die Matrix  $(\tilde{\mathbf{C}}^T \tilde{\mathbf{C}})^{-1}$  ist symmetrisch und positiv definit, und somit bilden  $\nabla F$  und  $\mathbf{v}$  einen stumpfen Winkel.  $\square$

**Satz 6.6.** *Die euklidische Norm  $\|\mathbf{v}\|_2$  des Vektors  $\mathbf{v}$  als Lösung von (6.96) ist mit zunehmendem  $\lambda$  eine monoton abnehmende Funktion.*

*Beweis.* Die Matrix  $\mathbf{A}$  der Normalgleichungen zu (6.97) ist wegen der speziellen Struktur von  $\tilde{\mathbf{C}}$  gegeben durch

$$\mathbf{A} = \tilde{\mathbf{C}}^T \tilde{\mathbf{C}} = \mathbf{C}^T \mathbf{C} + \lambda^2 \mathbf{I}.$$

Zur symmetrischen, positiv semidefiniten Matrix  $\mathbf{C}^T \mathbf{C}$  existiert eine orthogonale Matrix  $\mathbf{U} \in \mathbb{R}^{n,n}$ , so dass gelten

$$\begin{aligned}\mathbf{U}^T \mathbf{C}^T \mathbf{C} \mathbf{U} &= \mathbf{D} \quad \text{und} \quad \mathbf{U}^T \mathbf{A} \mathbf{U} = \mathbf{D} + \lambda^2 \mathbf{I}, \\ \mathbf{D} &= \text{diag } (d_1, d_2, \dots, d_n), \quad d_i \geq 0.\end{aligned}\tag{6.99}$$

Aus (6.98) und (6.99) folgt für das Quadrat der euklidischen Norm

$$\begin{aligned}\|\mathbf{v}\|_2^2 &= \mathbf{v}^T \mathbf{v} = \mathbf{d}^T \mathbf{C} \mathbf{U} (\mathbf{D} + \lambda^2 \mathbf{I})^{-1} \mathbf{U}^T \mathbf{U} (\mathbf{D} + \lambda^2 \mathbf{I})^{-1} \mathbf{U}^T \mathbf{C}^T \mathbf{d} \\ &= \sum_{j=1}^n \frac{h_j^2}{(d_j + \lambda^2)^2} \quad \text{mit } \mathbf{h} := \mathbf{U}^T \mathbf{C}^T \mathbf{d} = (h_1, h_2, \dots, h_n)^T\end{aligned}$$

und damit die Behauptung.  $\square$

Im *Verfahren von Marquardt* zur Minimierung der Summe der Quadrate der Residuen wird die euklidische Norm des Vektors  $\mathbf{v}^{(k+1)}$  durch den Parameter  $\lambda$  so gesteuert, dass mit

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{v}^{(k+1)}, \quad F(\mathbf{x}^{(k+1)}) < F(\mathbf{x}^{(k)}), \quad k = 0, 1, 2, \dots,\tag{6.100}$$

gilt. Die Wahl des Wertes  $\lambda$  erfolgt auf Grund des Rechenablaufes. Ist die Bedingung (6.100) für das momentane  $\lambda$  erfüllt, dann soll  $\lambda$  für den nachfolgenden Schritt verkleinert werden, beispielsweise halbiert. Ist aber (6.100) für das momentane  $\lambda$  nicht erfüllt, soll  $\lambda$  solange vergrößert, beispielsweise verdoppelt werden, bis ein Vektor  $\mathbf{v}^{(k+1)}$  resultiert, für den die Bedingung gilt. Selbstverständlich muss mit dem Startvektor  $\mathbf{x}^{(0)}$  auch ein Startwert  $\lambda^{(0)}$  vorgegeben werden. Ein problemabhängiger Vorschlag ist

$$\lambda^{(0)} = \|\mathbf{C}^{(0)}\|_F / \sqrt{nN} = \sqrt{\frac{1}{nN} \sum_{i,j} (c_{ij}^{(0)})^2}$$

mit der Frobenius-Norm der Matrix  $\mathbf{C}^{(0)}$  zum Startvektor  $\mathbf{x}^{(0)}$ .

Ein Iterationsschritt des Verfahrens von Marquardt erfordert die Berechnung des Vektors  $\mathbf{v}^{(k+1)}$  aus dem Fehlergleichungssystem (6.97) für möglicherweise mehrere Werte des Parameters  $\lambda$ . Um diesen Schritt möglichst effizient zu gestalten, erfolgt die Behandlung von (6.97) in zwei Teilen [Gol 73]. In einem vorbereitenden Teilschritt werden die ersten  $N$ , von  $\lambda$  unabhängigen Fehlergleichungen mit einer orthogonalen Matrix  $\mathbf{Q}_1$  so transformiert, dass

$$\begin{aligned}\mathbf{Q}_1^T \tilde{\mathbf{C}} &= \begin{pmatrix} \mathbf{R}_1 \\ \mathbf{0}_1 \\ \lambda \mathbf{I} \end{pmatrix}, \quad \mathbf{Q}_1^T \tilde{\mathbf{d}} = \begin{pmatrix} \hat{\mathbf{d}}_1 \\ \hat{\mathbf{d}}_2 \\ \mathbf{0} \end{pmatrix}, \\ \mathbf{R}_1 &\in \mathbb{R}^{n,n}, \mathbf{0}_1 \in \mathbb{R}^{(N-n),n}\end{aligned}\tag{6.101}$$

gilt. Unter der Annahme, dass  $\mathbf{C}$  Maximalrang hat, ist  $\mathbf{R}_1$  eine reguläre Rechtsdreiecksmatrix, und die Transformation kann entweder mit der Methode von Givens oder Householder erfolgen. Auf jeden Fall bleiben die letzten  $n$  Fehlergleichungen unverändert. Ausgehend

von (6.101) wird zu gegebenem  $\lambda$  mit einer orthogonalen Matrix  $\mathbf{Q}_2$  die Transformation beendet, um die Matrizen und Vektoren

$$\mathbf{Q}_2^T \mathbf{Q}_1^T \tilde{\mathbf{C}} = \begin{pmatrix} \mathbf{R}_2 \\ \mathbf{0}_1 \\ \mathbf{0}_2 \end{pmatrix}, \quad \mathbf{Q}_2^T \mathbf{Q}_1^T \tilde{\mathbf{d}} = \begin{pmatrix} \hat{\mathbf{d}}_1 \\ \hat{\mathbf{d}}_2 \\ \hat{\mathbf{d}}_3 \end{pmatrix}, \quad (6.102)$$

$\mathbf{R}_2, \mathbf{0}_2 \in \mathbb{R}^{n,n}$

zu erhalten. Der gesuchte Vektor  $\mathbf{v}^{(k+1)}$  ergibt sich aus  $\mathbf{R}_2 \mathbf{v}^{(k+1)} - \hat{\mathbf{d}}_1 = \mathbf{0}$  mit der stets regulären Rechtsdreiecksmatrix  $\mathbf{R}_2$  durch Rücksubstitution. Muss der Wert  $\lambda$  vergrößert werden, so ist nur der zweite Teilschritt zu wiederholen. Dazu sind  $\mathbf{R}_1$  und  $\hat{\mathbf{d}}_1$  als Ergebnis des ersten Teils abzuspeichern. Da die Nullmatrix  $\mathbf{0}_1$  und der Teilvektor  $\hat{\mathbf{d}}_2$  (6.101) für den zweiten Teilschritt bedeutungslos sind, kann er sowohl speicherökonomisch als auch sehr effizient durchgeführt werden. Die sehr spezielle Struktur der noch zu behandelnden Fehlergleichungen des zweiten Teilschrittes weist auf die Anwendung der Givens-Transformation hin. Die im Abschnitt 6.2.2 dargelegte Rechentechnik erfordert den kleinsten Speicherbedarf. Soll hingegen die Householder-Transformation angewandt werden, ist der zweite Teilschritt mit einer Hilfsmatrix  $\mathbf{C}_H \in \mathbb{R}^{(2n),n}$  und einem Hilfsvektor  $\mathbf{d}_H \in \mathbb{R}^{2n}$  durchführbar, wobei auch hier die sehr spezielle Struktur zur Erhöhung der Effizienz berücksichtigt werden kann. In manchen praktischen Aufgabenstellungen der Naturwissenschaften liegt den Fehlergleichungen eine bestimmte Funktion

$$f(x) = \sum_{j=1}^{\mu} a_j \varphi_j(x; \alpha_1, \alpha_2, \dots, \alpha_{\nu})$$

zu Grunde, in welcher die unbekannten Parameter  $a_1, a_2, \dots, a_{\mu}; \alpha_1, \alpha_2, \dots, \alpha_{\nu}$  aus einer Anzahl von  $N$  Beobachtungen der Funktion  $f(x)$  für  $N$  verschiedene Argumente  $x_i$  zu bestimmen sind. Dabei sind die Funktionen  $\varphi_j(x; \alpha_1, \alpha_2, \dots, \alpha_{\nu})$  in den  $\alpha_k$  nichtlinear. In den resultierenden Fehlergleichungen verhalten sich die  $a_j$  bei festen  $\alpha_k$  linear und umgekehrt die  $\alpha_k$  bei festgehaltenen  $a_j$  nichtlinear. In diesem Sinn existieren zwei Klassen von unbekannten Parametern, und zur effizienten Behandlung solcher Probleme ist ein spezieller Algorithmus entwickelt worden [Gol 73].

## 6.5 Software

Die Themen dieses Kapitels sind einerseits dem Bereich *Numerische lineare Algebra*, andererseits dem der *Optimierung* bzw. genauer dem der *Minimierung positiver Funktionen* zuzuordnen. Software zu den Verfahren findet man dementsprechend in unterschiedlichen Bereichen. Alle beschriebenen Methoden sind wieder in den großen numerischen Bibliotheken realisiert. In der NAG-Bibliothek sind die Verfahren, die mit orthogonalen Transformationen arbeiten, in den Kapiteln F02 und F08 zu finden, in F02 auch Black-box-Routinen zur Singulärwertzerlegung (SVD). Verfahren zum Minimieren und Maximieren von Funktionen und damit auch zur Methode der kleinsten Quadrate sind im Kapitel E04 zusammengefasst.

In MATLAB wird ein lineares Gleichungssystem  $\mathbf{Ax} = \mathbf{b}$  durch den einfachen Befehl `x=A\b` gelöst. Ist das Gleichungssystem unter- oder überbestimmt, dann wird automatisch die Lösung kleinster Quadrate berechnet. Mit dem Befehl `lsqnonneg` berechnet MATLAB eine nicht negative Lösung des Problems kleinster Quadrate. Die Lösungen der beiden zuletzt genannten Probleme können sich stark unterscheiden.

Eine spezielle Sammlung von FORTRAN90-Routinen zum Problem der kleinsten Quadrate ist im Zusammenhang mit [Law 95] erschienen<sup>1</sup>.

Unsere Problemlöseumgebung PAN (<http://www.upb.de/SchwarzKoeckler/>) verfügt über ein Programm zur Singulärwertzerlegung und eines zur Lösung allgemeiner linearer Gleichungssysteme, deren Koeffizientenmatrix singulär oder rechteckig sein kann, mit der Methode der kleinsten Quadrate.

## 6.6 Aufgaben

**Aufgabe 6.1.** An einem Quader werden die Längen seiner Kanten und die Umfänge senkrecht zur ersten und zweiten Kante gemessen. Die Messwerte sind:

$$\begin{aligned} \text{Kante 1: } & 26 \text{ mm; Kante 2: } 38 \text{ mm; Kante 3: } 55 \text{ mm;} \\ \text{Umfang } \perp \text{ Kante 1: } & 188 \text{ mm; Umfang } \perp \text{ Kante 2: } 163 \text{ mm.} \end{aligned}$$

Wie groß sind die ausgeglichenen Kantenlängen nach der Methode der kleinsten Quadrate?

**Aufgabe 6.2.** Um Amplitude  $A$  und Phasenwinkel  $\phi$  einer Schwingung  $x = A \sin(2t + \phi)$  zu bestimmen, sind an vier Zeitpunkten  $t_k$  die Auslenkungen  $x_k$  beobachtet worden.

$t_k =$	0	$\pi/4$	$\pi/2$	$3\pi/4$
$x_k =$	1.6	1.1	-1.8	-0.9

Um ein lineares Fehlergleichungssystem zu erhalten, sind auf Grund einer trigonometrischen Formel neue Unbekannte einzuführen.

**Aufgabe 6.3.** Die Funktion  $y = \sin(x)$  ist im Intervall  $[0, \pi/4]$  durch ein Polynom  $P(x) = a_1x + a_3x^3$  zu approximieren, das wie  $\sin(x)$  ungerade ist. Die Koeffizienten  $a_1$  und  $a_3$  sind nach der Methode der kleinsten Quadrate für die diskreten Stützstellen  $x_k = k\pi/24$ ,  $k = 1, 2, \dots, 6$ , zu bestimmen. Mit dem gefundenen Polynom  $P(x)$  zeichne man den Graphen der Fehlerfunktion  $r(x) := P(x) - \sin(x)$ .

**Aufgabe 6.4.** Man schreibe oder benutze Programme zur Lösung von linearen Fehlergleichungssystemen, um mit Hilfe der Normalgleichungen und der beiden Varianten der Orthogonaltransformation die Funktionen

a)  $f(x) = \cos(x)$ ,  $x \in [0, \pi/2]$ ;      b)  $f(x) = e^x$ ,  $x \in [0, 1]$

durch Polynome  $n$ -ten Grades  $P_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$  für  $n = 2, 3, \dots, 8$  so zu approximieren, dass die Summe der Quadrate der Residuen an  $N = 10, 20$  äquidistanten Stützstellen minimal ist. Zur Erklärung der verschiedenen Ergebnisse berechne man Schätzwerte der

<sup>1</sup>[http://orion.math.iastate.edu/burkardt/f\\_src/lawson.html](http://orion.math.iastate.edu/burkardt/f_src/lawson.html)

Konditionszahlen der Normalgleichungsmatrix  $\mathbf{A}$  mit Hilfe der Inversen der Rechtsdreiecksmatrix  $\mathbf{R}$  unter Benutzung von  $\|\mathbf{A}^{-1}\|_F \leq \|\mathbf{R}^{-1}\|_F \|\mathbf{R}^{-T}\|_F = \|\mathbf{R}^{-1}\|_F^2$ , wo  $\|\cdot\|_F$  die Frobenius-Norm bedeutet.

**Aufgabe 6.5.** An einem Quader misst man die Kanten der Grundfläche  $a = 21$  cm,  $b = 28$  cm und die Höhe  $c = 12$  cm. Weiter erhält man als Messwerte für die Diagonale der Grundfläche  $d = 34$  cm, für die Diagonale der Seitenfläche  $e = 24$  cm und für die Körperdiagonale  $f = 38$  cm. Zur Bestimmung der Längen der Kanten des Quaders nach der Methode der kleinsten Quadrate verwende man das Verfahren von Gauß-Newton und Minimierungsmethoden.

**Aufgabe 6.6.** Um den Standort eines illegalen Senders festzustellen, werden fünf Peilwagen eingesetzt, mit denen die Richtungen zum Sender ermittelt werden. Die Aufstellung der Peilwagen ist in einem  $(x, y)$ -Koordinatensystem gegeben, und die Richtungswinkel  $\alpha$  sind von der positiven  $x$ -Achse im Gegenuhrzeigersinn angegeben.

Peilwagen	1	2	3	4	5
$x$ -Koordinate	4	18	26	13	0
$y$ -Koordinate	1	0	15	16	14
Richtungswinkel $\alpha$	$45^\circ$	$120^\circ$	$210^\circ$	$270^\circ$	$330^\circ$

Die Situation ist an einer großen Zeichnung darzustellen. Welches sind die mutmaßlichen Koordinaten des Senders nach der Methode der kleinsten Quadrate? Als Startwert für das Verfahren von Gauß-Newton und für Minimierungsmethoden wähle man beispielsweise  $P_0(12.6, 8.0)$ .

**Aufgabe 6.7.** Die Konzentration  $z(t)$  eines Stoffes in einem chemischen Prozess gehorcht dem Gesetz

$$z(t) = a_1 + a_2 e^{\alpha_1 t} + a_3 e^{\alpha_2 t}, \quad a_1, a_2, a_3, \alpha_1, \alpha_2 \in \mathbb{R}, \quad \alpha_1, \alpha_2 < 0.$$

Zur Bestimmung der Parameter  $a_1, a_2, a_3, \alpha_1, \alpha_2$  liegen für  $z(t)$  folgende Messwerte  $z_k$  vor.

$t_k =$	0	0.5	1.0	1.5	2.0	3.0	5.0	8.0	10.0
$z_k =$	3.85	2.95	2.63	2.33	2.24	2.05	1.82	1.80	1.75

Als Startwerte verwende man beispielsweise  $a_1^{(0)} = 1.75$ ,  $a_2^{(0)} = 1.20$ ,  $a_3^{(0)} = 0.8$ ,  $\alpha_1^{(0)} = -0.5$ ,  $\alpha_2^{(0)} = -2$  und behandle die nichtlinearen Fehlgleichungen mit dem Verfahren von Gauß-Newton und mit Minimierungsmethoden.

## 7 Numerische Integration

Integralberechnungen sind meistens Teil einer umfassenderen mathematischen Problemstellung. Dabei sind die auftretenden Integrationen oft nicht analytisch ausführbar, oder ihre analytische Durchführung stellt im Rahmen der Gesamtaufgabe eine praktische Schwierigkeit dar. In solchen Fällen wird der zu berechnende Integralausdruck angenähert ausgewertet durch numerische Integration, die auch numerische Quadratur genannt wird. Zu den zahlreichen Anwendungen der numerischen Quadratur gehören die Berechnung von Oberflächen, Volumina, Wahrscheinlichkeiten und Wirkungsquerschnitten, die Auswertung von Integraltransformationen und Integralen im Komplexen, die Konstruktion von konformen Abbildungen für Polygonbereiche nach der Formel von Schwarz-Christoffel [Hen 97], die Behandlung von Integralgleichungen etwa im Zusammenhang mit der Randelementmethode und schließlich die Methode der finiten Elemente, siehe etwa [Sch 91b] oder Abschnitt 10.3.

Wir behandeln in diesem Kapitel die Berechnung eines bestimmten Integrals  $I$ , das durch eine Summe – die numerische *Quadraturformel*  $\tilde{I}$  – approximiert wird

$$I = \int_a^b f(x) dx \quad \longrightarrow \quad \tilde{I} = \sum_{i=1}^n w_i f(x_i), \quad (7.1)$$

wobei die Wahl der Koeffizienten oder Integrationsgewichte  $w_i$  und der Stützstellen  $x_i$  die Regel festlegen. Für einige Regeln wird von 0 bis  $n$  summiert.

Es gibt numerische Verfahren, die als Ergebnis die Entwicklung des Integranden in eine Potenzreihe liefern. Die Potenzreihe kann dann analytisch integriert werden. Dadurch ist auch die unbestimmte Integration möglich. Sie entspricht der Lösung einer Differentialgleichung, siehe Lemma 8.3. Eine Stammfunktion kann daher auch mit den Methoden des Kapitels 8 berechnet werden.

Die “Integration von Tabellendaten” wird hier nicht behandelt. Durch eine Wertetabelle kann eine interpolierende oder approximierende Funktion gelegt werden, die dann exakt integriert wird, siehe Kapitel 3.

Zur Berechnung bestimmter Integrale stehen mehrere Verfahrensklassen zur Verfügung. Das einfachste und bekannteste Verfahren ist sicher die Trapezregel, die zur Klasse der Newton-Cotes-Regeln gehört. Die Trapezregel ist eine einfache Quadraturmethode mit niedriger Genauigkeitsordnung, die aber für spezielle Integranden, etwa periodische Funktionen, durchaus gut geeignet ist. Die Romberg-Quadratur nutzt die Fehlerentwicklung der Trapezregel zur Genauigkeitssteigerung aus. Am wichtigsten für das Verständnis der Algorithmen, die in Softwaresystemen verwendet werden, ist die Klasse der Gauß-Regeln. In dieser Klasse können auch viele Spezialfälle wie Singularitäten behandelt werden. Die Integration von Funktionen mit Singularitäten oder über unbeschränkte Bereiche kann auch mit

Hilfe von Transformationen erleichtert werden. Optimale Black-box-Methoden mit automatischer Steuerung der Genauigkeit erhält man durch adaptive Methoden, die meistens zwei Verfahren einer Klasse kombinieren.

Auch auf mehrdimensionale Integrationen gehen wir kurz ein.

Sollen numerische Integrationsregeln verglichen werden, so wird der Aufwand in *Anzahl Funktionsauswertungen* gemessen, da dieser rechnerische Aufwand den der anderen Rechenoperationen in allen wichtigen Fällen dominiert.

## 7.1 Newton-Cotes-Formeln

Dies ist die einfachste Idee: Um das Integral  $I$  zu berechnen, wird die Funktion  $f$  durch ein interpolierendes Polynom  $p$  ersetzt und dieses wird exakt integriert.

### 7.1.1 Konstruktion von Newton-Cotes-Formeln

Zu  $m+1$  Stützstellen werden die Werte  $(x_i, f(x_i))$ ,  $i = 0, 1, \dots, m$ , mit Lagrange interpoliert

$$p_m(x) = \sum_{i=0}^m f(x_i) L_i(x). \quad (7.2)$$

Dabei sind  $L_i(x)$  die Lagrange-Polynome der Ordnung  $m$ , siehe Abschnitt 3.1.2. Damit wird als Integralnäherung

$$\begin{aligned} \tilde{I}_m &= (b-a) \sum_{i=0}^m w_i f(x_i) \quad \text{mit} \\ w_i &= \frac{1}{b-a} \int_a^b L_i(x) dx \end{aligned} \quad (7.3)$$

berechnet. Für den Fall äquidistanter Stützstellen

$$x_i = a + i h \quad \text{mit } h = \frac{b-a}{m} \quad (7.4)$$

ergeben sich die  $w_i$  mit der Substitution  $s = (x - a)/h$  zu

$$w_i = \frac{1}{b-a} \int_a^b \prod_{\substack{j=0 \\ j \neq i}}^m \frac{(x-x_j)}{(x_i-x_j)} dx = \frac{1}{m} \int_0^m \prod_{\substack{j=0 \\ j \neq i}}^m \frac{(s-j)}{(i-j)} ds. \quad (7.5)$$

### Beispiel 7.1. Konstruktion der Simpson-Regel

Hier sind  $m = 2$ ,  $x_0 = a$ ,  $x_1 = (a+b)/2$ ,  $x_2 = b$ ,  $h = (b-a)/2$ . Damit ergeben sich die Koeffizienten  $w_i$  zu

$$\begin{aligned} w_0 &= \frac{1}{2} \int_0^2 \frac{(s-1)(s-2)}{2} ds = \frac{1}{6}, \\ w_1 &= \frac{1}{2} \int_0^2 \frac{s(s-2)}{-1} ds = \frac{4}{6}, \end{aligned} \quad (7.6)$$

$$w_2 = \frac{1}{2} \int_0^2 \frac{s(s-1)}{2} ds = \frac{1}{6}.$$

Das ergibt die unten aufgeführte Simpson-Regel.  $\triangle$

Auf Grund der Definition (7.3) ist klar, dass  $\tilde{I}_m$  den exakten Wert des Integrals liefert, falls  $f$  ein Polynom vom Höchstgrad  $m$  ist. Andernfalls stellt  $\tilde{I}_m$  eine Näherung für  $I$  mit einem vom Integranden abhängigen Fehler

$$E_m[f] := \tilde{I}_m - I = (b-a) \sum_{i=0}^m w_i f(x_i) - \int_a^b f(x) dx \quad (7.7)$$

dar. In bestimmten Fällen ist die Quadraturformel (7.3) auch noch exakt für Polynome höheren Grades. Als Maß für die Güte der Genauigkeit einer Quadraturformel wird folgender Begriff eingeführt.

**Definition 7.1.** Eine Quadraturformel besitzt den *Genauigkeitsgrad*  $m \in \mathbb{N}^*$ , wenn sie alle Polynome vom Höchstgrad  $m$  exakt integriert, und  $m$  die größtmögliche Zahl mit dieser Eigenschaft ist.

Da  $E_m[f]$  ein lineares Funktional in  $f$  ist, besitzt eine Quadraturformel genau dann den Genauigkeitsgrad  $m$ , wenn

$$E[x^j] = 0 \quad \text{für } j = 0, 1, \dots, m \quad \text{und} \quad E[x^{m+1}] \neq 0 \quad (7.8)$$

gilt. Aus den bisherigen Betrachtungen folgt zusammenfassend

**Satz 7.2.** Zu beliebig vorgegebenen  $(m+1)$  paarweise verschiedenen Stützstellen  $x_i \in [a, b]$  existiert eine eindeutig bestimmte Newton-Cotes-Formel (7.3), deren Genauigkeitsgrad mindestens  $m$  ist.

Der Quadraturfehler  $E_m[f] := \tilde{I}_m - I$  einer Newton-Cotes-Formel  $m$ -ten Grades zu  $m+1$  Stützstellen hat für eine  $(m+1)$  mal stetig differenzierbare Funktion  $f(x)$  wegen (3.5) die Darstellung

$$E_m[f] = \frac{1}{(m+1)!} \int_a^b f^{(m+1)}(\xi(x)) \omega(x) dx \quad \text{mit } \omega(x) = \prod_{i=0}^m (x - x_i). \quad (7.9)$$

Die Methode der Newton-Cotes-Regeln ist nur für kleine Werte von  $m$  stabil; es werden deshalb hier nur die Formeln für  $m = 1$  und  $m = 2$  angegeben:

$$\text{Trapezregel : } \int_a^b f(x) dx \approx \frac{b-a}{2} (f(a) + f(b)), \quad (7.10)$$

$$\text{Simpson-Regel : } \int_a^b f(x) dx \approx \frac{b-a}{6} \left( f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right). \quad (7.11)$$

Soll die Genauigkeit erhöht werden, so werden diese einfachen Näherungsformeln mehrfach aneinandergesetzt. Sei zu gegebenem  $n$

$$h = \frac{b-a}{n} \quad \text{und} \quad x_j = a + j h, \quad j = 0, 1, \dots, n. \quad (7.12)$$

Dann liefert das Aneinanderhängen von  $n$  Trapezregeln die Näherungsformel

$$\tilde{I} = T(h) = \frac{h}{2} [f(x_0) + 2f(x_1) + \cdots + 2f(x_{n-1}) + f(x_n)], \quad (7.13)$$

Für gerades  $n$  können  $n/2$  Simpson-Regeln zusammengefügt werden:

$$\tilde{I} = S(h) = \frac{h}{3} [f(x_0) + 4f(x_1) + 2f(x_2) + \cdots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)]. \quad (7.14)$$

Sind Schranken für die 2. bzw. 4. Ableitung der zu integrierenden Funktion bekannt, so lässt sich der Fehler dieser Regeln abschätzen

$$|I - T(h)| \leq \frac{|b-a|}{12} h^2 \max_{x \in [a,b]} |f''(x)|, \quad (7.15)$$

$$|I - S(h)| \leq \frac{|b-a|}{180} h^4 \max_{x \in [a,b]} |f^{(4)}(x)|. \quad (7.16)$$

Der Beweis dieser Abschätzungen erfordert umfangreiche Hilfsmittel, man findet ihn etwa in [Häm 91].

### Beispiel 7.2.

$$I = \int_0^{\pi/2} \frac{5.0}{e^\pi - 2} \exp(2x) \cos(x) dx = 1.0. \quad (7.17)$$

Abb. 7.1 links zeigt den Integranden und die Trapezfläche als Näherung für das Integral mit  $n = 4$ . Die Ergebnisse für Trapez- und Simpson-Regel sind in der folgenden Tabelle festgehalten:

Regel	$h$	$\tilde{I}$	$\tilde{I} - I$	$(7.15)/(7.16)$
Trapez	$\pi/8$	0.926	-0.074	0.12
Simpson	$\pi/8$	0.9925	-0.0075	0.018

△

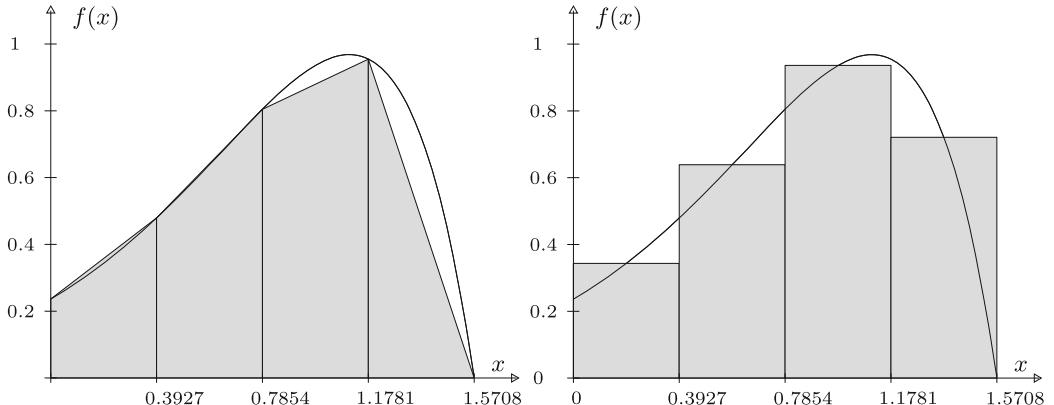
### 7.1.2 Verfeinerung der Trapezregel

Eine anschauliche Approximation des Integrals (7.1), welche der Riemannschen Summe entspricht, bildet die *Mittelpunktsumme*

$$M(h) := h \sum_{j=0}^{n-1} f\left(x_{j+\frac{1}{2}}\right), \quad x_{j+\frac{1}{2}} := a + \left(j + \frac{1}{2}\right) h. \quad (7.18)$$

$M(h)$  stellt die Fläche unterhalb der Treppenkurve in Abb. 7.1 rechts dar. Aus (7.13) und (7.18) folgt unmittelbar die Relation

$$T(h/2) = \frac{1}{2}[T(h) + M(h)]. \quad (7.19)$$

Abb. 7.1 Trapez- und Mittelpunktregel für Beispiel 7.2 ( $n = 4$ ).

Die Beziehung (7.19) erlaubt die Verbesserung der Trapezapproximationen durch sukzessive Halbierung der Schrittänge in der Weise, dass zur bereits berechneten Näherung  $T(h)$  noch  $M(h)$  berechnet wird. Bei jeder Halbierung der Schrittweite wird der Rechenaufwand, gemessen mit der Anzahl der Funktionsauswertungen, etwa verdoppelt, doch werden die schon berechneten Funktionswerte auf ökonomische Weise wieder verwendet. Die sukzessive Halbierung der Schrittweite kann beispielsweise dann abgebrochen werden, wenn sich  $T(h)$  und  $M(h)$  um weniger als eine gegebene Toleranz  $\varepsilon > 0$  unterscheiden. Dann ist der Fehler  $|T(h/2) - I|$  im Allgemeinen höchstens gleich  $\varepsilon$ .

Die Berechnung der Trapezsummen  $T(h)$  bei sukzessiver Halbierung der Schrittweite  $h$  fassen wir im folgenden Algorithmus zusammen. Es sind  $\varepsilon$  die vorzugebende Toleranz,  $f(x)$  der Integrand und  $a, b$  die Integrationsgrenzen.

$$h = b - a; \quad n = 1; \quad T = h \times (f(a) + f(b))/2$$

für  $k = 1, 2, \dots, 10 :$

$$M = 0$$

für  $j = 0, 1, \dots, n - 1 :$

$$M = M + f(a + (j + 0.5) \times h)$$

$$M = h \times M; \quad T = (T + M)/2; \quad h = h/2; \quad n = 2 \times n$$

falls  $|T - M| < \varepsilon :$  STOP

(7.20)

Ohne weitere Maßnahmen konvergieren die Trapezsummen im Allgemeinen recht langsam gegen den Integralwert  $I$ . Falls aber  $f(x)$  periodisch und analytisch auf  $\mathbb{R}$  ist, und falls  $(b-a)$  gleich der Periode ist, dann bedarf der Algorithmus (7.20) keiner weiteren Verbesserung mehr (vgl. Abschnitt 7.3).

Die Trapezregel erweist sich auch als günstig zur genäherten Berechnung von Integralen über  $\mathbb{R}$  von genügend rasch abklingenden Funktionen  $f(x)$ . Dazu werden die Definitionen (7.13) und (7.18) für das beidseitig unbeschränkte Intervall verallgemeinert, und es wird zusätzlich eine frei wählbare Verschiebung  $s$  eingeführt.

Mit dieser Verallgemeinerung werden die Trapez- und Mittelpunktsummen definiert als

$$\begin{aligned} T(h, s) &:= h \sum_{j=-\infty}^{\infty} f(s + jh); \\ M(h, s) &:= h \sum_{j=-\infty}^{\infty} f\left(s + \left(j + \frac{1}{2}\right)h\right) = T(h, s + h/2). \end{aligned} \quad (7.21)$$

In Analogie zu (7.19) gilt

$$T(h/2, s) = [T(h, s) + M(h, s)]/2.$$

Wegen der sich ins Unendliche erstreckenden Summen ist die Anwendung von (7.21) nur für genügend rasch, beispielsweise exponentiell abklingende Integranden praktikabel. Ausgehend von einer geeignet gewählten Verschiebung  $s$ , welche dem Verlauf des Integranden  $f(x)$  Rechnung trägt, und einem Anfangsschritt  $h_0$  werden die Werte  $T$  und  $M$  zweckmäßig mit  $j = 0$  beginnend und dann mit zunehmendem  $|j|$  nach jeder Seite hin aufsummiert. Die (unendliche) Summation über  $j$  muss abgebrochen werden, sobald die Beträge der Funktionswerte kleiner als eine vorgegebene Abbruchtoleranz  $\delta$  werden. Damit ergibt sich der folgende modifizierte Algorithmus zur Berechnung der Trapezsummen für das uneigentliche Integral

$$I = \int_{-\infty}^{\infty} f(x) dx$$

einer genügend rasch abklingenden Funktion  $f(x)$ .

$$\begin{aligned} h &= h_0; \quad T = f(s); \quad j = 1; \quad z = 0 \\ \text{ST:} \quad f1 &= f(s + j \times h); \quad f2 = f(s - j \times h); \\ &T = T + f1 + f2; \quad j = j + 1 \\ &\text{falls } |f1| + |f2| > \delta : z = 0; \quad \text{gehe nach ST} \\ &z = z + 1; \quad \text{falls } z = 1 : \quad \text{gehe nach ST} \\ &T = h \times T \\ &\text{für } k = 1, 2, \dots, 10 : \\ &\quad M = f(s + 0.5 \times h); \quad j = 1; \quad z = 0 \\ \text{SM :} \quad f1 &= f(s + (j + 0.5) \times h); \\ f2 &= f(s - (j - 0.5) \times h) \\ M &= M + f1 + f2; \quad j = j + 1 \\ &\text{falls } |f1| + |f2| > \delta : z = 0; \quad \text{gehe nach SM} \\ &z = z + 1; \quad \text{falls } z = 1 : \quad \text{gehe nach SM} \\ M &= h \times M; \quad T = (T + M)/2; \quad h = h/2 \\ &\text{falls } |T - M| < \varepsilon : \quad \text{STOP} \end{aligned} \quad (7.22)$$

Um unnötige Funktionsauswertungen im Fall eines asymmetrisch abklingenden Integranden zu vermeiden, könnte der Algorithmus (7.22) so verbessert werden, dass die Summationen nach oben und unten getrennt ausgeführt werden.

**Beispiel 7.3.** Zur genäherten Berechnung von

$$I = \int_{-\infty}^{\infty} e^{-0.25x^2} \cos(2x) dx$$

mit oszillierendem, aber rasch abnehmendem Integranden mit dem Algorithmus (7.22) ergeben sich Trapezsummen gemäß Tab. 7.1. Mit der Startschrittweite  $h_0 = 2$  ist die Zahl  $Z$  der Funktionsauswertungen für drei verschiedene Werte von  $s$  angegeben. Die verwendeten Toleranzen sind  $\delta = 10^{-14}$  und  $\varepsilon = 10^{-8}$ . Das Verhalten des Algorithmus ist von der Wahl von  $s$  praktisch unabhängig. In allen drei Fällen stimmen bei Rechnung mit sechzehn wesentlichen Dezimalstellen die letzten Werte für die Trapez- und die Mittelpunktregel in vierzehn wesentlichen Stellen überein.  $\triangle$

Tab. 7.1 Uneigentliches Integral mit oszillierenden Integranden.

	$s = 0$	$Z$	$s = 0.12$	$Z$	$s = 0.3456$	$Z$
$T$	1.0279082242	15	0.9602843084	15	0.5139297286	15
$M$	-0.8980536479	17	-0.8304297538	17	-0.3840752730	17
$T$	0.0649272881		0.0649272773		0.0649272278	
$M$	0.0649272112	27	0.0649272215	27	0.0649272710	29
$T$	0.0649272494		0.0649272494		0.0649272494	
$M$	0.0649272494	51	0.0649272494	51	0.0649272494	51
$T$	0.0649272494		0.0649272494		0.0649272494	

## 7.2 Romberg-Integration

Eine erhebliche Verkleinerung des Fehlers von Trapez- oder Simpson-Regel hat Romberg erreicht, indem er Richardsons *Extrapolation auf  $h^2 = 0$*  anwandte. Er hat ausgenutzt, dass sich der Fehler (7.9) für die Trapezregel  $T(h)$  in eine Potenzreihe in  $h^2$  (statt nur in  $h$ ) entwickeln lässt. Mit den Bernoulli-Zahlen  $B_k$  ( $B_0 = 1$ ,  $B_1 = -1/2$ ,  $B_2 = 1/6$ ,  $B_3 = B_5 = B_7 = \dots = 0$ ,  $B_4 = -1/30$ ,  $B_6 = 1/42$ ,  $B_8 = -1/30$ ,  $B_{10} = 5/66$ , ...) ergibt sich

$$T(h) - I = \sum_{k=1}^{\infty} \frac{B_{2k}}{(2k)!} h^{2k} [f^{(2k-1)}(b) - f^{(2k-1)}(a)]. \quad (7.23)$$

Wegen des oft starken Wachstums der Ableitungen  $f^{(2k-1)}(x)$  mit zunehmendem  $k$  konvergiert diese Reihe im Allgemeinen nicht, sodass (7.23) ungültig ist. In den meisten Fällen kann jedoch gezeigt werden, dass jede endliche Partialsumme in (7.23) für  $h \rightarrow 0$  zu  $T(h) - I$  asymptotisch ist. In diesen Fällen gilt die so genannte *Euler-MacLaurinsche Summenformel* in der Form

$$T(h) - I = c_1 h^2 + c_2 h^4 + \dots + c_N h^{2N} + R_{N+1}(h) \quad (7.24)$$

für jedes feste  $N$ . Dabei sind die Koeffizienten dieser Fehlerentwicklung unabhängig von  $h$  und es ist  $R_{N+1}(h) = O(h^{2N+2})$ . Das bedeutet, dass für Trapezregelauswertungen mit verschiedenen Schrittweiten die Koeffizienten der Fehlerentwicklung gleich bleiben. Rechnet man z.B. mit zwei Schrittweiten  $h$  und  $H$ , so können die Werte  $T(h)$  und  $T(H)$  so kombiniert werden, dass in der Fehlerentwicklung der Kombination der  $h^2$ -Term verschwindet. Für  $H = 2h$  entsteht die Regel

$$\tilde{I} := \frac{1}{3} (4T(h) - T(H)) \quad \text{mit } \tilde{I} - I = O(h^4). \quad (7.25)$$

Offenbar ergibt diese Kombination gerade die Simpson-Regel. Das Verfahren kann systematisch fortgesetzt werden. Werden mehr als zwei Schrittweiten benutzt, dann ergibt sich ein Dreiecksschema. Dazu werden zunächst die Trapezregeln definiert als  $T_{k,0} := T(h_k)$ . Für die Schrittweitenwahl

$$h_k := \frac{b-a}{2^k}, \quad k = 0, 1, \dots, m, \quad (7.26)$$

lassen sich die Trapezregeln  $T_{k,0}$ ,  $k > 0$ , mit Hilfe der Vorgängerregel  $T_{k-1,0}$  leicht rekursiv berechnen:

$$T_{k,0} = \frac{1}{2} T_{k-1,0} + h_k [f(a + h_k) + f(a + 3h_k) + \dots + f(b - 3h_k) + f(b - h_k)] \quad (7.27)$$

Extrapolation auf  $h^2 = 0$  ergibt das rekursive Dreiecksschema

$$T_{k,j} := \frac{4^j T_{k,j-1} - T_{k-1,j-1}}{4^j - 1} \quad \left\{ \begin{array}{l} j = 1, 2, \dots, m, \\ k = j, j+1, \dots, m. \end{array} \right. \quad (7.28)$$

Der zuletzt berechnete Wert  $T_{m,m}$  ist in der Regel am genauesten, es gilt die Fehlerabschätzung

$$|T_{k,j} - I| \leq \frac{(b-a)^{2j+3} |B_{2j+2}|}{4^{k-j} 2^{j(j+1)} (2j+2)!} \max_{x \in [a,b]} |f^{(2j+2)}(x)| \quad (7.29)$$

mit den Bernoulli-Zahlen  $B_{2j+2}$  wie in (7.23).

**Beispiel 7.4.** Anwendung dieses Algorithmus auf Beispiel 7.2 ergibt das Schema

$k$	$h_k$	$T_{k,0}$	$T_{k,1}$	$T_{k,2}$	$T_{k,3}$
0	$\pi/2$	0.18			
1	$\pi/4$	0.72	0.9044		
2	$\pi/8$	0.93	0.9925	0.998386	
3	$\pi/16$	0.98	0.9995	0.999974	0.999999

Für den auf sechs Stellen genauen Wert  $T_{3,3}$  liefert die Fehlerabschätzung die obere Schranke

$$|T_{3,3} - I| \leq 3 \cdot 10^{-5}.$$

Der Wert  $T_{2,2} = 0.998386$ , der sich aus den ersten drei Zeilen des Dreiecksschemas ergibt, benötigt ebenso fünf Funktionsauswertungen wie die Simpson-Regel in Beispiel 7.2, die aber einen fast fünffach so großen Fehler aufweist. Bei neun Funktionsauswertungen ist der Fehler der Simpson-Regel  $T_{3,1}$  500 mal so groß wie der der Romberg-Integration  $T_{3,3}$ .  $\triangle$

Zur Genauigkeitssteuerung wird oft die obere Diagonale mit den Werten  $T_{k,k}$  benutzt, indem für eine vorgegebene Toleranz  $\varepsilon$  auf  $|T_{k,k} - T_{k-1,k-1}| < \varepsilon$  abgefragt wird. Im Allgemeinen konvergiert diese Diagonale *superlinear*. Wegen (7.23)/(7.24) geht dem Romberg-Algorithmus bei Funktionen mit Singularitäten, Sprungstellen oder Sprüngen in einer Ableitung die Grundlage verloren. Dementsprechend schlechter sind dann die Ergebnisse.

**Beispiel 7.5.** Die zweite Ableitung des scheinbar harmlosen Integranden von

$$I = \int_0^1 x^{3/2} dx = 0.4 \quad (7.30)$$

besitzt an der Stelle  $x = 0$  eine Singularität. Die Romberg-Quadratur ergibt das Schema

$k$	$h_k$	$T_{k,0}$	$T_{k,1}$	$T_{k,2}$	$T_{k,3}$	$T_{k,4}$
0	1	0.50000000				
1	0.5	0.42677670	0.40236893			
2	0.25	0.40701811	0.40043191	0.40030278		
3	0.125	0.40181246	0.40007724	0.40005361	0.40004965	
4	0.0625	0.40046340	0.40001371	0.40000948	0.40000878	0.40000862

Hier bringt nur die zweite Spalte mit den  $T_{k,1}$  – das sind die Simpson-Regeln – noch eine wesentliche Verbesserung gegenüber den Trapezregeln  $T_{k,0}$ , allerdings auch nicht mit einem Fehler  $O(h^4)$  wie bei mindestens viermal stetig differenzierbaren Funktionen. Das liegt daran, dass wegen der Singularität der zweiten Ableitung die Reihenentwicklung des Fehlers (7.24) nach  $c_1 h^2$  ungültig wird.  $\triangle$

Das Romberg-Verfahren besitzt den Vorteil einfach durchführbar zu sein, doch es ist zu aufwändig, wenn Integrale mit hoher Genauigkeit berechnet werden müssen. Beim Romberg-Verfahren verdoppelt sich der Aufwand bei jedem Halbierungsschritt. Die Situation kann etwas verbessert werden, wenn andere Schrittweitenfolgen benutzt werden, siehe etwa [Sto 02]. Der damit erzielbare Genauigkeitsgewinn wird aber von anderen Verfahrensklassen wie z.B. den Gauß-Regeln weit übertroffen, siehe Abschnitt 7.4.

## 7.3 Transformationsmethoden

In diesem Abschnitt behandeln wir solche Integrale, für welche im Fehlergesetz (7.24) der Trapezregel alle Terme endlicher Ordnung verschwinden. Das übrig bleibende Restglied ist dann exponentiell klein. Integrale mit dieser Eigenschaft treten nicht allzu selten auf und haben wichtige Anwendungen. Zudem können Integrale mit analytischen Integranden  $f(x)$  durch geeignete Transformationen auf die erwähnten Fälle zurückgeführt werden.

### 7.3.1 Periodische Integranden

Ein erster Fall, wo in (7.24) alle  $c_k$  verschwinden, liegt vor, wenn der in  $\mathbb{R}$  analytische Integrand  $f(x)$   $\tau$ -periodisch ist,

$$f(x + \tau) = f(x) \quad \text{für alle } x \in \mathbb{R},$$

und sich die Integration über eine volle Periode erstreckt. Ohne Einschränkung der Allgemeinheit setzen wir  $a = 0, b = \tau$ . Dann gilt

$$f^{(2k-1)}(b) - f^{(2k-1)}(a) = 0, \quad k = 1, 2, \dots \quad (7.31)$$

Für jedes  $N$  ist im Fehlergesetz (7.24) nur das Restglied vorhanden. Anstatt das Restglied in (7.24) für periodische Funktionen zu untersuchen, ist es einfacher, direkt den Fehler der Trapezsummen durch die Fourier-Reihe von  $f(x)$  auszudrücken. In komplexer Schreibweise mit  $i^2 = -1$  sei also

$$f(x) = \sum_{k=-\infty}^{\infty} f_k e^{ikx \frac{2\pi}{\tau}} \quad (7.32)$$

mit den komplexen Fourier-Koeffizienten

$$f_k := \frac{1}{\tau} \int_0^\tau f(x) e^{-ikx \frac{2\pi}{\tau}} dx. \quad (7.33)$$

Wegen  $f(\tau) = f(0)$  schreibt sich die Trapezsumme (7.13) bei  $n$  Teilintervallen als

$$T(h) = h \sum_{j=0}^{n-1} f(jh), \quad h = \frac{\tau}{n}, \quad n \in \mathbb{N}^*. \quad (7.34)$$

Setzen wir die Fourier-Reihe (7.32) in (7.34) ein und vertauschen die Summations-Reihenfolge, ergibt sich

$$T(h) = \frac{\tau}{n} \sum_{k=-\infty}^{\infty} f_k \sum_{j=0}^{n-1} e^{ijk \frac{2\pi}{n}}.$$

Von der Summe über  $k$  bleiben nur die Terme mit  $k = nl, l \in \mathbb{Z}$ , übrig, weil

$$\sum_{j=0}^{n-1} e^{ijk \frac{2\pi}{n}} = \begin{cases} n, & \text{für } k \equiv 0 \pmod{n} \\ 0, & \text{sonst} \end{cases},$$

und wir erhalten

$$T(h) = \tau \sum_{l=-\infty}^{\infty} f_{nl}. \quad (7.35)$$

Speziell gilt gemäß (7.33)  $\tau f_0 = \int_0^\tau f(x) dx = I$ , und somit ergibt sich aus (7.35) das Fehlergesetz

$$T(h) - I = \tau(f_n + f_{-n} + f_{2n} + f_{-2n} + \dots). \quad (7.36)$$

Zur weiteren Diskussion benutzen wir aus der komplexen Analysis den folgenden [Hen 91]

**Satz 7.3.** Sei  $f(z)$   $\tau$ -periodisch und analytisch im Streifen  $|Im(z)| < \omega$ ,  $0 < \omega < \infty$ , wobei der Rand des Streifens Singularitäten von  $f(z)$  enthält. Dann klingen die Fourier-Koeffizienten  $f_k$  (7.33) von  $f(z)$  ab wie eine geometrische Folge gemäß

$$|f_k| = O\left(e^{-|k|(\omega-\varepsilon)\frac{2\pi}{\tau}}\right), \quad k \rightarrow \infty, \quad \varepsilon > 0.$$

Für Funktionen  $f(x)$ , welche die Voraussetzungen des Satzes 7.3 erfüllen, folgt aus (7.36) wegen  $h = \tau/n$

$$|T(h) - I| = O\left(e^{-(\omega-\varepsilon)\frac{2\pi}{h}}\right), \quad h > 0, \varepsilon > 0.$$

(7.37)

Somit nimmt der Fehler der Trapezsumme  $T(h)$  mit abnehmender Schrittänge  $h$  exponentiell ab. Für hinreichend kleines  $h$  bewirkt die Halbierung der Schrittänge etwa die Quadraturierung des Fehlers. Die Anzahl der richtigen Dezimalstellen nimmt also etwa proportional zum geleisteten Rechenaufwand zu.

Der Algorithmus (7.20) eignet sich gut zur genäherten Berechnung von solchen Integralen. Um dem Konvergenzverhalten Rechnung zu tragen, kann als Abbruchkriterium in (7.20) die Bedingung  $|T - M| < \sqrt{\varepsilon}$  verwendet werden, falls die zuletzt berechnete Trapezsumme etwa den Fehler  $\varepsilon$  gegenüber  $I$  aufweisen darf.

Integrale der diskutierten Art treten in der Praxis recht häufig auf. Zu nennen wären beispielsweise die Berechnung der Oberfläche eines Ellipsoïdes, die Integraldarstellung der Bessel-Funktionen, die Berechnung der reellen Fourier-Koeffizienten und die Bestimmung von Mittelwerten und Perioden.

**Beispiel 7.6.** Der Umfang  $U$  einer Ellipse mit den Halbachsen  $A$  und  $B$  mit  $0 < B < A$  ist gegeben durch

$$U = \int_0^{2\pi} \sqrt{A^2 \sin^2 \varphi + B^2 \cos^2 \varphi} d\varphi = 4A \int_0^{\pi/2} \sqrt{1 - e^2 \cos^2 \varphi} d\varphi. \quad (7.38)$$

wo  $e := \sqrt{A^2 - B^2}/A$  ihre Exzentrizität ist. Wir berechnen die Trapezsummen mit den Schrittängen  $h = 2\pi/n$ , ( $n = 8, 16, 24, \dots, 64$ ) für  $A = 1$ ,  $B = 0.25$  und damit  $e \doteq 0.968246$ . In Tab. 7.2 sind die Ergebnisse zusammengestellt.

Die  $q$ -Werte sind die Quotienten von aufeinander folgenden Fehlern. Zur besseren Illustration des Konvergenzverhaltens wurde die Zahl  $n$  der Teilintervalle in arithmetischer Folge mit der Differenz  $d = 8$  erhöht. Gemäß (7.37) verhalten sich dann die Fehler etwa wie eine geometrische Folge mit dem Quotienten  $e^{-d\omega} = e^{-8\omega}$ . Der Integrand (7.38) hat bei  $\varphi = \pm i\omega$  mit  $\cosh(\omega) = 4/\sqrt{15}$  Verzweigungspunkte und ist damit nur analytisch im Streifen  $|Im(\varphi)| < 0.2554128$ . Nach Satz 7.3 und (7.37) folgt daraus  $\lim_{n \rightarrow \infty} |q_n| = e^{-8\omega} \doteq 0.1296$  in guter Übereinstimmung mit den festgestellten Quotienten in Tab. 7.2. Infolge des kleinen Wertes von  $\omega$  für das gewählte Achsenverhältnis ist die Konvergenz der Trapezsummen relativ langsam. Für ein Achsenverhältnis  $A/B = 2$  liefern bereits  $n = 32$  Teilintervalle, d.h. acht Teilintervalle der Viertelperiode zehnstellige Genauigkeit.  $\triangle$

Tab. 7.2 Trapezsummen für periodischen Integranden.

$n$	$T\left(\frac{2\pi}{n}\right)$	$q_n$
8	4.2533048630	
16	4.2877583000	0.0405
24	4.2891119296	0.0681
32	4.2892026897	0.0828
40	4.2892101345	0.0919
48	4.2892108138	0.0980
56	4.2892108800	0.1024
64	4.2892108868	0.1057

### 7.3.2 Integrale über $\mathbb{R}$

Den zweiten Spezialfall des Fehlergesetzes (7.24) erhalten wir für uneigentliche Integrale der Form

$$I = \int_{-\infty}^{\infty} f(x) dx. \quad (7.39)$$

Dabei sei  $f(x)$  absolut integrierbar und auf der ganzen reellen Achse analytisch. Zudem soll  $f^{(k)}(a) \rightarrow 0$  für  $a \rightarrow \pm\infty$ ,  $k = 0, 1, 2, \dots$ , gelten. Der formale Grenzübergang  $a \rightarrow -\infty$  und  $b \rightarrow \infty$  in (7.23) lässt erwarten, dass die Trapezintegration (7.22) für die Berechnung des Integrals (7.39) ebenfalls besonders gute Approximationen liefert. Dies ist in der Tat der Fall, denn das Fehlergesetz wird jetzt durch die *Poissonsche Summenformel* geliefert. Wir begnügen uns mit einer formalen Diskussion und verweisen für eine strenge Behandlung auf [Hen 91]. Die zu (7.39) definierte Trapezsumme (7.21)

$$T(h, s) := h \sum_{j=-\infty}^{\infty} f(jh + s) \quad (7.40)$$

ist als Funktion von  $s$  periodisch mit der Periode  $h$  und kann deshalb als Fourier-Reihe

$$T(h, s) = \sum_{k=-\infty}^{\infty} t_k e^{iks \frac{2\pi}{h}} \quad (7.41)$$

mit den Fourier-Koeffizienten

$$t_k = \frac{1}{h} \int_0^h T(h, s) e^{-iks \frac{2\pi}{h}} ds \quad (7.42)$$

geschrieben werden. Einsetzen von (7.40) in (7.42) ergibt nach Vertauschung von Integration und Summation

$$t_k = \sum_{j=-\infty}^{\infty} \int_0^h f(jh + s) e^{-iks \frac{2\pi}{h}} ds = \int_{-\infty}^{\infty} f(s) e^{-iks \frac{2\pi}{h}} ds.$$

Führen wir das Fourier-Integral

$$g(t) := \int_{-\infty}^{\infty} f(s) e^{-ist} ds \quad (7.43)$$

des Integranden  $f(s)$  ein, so erhalten wir aus (7.41) die *Poissonsche Summenformel*

$$T(h, s) = \text{HW} \left\{ \sum_{k=-\infty}^{\infty} g\left(k \frac{2\pi}{h}\right) e^{isk \frac{2\pi}{h}} \right\}, \quad (7.44)$$

wobei HW für den Hauptwert steht, der bei symmetrischer Bildung der unendlichen Summe resultiert. Nun ist speziell  $g(0) = I$ , und somit folgt aus (7.44)

$$T(h, s) - I = \text{HW} \left\{ \sum_{k \neq 0} g\left(k \frac{2\pi}{h}\right) e^{isk \frac{2\pi}{h}} \right\}. \quad (7.45)$$

Für das Verhalten des Fehlers bei  $h \rightarrow 0$  ist das Verhalten des Fourier-Integrals (7.43) bei  $t \rightarrow \infty$  maßgebend. Dazu gilt

**Satz 7.4.** Sei  $f(z)$  eine über  $\mathbb{R}$  integrierbare, im Streifen  $|Im(z)| < \omega$ ,  $0 < \omega < \infty$ , analytische Funktion, wobei der Rand des Streifens Singularitäten von  $f(z)$  enthält. Dann gilt für das Fourier-Integral (7.43) asymptotisch  $|g(t)| = O(e^{-|t|(\omega-\varepsilon)})$  für  $|t| \rightarrow \infty$  und jedes  $\varepsilon > 0$ .

Auf Grund von Satz 7.4 folgt aus (7.45)

$$|T(h, s) - I| = O\left(e^{-(\omega-\varepsilon)\frac{2\pi}{h}}\right), \quad h > 0, \varepsilon > 0. \quad (7.46)$$

In formaler Übereinstimmung mit (7.37) ist der Fehler der Trapezsumme wiederum für  $h \rightarrow 0$  exponentiell klein.

**Beispiel 7.7.** Für  $f(x) = e^{-x^2/2}$  sind

$$I = \int_{-\infty}^{\infty} e^{-x^2/2} dx = \sqrt{2\pi} \doteq 2.50662827463, \quad g(t) = \sqrt{2\pi} e^{-t^2/2}. \quad (7.47)$$

Mit dem Verfahren (7.22) erhalten wir  $T(2, 0) = 2.542683044$ ,  $T(1, 0) = 2.506628288$ ,  $T\left(\frac{1}{2}, 0\right) = 2.506628275$ . Da der Integrand sehr rasch abklingt, liefert die Summation im Intervall  $[-7, 7]$  schon zehnstellige Genauigkeit. Bei hinreichend hoher Rechengenauigkeit wäre gemäß (7.45) und (7.47) der Fehler in  $T\left(\frac{1}{2}, 0\right)$  betragsmäßig kleiner als  $3 \cdot 10^{-34}$ .  $\triangle$

Die rasche Konvergenz von  $T(h, s)$  bezüglich Verkleinerung von  $h$  besteht nach (7.46) auch in Fällen von langsam abklingendem  $f(x)$ . Allerdings wird dann die Berechnung von  $T(h, s)$  nach (7.22) sehr aufwändig.

**Beispiel 7.8.** Für die Funktion  $f(x) = 1/(1+x^2)$  ist das Fourier-Integral (7.43)  $g(t) = \pi e^{-|t|}$ . Somit besitzen die Trapezsummen für das uneigentliche Integral nach (7.44) die explizite Darstellung

$$T(h, s) = \pi + 2\pi \sum_{k=1}^{\infty} e^{-k \frac{2\pi}{h}} \cos\left( sk \frac{2\pi}{h} \right).$$

Aus dieser expliziten Formel berechnen sich die Werte  $T(2, 0) = 3.4253772$ ,  $T(1, 0) = 3.1533481$ ,  $T\left(\frac{1}{2}, 0\right) = 3.141614566$ ,  $T\left(\frac{1}{4}, 0\right) = 3.141592654$ . Die rasche Konvergenz der Trapezsummen gegen den Wert  $I = \pi$  besteht auch hier, doch wären zur Berechnung von  $T(h, 0)$  nach (7.22) mit derselben Genauigkeit rund  $10^{10}$  Terme nötig.  $\triangle$

### 7.3.3 Variablensubstitution

Die klassische Technik der Variablensubstitution soll im folgenden dazu eingesetzt werden, das Integral (7.1) so zu transformieren, dass es mit einer schnell konvergenten Quadraturmethode ausgewertet werden kann. Wir definieren die Substitution durch

$$x = \varphi(t), \quad \varphi'(t) > 0, \tag{7.48}$$

wo  $\varphi(t)$  eine geeignet gewählte, einfach berechenbare und streng monotone, analytische Funktion ist. Ihre Inverse bildet das Integrationsintervall  $[a, b]$  bijektiv auf das Intervall  $[\alpha, \beta]$  mit  $\varphi(\alpha) = a$ ,  $\varphi(\beta) = b$  ab. Damit erhalten wir

$$I = \int_a^b f(x) dx = \int_{\alpha}^{\beta} F(t) dt \quad \text{mit } F(t) := f(\varphi(t))\varphi'(t). \tag{7.49}$$

Hauptanwendungen der Variablensubstitution sind die Behandlung von Integralen mit singulärem Integranden und von Integralen über unbeschränkte Intervalle mit schwach abklingenden Integranden. Die ersten Ansätze dieser Methoden gehen zurück auf [Goo 49, Sag 64, Sch 69, Ste 73a, Tak 73]. Im Folgenden stellen wir einige der wichtigsten Substitutionen vor.

a) *Algebraische Substitution.* Als Modell eines Integrals mit einer algebraischen Randsingularität betrachten wir

$$I = \int_0^1 x^{p/q} f(x) dx, \quad q = 2, 3, \dots; \quad p > -q, \quad p \in \mathbb{Z}, \tag{7.50}$$

wobei  $f(x)$  in  $[0, 1]$  analytisch sei. Die Bedingung für  $p$  und  $q$  garantiert die Existenz von  $I$ . Die Variablensubstitution

$$x = \varphi(t) = t^q, \quad \varphi'(t) = qt^{q-1} > 0 \text{ in } (0, 1)$$

führt (7.50) über in das Integral

$$I = q \int_0^1 t^{p+q-1} f(t^q) dt,$$

welches wegen  $p + q - 1 \geq 0$  existiert und keine Singularität aufweist. Es kann mit dem Romberg-Verfahren oder der Gauß-Integration (vgl. Abschnitt 7.4) effizient ausgewertet

werden. Das Integral  $I = \int_0^1 x^{3/2} dx$  von Beispiel 7.5 geht durch die Substitution  $x = t^2$  in das Integral  $I = 2 \int_0^1 t^4 dt$  mit polynomialem Integranden über, das auch numerisch problemlos zu berechnen ist.

b) *tanh-Substitution*. Sind integrierbare Singularitäten von unbekannter, eventuell von logarithmischer Natur an den beiden Intervallenden vorhanden, soll  $\varphi(t)$  in (7.48) so gewählt werden, dass das Integrationsintervall auf die ganze reelle Achse ( $\alpha = -\infty, \beta = \infty$ ) abgebildet wird. Um exponentielles Abklingen des transformierten Integranden zu begünstigen, soll  $\varphi(t)$  asymptotisch exponentiell gegen die Grenzwerte  $a$  und  $b$  streben. Für das Integral

$$I = \int_{-1}^1 f(x) dx$$

erfüllt beispielsweise die Substitution

$$x = \varphi(t) = \tanh(t), \quad \varphi'(t) = \frac{1}{\cosh^2(t)} \quad (7.51)$$

die gestellten Bedingungen. Das transformierte Integral ist

$$I = \int_{-\infty}^{\infty} F(t) dt \quad \text{mit} \quad F(t) = \frac{f(\tanh(t))}{\cosh^2(t)}. \quad (7.52)$$

Wegen des sehr rasch anwachsenden Nenners wird der Integrand  $F(t)$  für  $t \rightarrow \pm\infty$  meistens exponentiell abklingen, so dass das Integral (7.52) mit der Trapezregel (7.22) effizient berechnet werden kann. Die numerische Auswertung von  $F(t)$  muss aber sehr sorgfältig erfolgen. Denn für große Werte von  $|t|$  kann im Rahmen der Genauigkeit des Rechners  $\tanh(t) = \pm 1$  resultieren. Hat  $f(x)$  Randsingularitäten, könnte  $F(t)$  für große  $|t|$  aus diesem Grund nicht berechnet werden. Um diese numerische Schwierigkeit zu beheben, können beispielsweise die Relationen

$$\begin{aligned} \tanh(t) &= -1 + e^t / \cosh(t) = -1 + \xi, \quad t \leq 0, \\ \tanh(t) &= 1 - e^{-t} / \cosh(t) = 1 - \eta, \quad t \geq 0. \end{aligned}$$

zusammen mit lokal gültigen Entwicklungen für  $f(-1 + \xi)$  und  $f(1 - \eta)$  verwendet werden. Eine weitere Schwierigkeit der tanh-Substitution besteht darin, dass sehr große Zahlen erzeugt werden, so dass sie auf Rechnern mit kleinem Exponentenbereich versagen kann.

Durch die Substitution (7.51) wird nicht in jedem Fall garantiert, dass der transformierte Integrand  $F(t)$  beidseitig exponentiell abklingt. Ein Beispiel dazu ist

$$f(x) = \frac{x^2}{(1-x^2)\operatorname{Artanh}^2(x)} \quad \text{mit} \quad F(t) = \frac{\tanh^2(t)}{t^2}.$$

c) *sinh-Substitution*. Wir betrachten Integrale mit unbeschränktem Integrationsintervall, die wegen zu langsamem Abklingens von  $f(x)$  die Berechnung von zu vielen Termen in der Trapezregel erfordern. In diesen Fällen eignet sich die Variablensubstitution

$$x = \varphi(t) = \sinh(t), \quad (7.53)$$

so dass wir

$$I = \int_{-\infty}^{\infty} f(x) dx = \int_{-\infty}^{\infty} F(t) dt \quad \text{mit} \quad F(t) = f(\sinh(t)) \cosh(t) \quad (7.54)$$

erhalten. Nach einer endlichen Anzahl von sinh-Substitutionen nach (7.53) resultiert ein

beidseitig exponentiell abklingender Integrand, auf den die Trapezregel (7.22) effizient anwendbar ist. Meistens genügt ein Substitutionsschritt. Nur in sehr speziellen Fällen von integrierbaren Funktionen kann die gewünschte Eigenschaft nicht mit einer endlichen Anzahl von sinh-Substitutionen erreicht werden [Sze 62].

**Beispiel 7.9.** Mit der *sinh*-Substitution ergibt sich für das Integral von Beispiel 7.8

$$I = \int_{-\infty}^{\infty} \frac{dx}{1+x^2} = \int_{-\infty}^{\infty} \frac{dt}{\cosh(t)} = \pi.$$

Die Trapezsummen sind

$T(2, 0)$	$T(1, 0)$	$T(1/2, 0)$	$T(1/4, 0)$
3.232618532	3.142242660	3.141592687	3.141592654

Um die Trapezsummen mit zehnstelliger Genauigkeit zu erhalten, sind nur Werte  $|t| \leq 23$  zu berücksichtigen. Die Trapezsumme  $T(1/2, 0)$  ist bereits eine in acht Stellen genaue Näherung für  $I$ , und  $T(1/4, 0)$  wäre bei hinreichend genauer Rechnung sogar in sechzehn Stellen genau.  $\triangle$

d) *exp-Substitution.* Integrale mit halbunendlichen Intervallen  $(a, \infty)$  lassen sich mittels der einfachen Substitution

$$x = \varphi(t) = a + e^t$$

überführen in

$$I = \int_a^{\infty} f(x) dx = \int_{-\infty}^{\infty} f(a + e^t) e^t dt.$$

Ein Nachteil der Substitutionsmethode mag sein, dass ein Quadraturverfahren erst angewendet werden kann, nachdem von Hand Umformungen ausgeführt worden sind. Wir zeigen nun, dass bei gegebener Abbildung  $x = \varphi(t)$  die Substitution rein numerisch, d.h. mit Werten von  $\varphi(t)$  und  $\varphi'(t)$  ausgeführt werden kann. Zu diesem Zweck nehmen wir an, dass die genäherte Berechnung des transformierten Integrals (7.49) mit einer  $n$ -Punkte-Quadraturformel (7.1) mit Knoten  $t_j$ , Gewichten  $v_j$  und Restglied  $R_{n+1}$  erfolgt.

$$I = \int_{\alpha}^{\beta} F(t) dt = \sum_{j=1}^n v_j F(t_j) + R_{n+1} \quad (7.55)$$

Auf Grund der Definition von  $F(t)$  in (7.49) lässt sich (7.55) auch als Quadraturformel für das ursprüngliche Integral interpretieren mit Knoten  $x_j$  und Gewichten  $w_j$ , nämlich

$$\tilde{I} = \sum_{j=1}^n w_j f(x_j), \quad x_j := \varphi(t_j), \quad w_j := v_j \varphi'(t_j), \quad j = 1, 2, \dots, n. \quad (7.56)$$

Am einfachsten ist es, die Werte von  $x_j$  und  $w_j$  direkt zu berechnen, wenn sie gebraucht werden. Dies ist wegen der Wahl von  $\varphi(t)$  als einfache Funktion meistens sehr effizient möglich. Um die Rechnung noch effizienter zu gestalten, kann aber auch eine Tabelle  $x_j$  und  $w_j$  bereitgestellt werden, auf die bei der Anwendung der Quadraturformel (7.56) zugegriffen wird. Weitere und insbesondere speziellere Realisierungen dieser Idee findet man in [Iri 70, Mor 78, Tak 74].

## 7.4 Gauß-Integration

Bei den bisher betrachteten Quadraturformeln sind wir stets von vorgegebenen (meistens äquidistanten) Stützstellen ausgegangen. Ein Ansatz, der auf Gauß zurückgeht, benutzt sowohl die Integrationsgewichte als auch die Stützstellen zur Konstruktion der Quadraturformel und übertrifft damit für glatte Funktionen noch den hohen Genauigkeitsgrad der Romberg-Integration. In seiner allgemeinen Formulierung erlaubt er darüberhinaus die Entwicklung von Quadraturformeln für spezielle Klassen von Integranden. Wir beschränken uns zunächst auf das Intervall  $[-1, 1]$ . Diese Festlegung geschieht wegen gewisser Eigenschaften der Legendre-Polynome, die hier ausgenutzt werden. Jedes endliche Intervall  $[a, b]$  lässt sich mit Hilfe einer linearen Transformation auf das Intervall  $[-1, 1]$  transformieren, sodass die Entwicklung der Quadraturformeln für  $[-1, 1]$  keine Einschränkung darstellt.

Bestimme für die numerische Integrationsformel

$$\int_{-1}^1 f(x) dx = \sum_{i=1}^n w_i f(x_i) + E_n[f] = Q_n + E_n[f], \quad x_i \in [-1, 1] \quad (7.57)$$

die Integrationsgewichte  $w_i$  und die Stützstellen  $x_i$  so, dass ein Polynom möglichst hohen Grades exakt integriert wird.

Die Newton-Cotes-Formeln integrieren bei  $n$  Stützstellen ein Polynom  $(n-1)$ -ten Höchstgrades exakt; die Gauß-Integration schafft das bis zum Polynomgrad  $2n-1$ .

**Beispiel 7.10.** :  $n = 2$

$$\int_{-1}^1 f(x) dx \approx \sum_{i=1}^2 w_i f(x_i). \quad (7.58)$$

Zur Konstruktion dieser Gauß-Formel müssen  $x_1$ ,  $x_2$ ,  $w_1$  und  $w_2$  so bestimmt werden, dass für jedes Polynom 3. Grades  $p(x)$

$$\int_{-1}^1 p(x) dx = \int_{-1}^1 (a_0 + a_1 x + a_2 x^2 + a_3 x^3) dx = w_1 p(x_1) + w_2 p(x_2) \quad (7.59)$$

gilt. Integration und Koeffizientenvergleich ergeben

$$\begin{aligned} w_1 &= 1, & w_2 &= 1, \\ x_1 &= \frac{-1}{\sqrt{3}}, & x_2 &= \frac{1}{\sqrt{3}}. \end{aligned}$$

Das ergibt die Quadraturformel

$$\int_{-1}^1 f(x) dx \approx f\left(\frac{-1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right). \quad (7.60)$$

Diese Formel lässt sich leicht auf ein allgemeines Intervall transformieren:

$$\begin{aligned} \int_a^b f(x) dx &\approx \frac{b-a}{2} (f(u_1) + f(u_2)) \quad \text{mit} \\ u_i &= \frac{a+b}{2} + \frac{b-a}{2} x_i, \quad i = 1, 2. \end{aligned} \quad (7.61)$$

△

**Satz 7.5.** Der Genauigkeitsgrad einer Quadraturformel (7.57) mit  $n$  Knoten ist höchstens  $(2n - 1)$ .

*Beweis.* Wir betrachten das Polynom vom Grad  $2n$

$$q(x) := \prod_{k=1}^n (x - x_k)^2,$$

welches mit den  $n$  paarweise verschiedenen Integrationsstützstellen der Quadraturformel gebildet wird. Für das nicht identisch verschwindende Polynom gilt  $q(x) \geq 0$  für alle  $x \in [-1, 1]$ , und folglich ist

$$I = \int_{-1}^1 q(x) dx > 0.$$

Die Quadraturformel (7.57) ergibt jedoch wegen  $q(x_k) = 0$ ,  $k = 1, 2, \dots, n$ , den Wert  $Q = 0$ , d.h. es ist  $E[q] \neq 0$ , und ihr Genauigkeitsgrad muss kleiner als  $2n$  sein.  $\square$

**Satz 7.6.** Es existiert genau eine Quadraturformel

$$Q = \sum_{k=1}^n w_k f(x_k), \quad x_k \in [-1, 1], \quad (7.62)$$

mit  $n$  Integrationsstützstellen  $x_k$ , die den maximalen Genauigkeitsgrad  $(2n - 1)$  besitzt. Die Stützstellen  $x_k$  sind die Nullstellen des  $n$ -ten Legendre-Polynoms  $P_n(x)$  (3.233), und die Integrationsgewichte sind gegeben durch

$$w_k = \int_{-1}^1 \prod_{\substack{j=1 \\ j \neq k}}^n \left( \frac{x - x_j}{x_k - x_j} \right)^2 dx > 0, \quad k = 1, 2, \dots, n. \quad (7.63)$$

*Beweis.* Die verschiedenen Aussagen beweisen wir in drei Teilen.

a) Zuerst befassen wir uns mit der Existenz einer Quadraturformel vom Genauigkeitsgrad  $(2n - 1)$ . Zu diesem Zweck benutzen wir die Tatsache, dass das Legendre-Polynom  $P_n(x)$  nach Satz 3.39  $n$  einfache Nullstellen  $x_1, x_2, \dots, x_n$  im Innern des Intervalls  $[-1, 1]$  besitzt. Zu diesen  $n$  paarweise verschiedenen Stützstellen existiert nach Satz 7.2 eine eindeutig bestimmte Newton-Cotes-Formel, deren Genauigkeitsgrad mindestens gleich  $(n - 1)$  ist.

Es sei  $p(x)$  ein beliebiges Polynom, dessen Grad höchstens gleich  $(2n - 1)$  ist. Wird  $p(x)$  durch das  $n$ -te Legendre-Polynom  $P_n(x)$  dividiert mit Rest, so erhalten wir folgende Darstellung

$$p(x) = q(x)P_n(x) + r(x) \quad (7.64)$$

mit  $\text{Grad } (q(x)) \leq n - 1$  und  $\text{Grad } (r(x)) \leq n - 1$ . Damit ergibt sich

$$\int_{-1}^1 p(x) dx = \int_{-1}^1 q(x) P_n(x) dx + \int_{-1}^1 r(x) dx = \int_{-1}^1 r(x) dx, \quad (7.65)$$

da ja das Legendre-Polynom  $P_n(x)$  auf Grund der Orthogonalitätseigenschaften (3.234) zu allen Legendre-Polynomen kleineren Grades  $P_0(x), P_1(x), \dots, P_{n-1}(x)$  und folglich zu  $q(x)$  orthogonal ist.

Für die Newton-Cotes-Formel zu den Stützstellen  $x_k$  mit den zugehörigen Gewichten  $w_k$  gilt für das Polynom  $p(x)$  wegen (7.64), wegen  $P_n(x_k) = 0$  und gemäß (7.65)

$$\begin{aligned} \sum_{k=1}^n w_k p(x_k) &= \sum_{k=1}^n w_k q(x_k) P_n(x_k) + \sum_{k=1}^n w_k r(x_k) = \sum_{k=1}^n w_k r(x_k) \\ &= \int_{-1}^1 r(x) dx = \int_{-1}^1 p(x) dx. \end{aligned} \quad (7.66)$$

Die zweitletzte Gleichung in (7.66) beruht auf der Tatsache, dass die Newton-Cotes-Formel mindestens den Genauigkeitsgrad  $(n - 1)$  besitzt. Somit ist gezeigt, dass die Quadraturformel (7.62) für jedes Polynom vom Grad kleiner  $2n$  exakt ist. Wegen Satz 7.5 ist der Genauigkeitsgrad maximal.

b) Die Integrationsgewichte  $w_k$  der Newton-Cotes-Formel sind wegen des gegenüber (7.3) leicht geänderten Ansatzes gegeben durch

$$w_k = \int_{-1}^1 L_k(x) dx = \int_{-1}^1 \prod_{\substack{j=1 \\ j \neq k}}^n \left( \frac{x - x_j}{x_k - x_j} \right) dx, \quad k = 1, 2, \dots, n, \quad (7.67)$$

wo  $L_k(x)$  das Lagrange-Polynom zu den Stützstellen  $x_1, \dots, x_n$  mit  $L_k(x_j) = \delta_{kj}$  vom Grad  $(n - 1)$  darstellt. Diese Darstellung führt nicht weiter. Deshalb nutzen wir die bereits bewiesene Tatsache aus, dass die zu den Stützstellen  $x_k$  gehörige Newton-Cotes-Formel den Genauigkeitsgrad  $(2n - 1)$  besitzt, und deshalb für das Polynom  $L_k^2(x)$  vom Grad  $(2n - 2)$  den exakten Wert liefert. Folglich gilt für  $k = 1, 2, \dots, n$

$$0 < \int_{-1}^1 L_k^2(x) dx = \int_{-1}^1 \prod_{\substack{j=1 \\ j \neq k}}^n \left( \frac{x - x_j}{x_k - x_j} \right)^2 dx = \sum_{\mu=1}^n w_\mu L_k^2(x_\mu) = w_k. \quad (7.68)$$

Damit ist (7.63) bewiesen. Insbesondere folgt aus dieser Darstellung, dass die Gewichte  $w_k$  für jedes  $n \in \mathbb{N}^*$  positiv sind.

c) Um die Eindeutigkeit der Quadraturformel zu beweisen, nehmen wir an, es existiere eine weitere Formel

$$Q^* := \sum_{k=1}^n w_k^* f(x_k^*), \quad x_k^* \neq x_j^* \text{ für alle } k \neq j, \quad (7.69)$$

deren Genauigkeitsgrad ebenfalls gleich  $(2n-1)$  ist. Auf Grund der Betrachtungen im Teil b) gilt auch für (7.69), dass die Gewichte  $w_k^* > 0$ ,  $k = 1, 2, \dots, n$ , sind. Wir wollen zeigen, dass die Integrationsstützstellen  $x_k^*$  unter den genannten Bedingungen bis auf eine Permutation mit den  $x_k$  von (7.62) übereinstimmen. Dazu betrachten wir das Hilfspolynom

$$h(x) := L_k^*(x)P_n(x), \quad L_k^*(x) := \prod_{\substack{j=1 \\ j \neq k}}^n \left( \frac{x - x_j^*}{x_k^* - x_j^*} \right), \quad \text{Grad}(h(x)) = 2n-1.$$

Wegen unserer Annahme liefert die Quadraturformel (7.69) den exakten Wert des Integrals für  $h(x)$ , und somit gilt für  $k = 1, 2, \dots, n$

$$\begin{aligned} 0 = \int_{-1}^1 h(x) dx &= \int_{-1}^1 L_k^*(x)P_n(x) dx = \sum_{\mu=1}^n w_\mu^* L_k^*(x_\mu^*) P_n(x_\mu^*) \\ &= w_k^* P_n(x_k^*), \end{aligned} \quad (7.70)$$

denn das zweite Integral ist infolge der Orthogonalität von  $P_n(x)$  zu allen Polynomen vom Grad kleiner als  $n$  gleich null. Da aber  $w_k^* > 0$  ist, muss  $P_n(x_k^*) = 0$ ,  $k = 1, 2, \dots, n$ , gelten, d.h. die Integrationsstützstellen  $x_k^*$  von (7.69) müssen notwendigerweise die Nullstellen des  $n$ -ten Legendre-Polynoms  $P_n(x)$  sein. Sie sind damit eindeutig festgelegt. Für die zugehörige Quadraturformel sind auch die Gewichte durch (7.67) eindeutig bestimmt.  $\square$

Die nach Satz 7.6 charakterisierten Integrationsmethoden mit maximalem Genauigkeitsgrad heißen *Gaußsche Quadraturformeln*. Für ihre hohe Genauigkeit muss man zwei Nachteile in Kauf nehmen:

- Die Bestimmung der Koeffizienten und Stützstellen hängt vom Integrationsintervall ab.
- Es ergeben sich für jedes  $n$  andere Koeffizienten und Stützstellen.

Der erste Nachteil lässt sich leicht durch die lineare Transformation

$$t = \frac{b-a}{2}x + \frac{a+b}{2} \quad (7.71)$$

ausräumen. Es ist

$$I = \int_a^b f(t) dt = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{b-a}{2}x + \frac{a+b}{2}\right) dx,$$

und die Quadraturformel erhält damit die Gestalt

$$Q = \frac{b-a}{2} \sum_{k=1}^n w_k f\left(\frac{b-a}{2}x_k + \frac{a+b}{2}\right). \quad (7.72)$$

Die Rechenschritte fassen wir im Algorithmus Tab. 7.3 zusammen.

Tab. 7.3 Gauß-Legendre-Integration über das Intervall  $[a, b]$ .

(1) Stützstellen-Transformation
Für $i = 1, \dots, n$
$t_i = \frac{a+b}{2} + \frac{b-a}{2}x_i$
(2) Transformiertes Integral
$\hat{I} := \sum_{i=1}^n w_i f(t_i)$
(3) Transformation auf $[-1, 1]$
$\tilde{I} = \hat{I} \frac{b-a}{2}$

Der zweite Nachteil wiegt schwerer. Zur rechnerischen Anwendung der Gaußschen Quadraturformeln werden die für jedes  $n$  unterschiedlichen Stützstellen  $x_k$  und Gewichte  $w_k$  als Zahlen in ausreichender Genauigkeit (meist 15 Dezimalstellen) benötigt. Um die Genauigkeit über  $n$  steuern zu können, müssen also sehr viele Zahlen verwaltet oder jeweils neu berechnet werden. Sie sind in einschlägigen Tabellen oder Programmbibliotheken enthalten. Nun kann zunächst berücksichtigt werden, dass die von null verschiedenen Stützstellen  $x_k$  paarweise symmetrisch zum Nullpunkt liegen. Wegen (7.63) sind die Gewichte für diese Paare gleich. Deshalb genügt die Angabe der nichtnegativen Knoten  $x_k$  und ihrer Gewichte  $w_k$ . In der Regel werden die Stützstellen  $x_k$  in absteigender Reihenfolge  $1 > x_1 > x_2 > \dots$  tabelliert [Abr 71, Sch 76, Str 74, Str 66].

Wir wollen aber hier noch eine numerisch stabile Methode darstellen, um die Nullstellen  $x_k$  des  $n$ -ten Legendre-Polynoms und die Integrationsgewichte  $w_k$  zu berechnen [Gau 70, Gol 69]. Als Grundlage dient der

**Satz 7.7.** Das  $n$ -te Legendre-Polynom  $P_n(x)$ ,  $n \geq 1$  ist gleich der Determinante  $n$ -ter Ordnung

$$P_n(x) = \begin{vmatrix} a_1 x & b_1 & & & \\ b_1 & a_2 x & b_2 & & \\ & b_2 & a_3 x & b_3 & \\ & & \ddots & \ddots & \ddots \\ & & & b_{n-2} & a_{n-1} x & b_{n-1} \\ & & & & b_{n-1} & a_n x \end{vmatrix}, \quad \begin{aligned} a_k &= \frac{2k-1}{k}, \\ b_k &= \sqrt{\frac{k}{k+1}}, \\ n &= 1, 2, 3, \dots \end{aligned} \quad (7.73)$$

*Beweis.* Wir zeigen, dass die Determinanten für drei aufeinanderfolgende Werte von  $n$  die Rekursionsformel (3.239) der Legendre-Polynome erfüllen. Dazu wird die Determinante

(7.73) nach der letzten Zeile entwickelt und wir erhalten

$$P_n(x) = a_n x P_{n-1}(x) - b_{n-1}^2 P_{n-2}(x), \quad n \geq 3. \quad (7.74)$$

Ersetzen wir darin  $n$  durch  $(n+1)$  und beachten die Definition der  $a_k$  und  $b_k$ , folgt in der Tat die bekannte Rekursionsformel (3.239), die mit  $P_0(x) = 1$  auch für  $n=1$  gilt.  $\square$

Nach Satz 7.6 sind die  $x_k$  die Nullstellen der Determinante (7.73). Diese können als Eigenwerte einer symmetrischen, tridiagonalen Matrix  $\mathbf{J}_n$  erhalten werden. Um dies einzusehen, eliminieren wir die verschiedenen Koeffizienten  $a_i$  in der Diagonale derart, dass die Symmetrie der Determinante erhalten bleibt. Dazu werden die  $k$ -te Zeile und Spalte durch  $\sqrt{a_k} = \sqrt{(2k-1)/k}$  dividiert, und aus (7.73) folgt

$$P_n(x) = \begin{vmatrix} x & \beta_1 & & & \\ \beta_1 & x & \beta_2 & & \\ & \beta_2 & x & \beta_3 & \\ & & \ddots & \ddots & \ddots \\ & & & \beta_{n-2} & x & \beta_{n-1} \\ & & & & \beta_{n-1} & x \end{vmatrix} \cdot \prod_{k=1}^n a_k. \quad (7.75)$$

Die Nebendiagonalelemente in (7.75) sind für  $k = 1, 2, \dots, n-1$

$$\beta_k = \frac{b_k}{\sqrt{a_k a_{k+1}}} = \sqrt{\frac{k \cdot k \cdot (k+1)}{(k+1)(2k-1)(2k+1)}} = \frac{k}{\sqrt{4k^2 - 1}}. \quad (7.76)$$

Da die Nullstellen  $x_k \neq 0$  von  $P_n(x)$  paarweise entgegengesetztes Vorzeichen haben, sind sie die Eigenwerte der symmetrischen, tridiagonalen Matrix

$$\mathbf{J}_n = \begin{pmatrix} 0 & \beta_1 & & & \\ \beta_1 & 0 & \beta_2 & & \\ & \beta_2 & 0 & \beta_3 & \\ & & \ddots & \ddots & \ddots \\ & & & \beta_{n-2} & 0 & \beta_{n-1} \\ & & & & \beta_{n-1} & 0 \end{pmatrix} \in \mathbb{R}^{n,n}, \quad (7.77)$$

die mit dem QR-Algorithmus (vgl. Abschnitt 5.5) stabil und effizient berechnet werden können. Aber auch die Integrationsgewichte  $w_k$  lassen sich mit Hilfe der Eigenwertaufgabe für  $\mathbf{J}_n$  berechnen. Um diese Verbindung herzustellen, verifizieren wir, dass

$$\begin{aligned} \mathbf{z}^{(k)} &:= (\alpha_0 \sqrt{a_1} P_0(x_k), \alpha_1 \sqrt{a_2} P_1(x_k), \alpha_2 \sqrt{a_3} P_2(x_k), \dots, \alpha_{n-1} \sqrt{a_n} P_{n-1}(x_k))^T \in \mathbb{R}^n \\ \text{mit } \alpha_0 &:= 1, \alpha_j := 1 / \prod_{l=1}^j b_l, \quad j = 1, 2, \dots, n-1, \end{aligned} \quad (7.78)$$

für  $k = 1, 2, \dots, n$  Eigenvektor von  $\mathbf{J}_n$  zum Eigenwert  $x_k$  ist. Für die erste Komponente von  $\mathbf{J}_n \mathbf{z}^{(k)}$  gilt wegen (7.73), (7.76) und (7.78)

$$\alpha_1 \beta_1 \sqrt{a_2} P_1(x_k) = x_k = x_k \{\alpha_0 \sqrt{a_1} P_0(x_k)\}.$$

Für die  $i$ -te Komponente,  $1 < i < n$ , erhalten wir nach mehreren Substitutionen und wegen (7.74)

$$\begin{aligned} & \alpha_{i-2}\beta_{i-1}\sqrt{a_{i-1}}P_{i-2}(x_k) + \alpha_i\beta_i\sqrt{a_{i+1}}P_i(x_k) \\ &= \frac{\alpha_{i-1}}{\sqrt{a_i}}\{P_i(x_k) + b_{i-1}^2P_{i-2}(x_k)\} = x_k\{\alpha_{i-1}\sqrt{a_i}P_{i-1}(x_k)\}. \end{aligned}$$

Diese Beziehung bleibt auch für die letzte Komponente mit  $i = n$  wegen  $P_n(x_k) = 0$  gültig.

Weiter benutzen wir die Tatsachen, dass die Legendre-Polynome  $P_0(x), P_1(x), \dots, P_{n-1}(x)$  durch die Gaußsche Quadratformel exakt integriert werden und dass die Orthogonalitätseigenschaft (3.234) gilt.

$$\begin{aligned} \int_{-1}^1 P_0(x)P_i(x) dx &= \int_{-1}^1 P_i(x) dx \\ &= \sum_{k=1}^n w_k P_i(x_k) = \begin{cases} 2 & \text{für } i = 0, \\ 0 & \text{für } i = 1, 2, \dots, n-1. \end{cases} \end{aligned} \quad (7.79)$$

Die Gewichte  $w_k$  erfüllen somit das Gleichungssystem (7.79). Wenn wir die erste Gleichung mit  $\alpha_0\sqrt{a_1} = 1$ , und die  $j$ -te Gleichung,  $j = 2, 3, \dots, n$ , mit  $\alpha_{j-1}\sqrt{a_j} \neq 0$  multiplizieren, dann enthält die Matrix  $C$  des resultierenden Systems

$$Cw = 2e_1, \quad w := (w_1, w_2, \dots, w_n)^T, \quad e_1 = (1, 0, 0, \dots, 0)^T \quad (7.80)$$

als Spalten die Eigenvektoren  $z^{(k)}$  (7.78). Als Eigenvektoren der symmetrischen Matrix  $J_n$  zu paarweise verschiedenen Eigenwerten sind sie paarweise orthogonal. Multiplizieren wir (7.80) von links mit  $z^{(k)T}$ , so ergibt sich

$$(z^{(k)T} z^{(k)})w_k = 2z^{(k)T} e_1 = 2z_1^{(k)} = 2, \quad (7.81)$$

wo  $z_1^{(k)} = 1$  die erste Komponente des nicht normierten Eigenvektors  $z^{(k)}$  (7.78) ist. Mit dem normierten Eigenvektor  $\tilde{z}^{(k)}$  folgt aus (7.81)

$w_k = 2(\tilde{z}_1^{(k)})^2, \quad k = 1, 2, \dots, n.$

(7.82)

Aus (7.81) oder (7.82) folgt wiederum, dass die Gewichte  $w_k$  der Gaußschen Quadraturformel für alle  $n \in \mathbb{N}^*$  positiv sind.

In Tab. 7.4 sind für  $n = 1, 2, 3, 4, 5$  die Stützstellen und Gewichte angegeben.

**Beispiel 7.11.** Um den hohen Genauigkeitsgrad der Gaußschen Quadraturformel darzulegen, berechnen wir

$$I_1 = \int_0^1 \frac{4}{1+x^2} dx = \pi, \quad I_2 = \int_0^{\pi/2} x \cos(x) dx = \frac{\pi}{2} - 1$$

für einige Werte von  $n$ . In Tab. 7.5 sind die Ergebnisse mit den Quadraturfehlern zusammengestellt. Die Rechnung erfolgte vierzehnstellig mit entsprechend genauen Knoten  $x_k$  und Gewichten  $w_k$ .  $\triangle$

**Beispiel 7.12.** Wir wollen Beispiel 7.2 aufgreifen, mit der Gauß-Legendre-Regel zwei Näherungen für das Integral berechnen und dann in einer Tabelle die Integrationsregeln vergleichen, die wir

Tab. 7.4 Knoten und Gewichte der Gaußschen Integration.

$n = 2$	$x_1 \doteq 0.57735026918963,$	$w_1 = 1$
$n = 3$	$x_1 \doteq 0.77459666924148,$ $x_2 = 0,$	$w_1 \doteq 0.55555555555556$ $w_2 \doteq 0.88888888888889$
$n = 4$	$x_1 \doteq 0.86113631159405,$ $x_2 \doteq 0.33998104358486,$	$w_1 \doteq 0.34785484513745$ $w_2 \doteq 0.65214515486255$
$n = 5$	$x_1 \doteq 0.90617984593866,$ $x_2 \doteq 0.53846931010568,$ $x_3 = 0,$	$w_1 \doteq 0.23692688505619$ $w_2 \doteq 0.47862867049937$ $w_3 \doteq 0.56888888888889$

Tab. 7.5 Gauß-Integration.

$n$	$I_1$		$I_2$	
	$Q_n$	$E_n$	$Q_n$	$E_n$
2	3.1475409836	-0.0059483300	0.563562244208	0.007234082587
3	3.1410681400	0.0005245136	0.570851127976	-0.000054801181
4	3.1416119052	-0.0000192517	0.570796127158	0.000000199637
5	3.1415926399	0.0000000138	0.570796327221	-0.000000000426

bisher kennengelernt haben. Wir benutzen bei jeder Regel 3 bzw. 5 Funktionsauswertungen. Damit kommen wir zu den Ergebnissen in Tab. 7.6:

Tab. 7.6 Vergleich der Integrationsregeln.

Anz.Fktsausw.	Trapez	Simpson	Romberg	Gauß-Legendre
3	0.72473	0.90439	0.90439	1.001545
5	0.925565	0.992511	0.998386	0.99999986

Da der exakte Integralwert 1.0 ist, haben wir auf die Angabe des jeweiligen Fehlers verzichtet. Die Überlegenheit der Gauß-Integration ist unübersehbar.  $\triangle$

Eine leichte Verallgemeinerung der Problemstellung macht aus der Gauß-Integration ein mächtiges Instrument zur Integration auch schwieriger Integranden, z.B. mit speziellen Singularitäten oder über unendliche Intervalle. Es wird eine Gewichtsfunktion in das Integral eingeführt. Das ergibt die Problemstellung:

*Gegeben:* Ein Intervall  $[a, b]$  und eine Gewichtsfunktion  $\omega$ .

*Bestimme:* Integrationsgewichte  $w_k$  und Stützstellen  $x_k$ ,  $k = 1, \dots, n$ , so, dass die Quadraturformel

$$\sum_{k=1}^n w_k f(x_k) \quad \text{für } \int_a^b f(x) \omega(x) dx \quad (7.83)$$

den Genauigkeitsgrad  $2n - 1$  hat.

Wenn die Legendre-Polynome durch ein dem Intervall  $[a, b]$  und der Gewichtsfunktion  $\omega(x)$  zugeordnetes orthogonales Polynomsystem ersetzt werden, dann gilt die Aussage von Satz 7.6, dass eine eindeutige Quadraturformel existiert, und dass die Integrationsstützstellen die Nullstellen des  $n$ -ten orthogonalen Polynoms sind. Die Gewichte  $w_k$  sind Lösungen eines linearen Gleichungssystems, etwa wie in der Konstruktion von Satz 7.7. Die vielen möglichen Integrationsregeln erhalten einen Doppelnamen wie ‘‘Gauß-Laguerre-Integration’’, dessen zweiter Teil auf den Namen des entsprechenden orthogonalen Polynomsystems hinweist. Die wichtigsten Systeme sind in Tab. 7.7 enthalten,

Tab. 7.7 Orthogonale Polynomsysteme.

Polynomsystem	Gewichtsfunktion $\omega(x)$	Norm-Intervall
Legendre: $P_n$	1	$[-1, 1]$
Tschebyscheff 1. Art : $T_n$	$(1 - x^2)^{-1/2}$	$[-1, 1]$
Tschebyscheff 2. Art : $U_n$	$(1 - x^2)^{1/2}$	$[-1, 1]$
Laguerre: $L_n$	$e^{-x}$	$[0, \infty]$
Hermite: $H_n$	$e^{-x^2}$	$[-\infty, \infty]$

**Beispiel 7.13.** Wir greifen noch einmal Beispiel 7.5:  $I = \int_0^1 x^{3/2} dx = 0.4$  auf, bei dem die zweite Ableitung des Integranden im Nullpunkt singulär wird. Auch die Qualität der Gauß-Legendre-Quadratur lässt bei Integranden mit Singularitäten nach. Sie liefert aber immer noch bessere Werte als die bisher kennen gelernten Verfahren. Die Genauigkeit könnte durch eine spezielle Regel mit einer Gewichtsfunktion  $\omega(x)$  verbessert werden, die die Singularität enthält, sodass  $f(x)$  eine glatte Funktion ohne Singularität ist. Darauf wollen wir verzichten. Wir vergleichen wieder Trapez-, Simpson- und Gauß-Legendre-Quadratur für drei und fünf Funktionsauswertungen.

Tab. 7.8 Vergleich der Integrationsregeln bei singulärer zweiter Ableitung.

Anz.Fktsausw.	Trapez	Simpson	Gauß-Legendre
3	0.42677670	0.40236893	0.39981241
5	0.40701811	0.40043191	0.39998245

Der Fehler der Gauß-Legendre-Quadratur ist um den Faktor 12 bzw. 24 kleiner als der der Simpson-Regel, deren Genauigkeit durch die Romberg-Quadratur nicht wesentlich gesteigert werden kann, wie Beispiel 7.5 gezeigt hat.  $\triangle$

#### 7.4.1 Eingebettete Gauß-Regeln

Ein Nachteil der Gauß-Integration sind die unterschiedlichen Stützstellen für jedes  $n$ . Für gewisse feste Folgen von Werten von  $n$  gelingt es aber, unter geringem Genauigkeitsverlust eingebettete Datensätze zu bekommen. Solche Folgen von Integrationsregeln werden auch *optimal* genannt.

Hier ist zunächst die Gauß-Kronrod-Quadratur zu erwähnen: Ausgehend von gewissen  $n$ -Punkte-Gauß-Formeln ist es Kronrod gelungen, optimale  $2n + 1$ -Punkte-Formeln durch Hinzufügen von  $n + 1$  Punkten zu konstruieren:

$$\tilde{I}_1 := \sum_{i=1}^n w_i f(x_i) \longrightarrow \tilde{I}_2 := \tilde{I}_1 + \sum_{j=1}^{n+1} v_j f(y_j). \quad (7.84)$$

Die kombinierte Regel integriert Polynome bis zum Grad  $3n + 1$  exakt statt  $4n + 1$  für eine unabhängige  $2n + 1$ -Punkte-Formel. Einzelheiten zu diesen Algorithmen und die Stützwerte und Koeffizienten für die Regelpaare mit  $n = 7, 10, 15, 20, 25$  und  $30$  kann man in [Pie 83] finden. Die Originalarbeit von Kronrod [Kro 66] geht auf theoretische Untersuchungen von Szegö zurück.

Patterson, [Pat 69], hat eine Folge von eingebetteten Gauß-Legendre-Regeln mit  $(1, 3, 7, 15, 31, 63, 127, 255, 511)$  Punkten angegeben. Sie integrieren Polynome bis zum Grad  $(1, 5, 11, 23, 47, 95, 191, 383, 767)$  exakt.

In Beispiel 7.15 werden beide Methoden adaptiv angewendet.

## 7.5 Adaptive Integration

Softwaresysteme müssen Quadraturformeln enthalten, die sich mit Hilfe einer Genauigkeitskontrolle selbst steuern. Dabei muss die starre Aufteilung des Integrationsintervalls  $[a, b]$  in äquidistante Teilintervalle wie bei der Romberg-Quadratur ebenso aufgegeben werden wie die Anwendung immer genauerer Regeln auf das ganze Intervall wie bei der Gauß-Quadratur. Vielmehr wird das Intervall  $[a, b]$  auf Grund von bestimmten Kriterien *adaptiv* (angepasst) in Teilintervalle aufgeteilt. Ein Integral über ein kurzes Intervall kann ohnehin genauer und schneller berechnet werden als ein entsprechendes Integral über ein langes Intervall. Deshalb ist es immer vorteilhaft, vor der Anwendung einer Quadraturformel das Integrationsintervall  $[a, b]$  geeignet zu unterteilen und dann die Teilintegrale aufzusummieren. Adaptive Verfahren unterteilen  $[a, b]$  fortgesetzt solange, bis in jedem Teilintervall mit der zu Grunde gelegten Quadraturformel die geforderte Genauigkeit erreicht wird. Dabei wird die Unterteilung automatisch dort feiner, wo  $f(x)$  stark variiert, und größer in Intervallen geringerer Variation, wie in den Beispielen 7.14 und 7.15 unten gut zu sehen ist. Die Entscheidung, ob ein Teilintervall weiter unterteilt werden soll, erfolgt auf Grund des Vergleichs von zwei verschiedenen Näherungswerten  $\tilde{I}_1$  und  $\tilde{I}_2$  für dasselbe Teilintegral.

Um das Prinzip darzulegen, verwenden wir als Näherung  $\tilde{I}_1$  den Trapezwert für das Teilintervall  $[a_j, b_j]$

$$\tilde{I}_1 = \frac{1}{2} h_j [f(a_j) + f(b_j)], \quad h_j := b_j - a_j.$$

Für  $\tilde{I}_2$  wird der Simpson-Wert

$$\tilde{I}_2 = \frac{1}{3} [\tilde{I}_1 + 2h_j f(m_j)], \quad m_j := \frac{1}{2}(a_j + b_j),$$

unter Verwendung der schon berechneten Trapezformel  $\tilde{I}_1$  berechnet.

Der Abbruch für die lokale Intervallhalbierung kann über zwei Genauigkeitsschranken  $\varepsilon$  für die absolute und  $\delta$  für die relative Genauigkeit gesteuert werden. Sei  $I_S > 0$  eine grobe Schätzung für den Absolutbetrag des zu berechnenden Integralwertes. Dann wird die lokale Intervallhalbierung abgebrochen, falls

$$|\tilde{I}_1 - \tilde{I}_2| \leq \max(\varepsilon, \delta I_S), \quad (7.85)$$

d.h. wenn die relative oder die absolute Genauigkeitsforderung schätzungsweise erfüllt ist. Meistens wird (7.85) durch eine Abfrage ersetzt, die die aktuelle Intervallbreite in der Weise berücksichtigt, dass die Abfragesumme die geforderte Toleranz ergibt, also

$$|\tilde{I}_1 - \tilde{I}_2| \leq \frac{b_j - a_j}{b - a} \max(\varepsilon, \delta I_S), \quad (7.86)$$

Gander schlägt in [Gan 92] eine Genauigkeitssteuerung vor, die die Maschinengenauigkeit  $\tau$  mit berücksichtigt. Soll ein Näherungswert für  $I \neq 0$  nur mit der *relativen Genauigkeit*  $\delta > 0$  bestimmt werden, dann erreicht man dies dadurch, dass für einen groben Schätzwert  $I_S$  ein Steuerungswert  $I_\delta := \delta I_S / \tau$  gesetzt wird. Die lokale Intervallhalbierung wird abgebrochen, wenn in Maschinenarithmetik

$$\tilde{I}_1 + I_\delta = \tilde{I}_2 + I_\delta \quad (7.87)$$

gilt.

Die eleganteste und kompakteste algorithmische Beschreibung der adaptiven Quadratur ergibt sich vermittels rekursiver Definition eines Unterprogramms [Gan 92]. Wenn wir von dieser Möglichkeit absehen, dann erfordert die algorithmische Realisierung die Abspeicherung der Teilpunkte  $a_j$  und der zugehörigen Funktionswerte  $f_j = f(a_j)$  als Vektoren, um sie wiederverwenden zu können. Zudem wird die Information über die Teilintervalle, über welche die Integrale noch zu berechnen sind, benötigt. Dazu wird im Algorithmus Tab. 7.9 ein Indexvektor  $\mathbf{u}$  verwendet, welcher die Indizes  $p$  der laufend generierten Teilpunkte  $a_p$  enthält, mit denen die Integrationsintervalle erklärt werden können. Die Zahl der Komponenten des Vektors  $\mathbf{u}$  variiert im Verlauf des Algorithmus Tab. 7.9, während jene der Vektoren  $\mathbf{a}$  und  $\mathbf{f}$  monoton zunimmt.

Tab. 7.9 Algorithmus zur adaptiven Trapez-Simpson-Quadratur.

Start: $a_0 = a; a_1 = b; f_0 = f(a); f_1 = f(b); I = 0$ $j = 0; k = 1; p = 1; l = 1; u_1 = 1$ HALB: $h = a_k - a_j; m = (a_j + a_k)/2; fm = f(m)$ $I1 = h \times (f_j + f_k)/2; I2 = (I1 + 2 \times h \times fm)/3$ falls $I_\delta + I1 \neq I_\delta + I2 :$ $p = p + 1; a_p = m; f_p = fm; k = p$ $l = l + 1; u_l = p; \text{ gehe nach HALB}$ sonst $I = I + I2; j = u_l; l = l - 1; k = u_l$ falls $l > 0 :$ gehe nach HALB	(7.88)
--	--------

**Beispiel 7.14.** Das singuläre Integral von Beispiel 7.5 wird mit dem Algorithmus Tab. 7.9 behandelt. Mit dem Steuerungswert  $I_\delta = 0.5 \delta / \tau$  und der Maschinengenauigkeit  $\tau \doteq 2.22 \cdot 10^{-16}$  erhält man die Resultate der Tab. 7.10.  $N$  ist die Anzahl der Auswertungen des Integranden. Für  $\delta = 10^{-5}$  sind auch die Teilintervallgrenzen in Tab. 7.10 zu sehen. Der Einfluss der Singularität links ist deutlich zu erkennen.  $\triangle$

Tab. 7.10 Adaptive Quadratur mit Intervalleinteilung für  $\delta = 10^{-5}$ .

$\delta =$	$10^{-3}$	$10^{-4}$	$10^{-5}$	$10^{-6}$	$10^{-7}$
$\tilde{I} =$	0.4000002562	0.4000000357	0.4000000167	0.4000000025	0.4000000001
$N =$	25	69	125	263	651



Adaptive Methoden zeichnen sich dadurch aus, dass sie für fast beliebige Integranden, die beispielsweise nur stückweise stetig oder sogar unbeschränkt sind, annehmbare Integralwerte liefern. Die adaptive Quadratur ist natürlich effizienter, falls Quadraturformeln mit höheren Fehlerordnungen kombiniert werden wie etwa die eingebetteten Gauß-Regeln aus Abschnitt 7.4.1, die in Beispiel 7.15 verwendet werden. Dabei wird in Kauf genommen, dass die Anzahl der Funktionsauswertungen meist recht hoch ist. Wir wollen an einem weiteren Beispiel die adaptive Anwendung des Kronrod-Paares (10,21) demonstrieren und mit der globalen Steuerung des Patterson-Verfahrens ohne adaptive Unterteilung des Intervalls vergleichen, siehe Abschnitt 7.4.1.

**Beispiel 7.15.** Wir definieren eine Funktionen mit stark unterschiedlicher Variation und einer Unstetigkeitsstelle:

$$f(x) = \begin{cases} \sin(30x) \exp(3x) & \text{falls } x < 13\pi/60 \\ 5 \exp(-(x - 13\pi/60)) & \text{falls } x \geq 13\pi/60 \end{cases}$$

Mit dieser Funktion wollen wir das Integral

$$I = \int_0^3 f(x) dx \doteq 4.56673516941143$$

bestimmen, siehe Abb. 7.2.

Zur Genauigkeitssteuerung wird bei der Kronrod-Quadratur die Differenz zwischen den beiden Regeln in jedem Teilintervall genommen. Bei der Patterson-Quadratur ist es auch die Differenz zweier aufeinanderfolgender Regeln, aber über das ganze Intervall, die die automatische Steuerung kontrolliert. Beide Regeln sind in den NAG-Bibliotheken [NAGa, NAGb] realisiert und können mit relativen und absoluten Genauigkeitsschranken gesteuert werden.

Als relative und absolute Genauigkeitsschranke haben wir  $\delta = 1.0 \cdot 10^{-9}$  gewählt. Die Kronrod-Regel-Kombination ergibt folgendes Ergebnis:

Gauß-Kronrod	Fehler	Fehlerschätzung
4.5667351695951	$1.84 \cdot 10^{-10}$	$1.32 \cdot 10^{-9}$

Funktion und adaptive Intervallunterteilung für  $I$  werden in Abb. 7.2 wiedergegeben.

Von den 29 Punkten, die die adaptive Intervallunterteilung für  $I$  erzeugt, liegen 21 im Intervall  $[0.67, 0.7]$ , einer kleinen Umgebung der Unstetigkeitsstelle  $\bar{x} \doteq 0.6806784$ . Die Ergebnisse sind sehr zufriedenstellend. Die Fehlerschätzung ist nur sehr wenig größer als der wirkliche Fehler. Berechnet man zum Vergleich das Integral  $I$  mit der globalen Patterson-Quadratur, so reichen 511 Punkte nicht einmal für eine Genauigkeit von  $1.0 \cdot 10^{-3}$  aus. Für eine relative und absolute Genauigkeitsforderung von  $1.0 \cdot 10^{-2}$  bekommt man den Integralwert  $I_P = 4.556027$  zusammen mit einer Fehlerschätzung von 0.02421 für den absoluten Fehler, die den wahren Fehler etwa um den Faktor 2 unterschätzt. Für die stark unterschiedliche Variation des Integranden und die Sprungstelle im Integrationsbereich ist die globale Steuerung also nicht geeignet.

Da ist es naheliegend, dass der Anwender selbst eine Intervallaufteilung vornimmt: Wir wollen zwei Patterson-Integrationen auf die Intervalle  $[0, 13\pi/60]$  und  $[13\pi/60, 3]$  anwenden und addieren, also ohne Unstetigkeitstelle rechnen. Die Genauigkeit dieser Summe ist bei weniger Funktionsauswertungen als Kronrod noch wesentlich genauer:

Gauß-Patterson	Fehler	Fehlerschätzung
4.5667351694115	$1 \cdot 10^{-13}$	$6 \cdot 10^{-12}$

△

Abschließend verweisen wir auf die Tatsache, dass es kein absolut sicher arbeitendes Black-box-Verfahren zur automatischen Quadratur geben kann. Die Fehlerkontrolle kann versagen, wenn auch nur in sehr wenigen, aber nicht unbedingt exotischen Fällen, wie Lyness in [Lyn 83] zeigt. Numerische Integration ist oft Teil einer größeren Aufgabenstellung wie z.B.

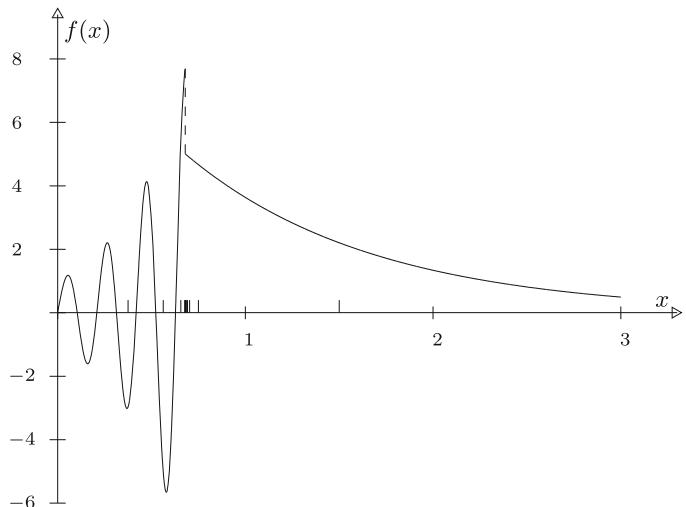


Abb. 7.2 Integrand und Intervalleinteilung für  $I$ .

einer Optimierungsaufgabe. Dann müssen Quadraturen ähnlicher Integranden sehr oft, z.B. mehrere tausend Male, ausgeführt werden. In solchen Fällen ist sorgfältig zu prüfen, ob nicht eine Quadratur mit fester Stützstellenzahl einer adaptiven Quadratur vorzuziehen ist. Das gilt besonders für mehrdimensionale Quadraturen.

## 7.6 Mehrdimensionale Integration

### 7.6.1 Produktintegration

Für die mehrdimensionale Integration bietet sich zunächst die Verwendung von Produktregeln an, die aus eindimensionalen Regeln gewonnen werden können. Für ein zweidimensionales Integral der Form

$$I := \int_a^b \int_c^d f(x, y) dx dy \quad (7.89)$$

kommen wir mit den  $n$  bzw.  $m$  eindimensionalen Stützwerten und Koeffizienten  $(x_i, w_i)$  für das Intervall  $[a, b]$  und  $(y_j, v_j)$  für das Intervall  $[c, d]$  zu der Regel

$$I \approx \sum_{i=1}^n w_i \int_c^d f(x_i, y) dy \approx \sum_{i=1}^n \sum_{j=1}^m w_i v_j f(x_i, y_j). \quad (7.90)$$

Diese Anwendung der Produktregel lässt sich leicht auf den Fall variabler Grenzen des inneren Integrals verallgemeinern:

$$I = \int_a^b \int_{\phi_1(y)}^{\phi_2(y)} f(x, y) dx dy. \quad (7.91)$$

**Beispiel 7.16.** Wir wollen das Volumen der zusammentreffenden Flachwasserwellen aus Beispiel 3.9 bestimmen. Dazu müssen wir die zweidimensionale Funktion  $F(x, y)$  (3.88) integrieren:

$$\int_{-8}^7 \int_{-8}^7 F(x, y) dx dy.$$

Wir tun das mit Hilfe der NAG-Routine D01DAF, die die in Abschnitt 7.4.1 kurz beschriebene Patterson-Methode auf zwei Dimensionen im Sinne der Produktregel verallgemeinert. In der folgenden Tabelle sind die steuernden Toleranzen  $\delta$ , die Integralwerte und die Anzahl der notwendigen Funktionsauswertungen angegeben.

Tab. 7.11 Integration der Flachwasserwelle mit der Patterson-Methode.

$\delta$	$I$	$n$
0.05	81.6632724161	449
0.005	81.6762461810	593
0.0005	81.6762980973	1857
0.00005	81.6763008759	1953

Der letzte Wert ist auf 10 wesentliche Stellen genau. Allerdings erscheint eine geringere Genauigkeitsforderung für diese Aufgabenstellung angemessen.  $\triangle$

### 7.6.2 Integration über Standardgebiete

Unregelmäßige Gebiete, wie sie bei der Lösung von partiellen Differentialgleichungen vorkommen, werden oft mit einer Menge von transformierten Standardgebieten überdeckt. Das sind z.B. bei der Methode der finiten Elemente oft Dreiecke, siehe Abschnitt 10.3. Hier ist also die Auswertung von Integralen auf vielen kleinen Gebieten der gleichen einfachen Form – Dreiecke, Vierecke, Tetraeder etc. – notwendig. Die zu integrierende Funktion ist glatt, und eine mehrdimensionale Gauß-Regel wird schon mit wenigen Punkten eine Genauigkeit erreichen, die die der übrigen bei der Problemlösung angewendeten numerischen Methoden übersteigt. Wir wollen eine Gauß-Regel mit sieben Stützstellen für das Standarddreieck  $T_0 = \{(0, 0), (1, 0), (0, 1)\}$  angeben, welche die exakten Integralwerte für Polynome bis zum Grad fünf liefert [Sch 91b].

$$\int_{T_0} f(x, y) dx dy \approx \sum_{i=1}^7 w_i f(\xi_i, \eta_i) \quad (7.92)$$

Die Integrationsstützpunkte  $R_i = (\xi_i, \eta_i)$ ,  $i = 1, \dots, 7$ , liegen auf den drei Mittellinien, der erste im Schwerpunkt des Dreiecks  $T_0$ , siehe Tab. 7.12 und Abb. 7.3.

Tab. 7.12 Integrationsstützpunkte  $R_i$  im Einheitsdreieck  $T_0$ .

$i$	$\xi_i$	$\eta_i$	$w_i$
1	$1/3 \doteq 0.333\ 333\ 333$	0.333 333 333	$9/80 = 0.1125$
2	$(6 + \sqrt{15})/21 \doteq 0.470\ 142\ 064$	0.470 142 064	$(155 + \sqrt{15})/2400$
3	$(9 - 2\sqrt{15})/21 \doteq 0.059\ 715\ 872$	0.470 142 064	$\doteq$
4	$(6 + \sqrt{15})/21 \doteq 0.470\ 142\ 064$	0.059 715 872	0.066 197 0764
5	$(6 - \sqrt{15})/21 \doteq 0.101\ 286\ 507$	0.101 286 507	$(155 - \sqrt{15})/2400$
6	$(9 + 2\sqrt{15})/21 \doteq 0.797\ 426\ 985$	0.101 286 507	$\doteq$
7	$(6 - \sqrt{15})/21 \doteq 0.101\ 286\ 507$	0.797 426 985	0.062 969 5903

**Beispiel 7.17.** Wie in Beispiel 7.16 soll die Flachwasserwelle aus Beispiel 3.9 integriert werden, indem das quadratische Gebiet  $(-8, 7) \times (-8, 7)$  gleichmäßig in rechtwinklige Dreiecke zerlegt, jedes Dreieck auf das Einheitsdreieck  $T_0$  transformiert und dort die Regel (7.92) angewendet wird.

Tab. 7.13 Gauß-Integration der Flachwasserwelle über eine Dreieckszerlegung.

# Dreiecke	$I$	$n$
32	81.553221963	224
128	81.679291706	896
512	81.676329426	3584
2048	81.676301173	14336
8192	81.676300880	57344

Die Ergebnisse in Tab. 7.13 zeigen bei diesem Beispiel die höhere Genauigkeit der Patterson-Quadratur (Tab. 7.11) bei gleichem Aufwand. Ist das Gebiet weder rechteckig noch in einer Form,

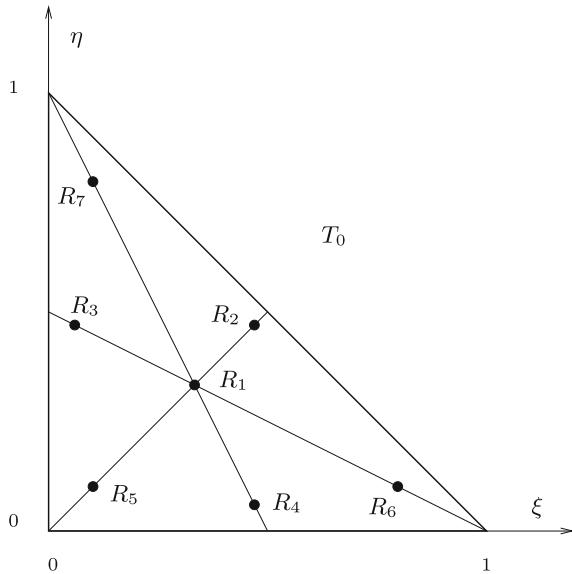


Abb. 7.3 Integrationsstützpunkte  $R_i$  im Einheitsdreieck  $T_0$  für Regel (7.92).

die die Darstellung (7.91) für das Integral erlaubt, ist eine Dreieckszerlegung und Anwendung von Regel (7.92) allerdings eine der wenigen guten Möglichkeiten.  $\triangle$

Neben den beiden beschriebenen Verfahrensklassen spielt noch die stochastische Monte-Carlo-Methode eine gewisse Rolle. Sie liefert jedoch nur eine sehr begrenzte Genauigkeit und ist abhängig von einem guten Zufallsgenerator. Allerdings bleibt sie bei hohen Raumdimensionen oft die einzige Möglichkeit.

## 7.7 Software

Ein frei erhältliches FORTRAN90-Paket für die eindimensionale numerische Integration ist QUADPACK<sup>1</sup>, das auf [Pie 83] basiert. Seine Routinen haben auch in größere Pakete Eingang gefunden wie z.B. in das frei erhältliche Paket SLATEC<sup>2</sup> mit über 1400 Routinen zu numerischen und statistischen Methoden.

Alle wichtigen Verfahren, also insbesondere die Gauß-Methoden für endliche oder halb-unendliche Intervalle oder über ganz  $\mathbb{R}$  sowie mit verschiedenen Gewichtsfunktionen zur Berücksichtigung des Abklingverhaltens oder von Singularitäten, sind in den großen numerischen Bibliotheken NAG und IMSL zu finden.

Bibliotheks routinen zur adaptiven Quadratur nach Gauß-Kronrod mit den Regelpaaren

<sup>1</sup>[http://orion.math.iastate.edu/burkardt/f\\_src/quadpack/quadpack.html](http://orion.math.iastate.edu/burkardt/f_src/quadpack/quadpack.html)

<sup>2</sup><http://www.netlib.org/slatec/>

für die 7-15-Punkte- und die 10-21-Punkte-Kombination findet man im Kapitel D01 der NAG-Bibliotheken [NAGa, NAGb]. Auch die ein- und mehrdimensionalen Patterson-Regeln zu der im Abschnitt 7.4.1 erwähnten Punktanzahl-Folge findet man in den Routinen von Kapitel D01 der NAG-Bibliotheken.

MATLAB kennt zwei Befehle zur eindimensionalen Quadratur, die die Simpson-Regel bzw. eine adaptive Gauß-Integration benutzen. Die zweidimensionale Integration ist auf ein Rechteck mit Benutzung einer Simpson-Produktregel beschränkt.

Unsere Problemlöseumgebung PAN (<http://www.upb.de/SchwarzKoeckler/>) verfügt über zwei Programme zur eindimensionalen numerischen Gauss-Kronrod- und Patterson-Quadratur und über ein Programm zur zweidimensionalen numerischen Patterson-Quadratur.

## 7.8 Aufgaben

**Aufgabe 7.1.** Seien  $a \leq b < c$  die Halbachsen eines Ellipsoides. Seine Oberfläche  $A$  ist gegeben durch das Integral einer periodischen Funktion

$$\begin{aligned} A &= ab \int_0^{2\pi} \left[ 1 + \left( \frac{1}{w} + w \right) \arctan(w) \right] d\varphi, \\ w &= \sqrt{\frac{c^2 - a^2}{a^2} \cos^2 \varphi + \frac{c^2 - b^2}{b^2} \sin^2 \varphi}. \end{aligned}$$

Wegen der Symmetrie genügt die Integration über eine Viertelperiode. Falls  $a \geq b > c$ , gilt in reeller Form

$$A = ab \int_0^{2\pi} \left[ 1 + \left( \frac{1}{v} - v \right) \operatorname{Artanh}(v) \right] d\varphi, \quad v = iw.$$

Man berechne  $A$  mit Trapezregeln. Man beginne mit einer Schrittweite  $h = \pi/4$  und halbiere  $h$  solange, bis zwei aufeinanderfolgende Trapezregel-Ergebnisse sich um weniger als eine vorgegebene Schranke  $\varepsilon$  unterscheiden.

**Aufgabe 7.2.** Man transformiere die folgenden Integrale so, dass sie mittels Algorithmus (7.22) für Integrale über  $\mathbb{R}$  effizient berechnet werden können.

a)  $\int_0^1 x^{0.21} \sqrt{\ln\left(\frac{1}{x}\right)} dx = \sqrt{\pi}/2.662$

b)  $\int_0^1 \frac{dx}{(3-2x)x^{3/4}(1-x)^{1/4}} = \pi\sqrt{2} \cdot 3^{-3/4}$

c)  $\int_0^\infty x^{-0.7} e^{-0.4x} \cos(2x) dx = \Gamma(0.3) \operatorname{Re}[(0.4+2i)^{-0.3}]$

d)  $\int_0^\infty \frac{dx}{x^{1-\alpha} + x^{1+\beta}} = \frac{\omega}{\sin(\alpha\omega)}, \quad \omega = \frac{\pi}{\alpha+\beta}, a > 0, \beta > 0$

$$\text{e)} \quad \int_{-\infty}^{\infty} \frac{dx}{(1+x^2)^{5/4}} = \left(\frac{\pi}{2}\right)^{3/2} \Gamma(1.25)^{-2}$$

$$\text{f)} \quad \int_{-\infty}^{\infty} \frac{dx}{x^2 + e^{4x}} \doteq 3.1603228697485$$

**Aufgabe 7.3.** Man forme das Integral von Beispiel 7.9 mit weiteren sinh-Transformationen  $t = t_1 = \sinh(t_2), \dots$  um und vergleiche die Effizienz von Algorithmus (7.22) für die entsprechenden Integrale.

**Aufgabe 7.4.** Der adaptive Quadraturalgorithmus (7.88) funktioniert sogar für viele unstetige Integranden, beispielsweise für

$$I = \int_1^x \frac{[\xi]}{\xi} d\xi = [x] \ln(x) - \ln([x]!), \quad x > 1,$$

wobei  $[x]$  die größte ganze Zahl kleiner oder gleich  $x$  bedeutet. Für das durch die Substitution  $\xi = e^t$  entstehende Integral

$$I = \int_0^{\ln x} [e^t] dt$$

versagt jedoch der Algorithmus (7.88), indem er im Allgemeinen zu früh abbricht. Was ist der Grund dafür?

**Aufgabe 7.5.** Man berechne die folgenden Integrale mit der Trapez- und der Simpson-Regel. Man verbessere die Ergebnisse durch Anwendung des Romberg-Verfahrens und der Gauß-Legendre-Quadratur mit  $n = 4, 5, 6, 7, 8$  Knoten. Man vergleiche die Genauigkeiten der Näherungswerte für die gleiche Anzahl von Funktionsauswertungen. Schließlich wende man eine adaptive Quadraturmethode an, um die gleiche Genauigkeit zu erzielen und zähle dabei die Funktionsauswertungen.

$$\text{a)} \quad \int_0^3 \frac{x}{1+x^2} dx = \frac{1}{2} \ln(10); \quad \text{b)} \quad \int_0^{0.95} \frac{dx}{1-x} = \ln(20);$$

$$\text{c)} \quad \frac{1}{\pi} \int_0^{\pi} \cos(x \sin \varphi) d\varphi = J_0(x) \quad (\text{Bessel-Funktion})$$

$$J_0(1) \doteq 0.7651976866, \quad J_0(3) \doteq -0.2600519549, \quad J_0(5) \doteq -0.1775967713;$$

$$\text{d)} \quad \int_0^{\pi/2} \frac{d\varphi}{\sqrt{1-m \sin^2 \varphi}} = K(m) \quad (\text{vollständiges elliptisches Integral erster Art})$$

$$K(0.5) \doteq 1.8540746773, \quad K(0.8) \doteq 2.2572053268, \quad K(0.96) \doteq 3.0161124925.$$

**Aufgabe 7.6.** Man berechne die Knoten  $x_k$  und Gewichte  $w_k$  der Gaußschen Quadraturformeln für  $n = 5, 6, 8, 10, 12, 16, 20$ , indem man den QR-Algorithmus und die inverse Vektoriteration auf die tridiagonalen Matrizen  $\mathbf{J}_n$  (7.77) anwendet.

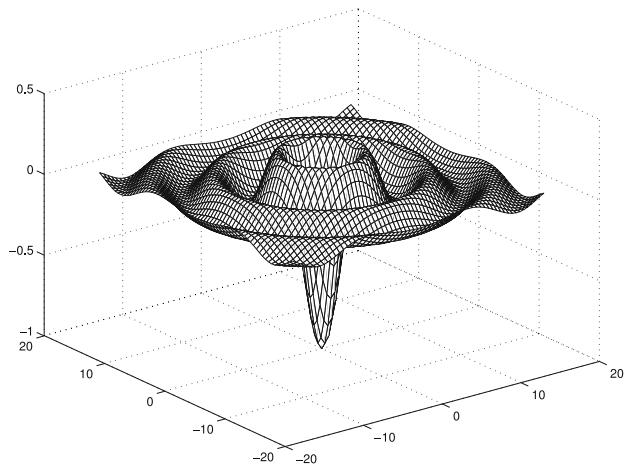


Abb. 7.4 Wir werfen einen Stein ins Wasser.

**Aufgabe 7.7.** Man verallgemeinere den Algorithmus (7.88) zur adaptiven Quadratur auf eine Produktintegration über ein Rechteck im  $\mathbb{R}^2$ .

Mit dieser Produktintegration berechne man eine Näherung für das Integral

$$\iint_G f(x, y) dx dy \quad \text{mit } f(x, y) = \frac{-\sin r}{r},$$

wo  $r := \sqrt{x^2 + y^2} + \varepsilon$  der Abstand vom Ursprung ist, der zur Vermeidung von Division durch null um  $\varepsilon$  verschoben wird. Man wähle z.B.  $\varepsilon = 10^{-12}$ .

Das Integrationsgebiet sei das Quadrat  $G = [-16, 16] \times [-16, 16]$ . Die zu integrierende Funktion sieht aus wie von einem Stein verursachte Wasserwellen, siehe Abb. 7.4.

**Aufgabe 7.8.** Die Funktion  $f(x, y)$  aus Aufgabe 7.7 soll über das Gebiet der Trommel aus Aufgabe 5.11 integriert werden. Dazu verschiebe man den Punkt  $(4, 4)$  in den Ursprung und zerlege das Gebiet gleichmäßig in Dreiecke. Man programmiere die Transformation der Dreiecke auf das Standarddreieck  $T_0 = \{(0, 0), (1, 0), (0, 1)\}$  und benutze dann die Integration aus Abschnitt 7.6.2.

Die Integration wird adaptiv, wenn die Dreiecke systematisch verkleinert und die Ergebnisse für zwei Verfeinerungsstufen verglichen werden. Dieser adaptive Integrationsvorgang wird abgebrochen, wenn die Differenz zwischen zwei Verfeinerungsstufen kleiner als eine vorgegebene Schranke wird.

## 8 Anfangswertprobleme bei gewöhnlichen Differenzialgleichungen

Gewöhnliche Differenzialgleichungen (DGL) sind Gleichungen für Funktionen von einer unabhängigen Variablen, in denen die unbekannten Funktionen und ihre Ableitungen bis zu einer bestimmten Ordnung vorkommen. Eine Lösung in geschlossener, analytischer Form lässt sich nur in relativ wenigen Fällen angeben. Deshalb sind numerische Verfahren gefragt, die eine hinreichend genaue Näherung der Lösungsfunktion liefern.

Für Anfangswertaufgaben bei gewöhnlichen Differenzialgleichungen gibt es eine reizvolle Vielfalt von Anwendungen. Dazu gehören chemische Reaktionen, Bevölkerungsentwicklungsmodelle und Wettbewerbsmodelle in den Wirtschaftswissenschaften, physiologische Indikatormodelle in der Medizin, Dopingtests und Schädlingsbekämpfung in der Pharmakologie, Epidemieverlauf, Artenverhalten und Symbiosemodelle in der Biologie. Zur Berechnung einer Lösung muss immer der Anfangszustand des jeweiligen Systems bekannt sein.

Wir beginnen mit einem einführenden Abschnitt, in dem wir neben der Problemstellung einige einfache Verfahren mit ihren Eigenschaften exemplarisch vorstellen. Danach beschäftigen wir uns mit der Theorie der Einschrittverfahren, stellen die wichtigen Runge-Kutta-Verfahren ausführlich vor und gehen auf die für die Praxis wichtige Schrittweitensteuerung ein. Ein weiterer Abschnitt beschäftigt sich in ähnlicher Weise mit den Mehrschrittverfahren. Schließlich widmen wir uns dem Begriff der Stabilität, dessen Wichtigkeit an so genannten steifen Differenzialgleichungen deutlich wird.

Wir befassen uns nicht mit Fragen der Existenz und Eindeutigkeit von Lösungen der Differenzialgleichungen, sondern setzen stillschweigend voraus, dass die betreffenden Voraussetzungen erfüllt sind [Ama 95, Col 90, Heu 09, Wal 00]. Weiterführende Darstellungen numerischer Verfahren für Anfangswertprobleme sind in [Aik 85, Alb 85, But 87, Deu 08a, Fat 88, Fox 88, Gea 71, Gri 72, Hai 93, Hai 96, Hen 62, Jai 84, Lam 91, Lap 71, Sew 05, Sha 94, Sha 84, Ste 73b, Str 95] zu finden.

Mit Randwertproblemen werden wir uns in Kapitel 9 befassen.

## 8.1 Einführung

### 8.1.1 Problemklasse und theoretische Grundlagen

Es soll das Anfangswertproblem (A)

$$(A) : \begin{aligned} y'(x) &= f(x, y(x)) && (\text{DGL}), \\ y(a) &= y_0 && (\text{AB}), \end{aligned} \quad (8.1)$$

für  $x \in [a, b]$  gelöst werden. Dazu sei die gegebene Funktion auf der rechten Seite stetig ( $f \in C([a, b] \times \mathbb{R}^n, \mathbb{R}^n)$ ) und die gesuchte Funktion  $y(x)$  stetig differenzierbar ( $y \in C^1([a, b], \mathbb{R}^n)$ ). Weiter ist der Vektor der Anfangsbedingungen  $y_0 \in \mathbb{R}^n$  gegeben.

$y' = f(x, y)$  ist also ein Differenzialgleichungssystem erster Ordnung für eine vektorwertige Funktion  $y$ , die von einer skalaren Variablen  $x$  abhängt:

$$\begin{aligned} y'_1(x) &= f_1(x, y_1(x), y_2(x), \dots, y_n(x)), \\ y'_2(x) &= f_2(x, y_1(x), y_2(x), \dots, y_n(x)), \\ &\vdots \\ y'_n(x) &= f_n(x, y_1(x), y_2(x), \dots, y_n(x)), \\ y(a) &= y_0 = (y_{10}, y_{20}, \dots, y_{n0})^T. \end{aligned} \quad (8.2)$$

Das Differenzialgleichungssystem heißt *explizit*, weil die höchste Ableitung (hier die erste) isoliert auf der linken Seite auftritt. Auf *implizite* Differenzialgleichungen, die nicht in explizite auflösbar sind, werden wir nicht eingehen.

Viele der angeführten Beispiele werden in [Heu 09] ausführlicher beschrieben, weitere sind in [Col 66] oder in [Sto 02] zu finden.

**Lemma 8.1.** *Jedes explizite System von  $n$  Differenzialgleichungen  $m$ -ter Ordnung lässt sich in ein äquivalentes System 1. Ordnung mit  $m \cdot n$  Gleichungen umformen.*

*Beweis.* Wir beweisen exemplarisch, dass sich eine einzelne Differenzialgleichung  $n$ -ter Ordnung in ein System mit  $n$  Differenzialgleichungen 1. Ordnung umformen lässt (der Rest ist einfache Kombinatorik).

Sei also eine Differenzialgleichung (mit  $z^{(k)} = k$ -te Ableitung von  $z$ )

$$z^{(n)} = g(x, z, z', \dots, z^{(n-1)}), \quad (8.3)$$

gegeben mit den Anfangswerten

$$z(a) = z_0, \quad z'(a) = z'_0, \quad \dots, \quad z^{(n-1)}(a) = z^{(n-1)}_0.$$

Definiere eine Vektorfunktion

$$\begin{aligned} y(x) &:= (y_1(x), y_2(x), \dots, y_n(x))^T \\ &:= \left( z(x), z'(x), \dots, z^{(n-1)}(x) \right)^T. \end{aligned} \quad (8.4)$$

Dann gilt offenbar

$$\begin{aligned}
 y'_1(x) &= y_2(x), \\
 y'_2(x) &= y_3(x), \\
 &\vdots \\
 y'_{n-1}(x) &= y_n(x), \\
 y'_n(x) &= g(x, z, z', \dots, z^{(n-1)}) \\
 &= g(x, y_1, y_2, \dots, y_n)
 \end{aligned} \tag{8.5}$$

und

$$y(a) = y_0 = \left( z_0, z'_0, \dots, z_0^{(n-1)} \right)^T. \tag{8.6}$$

(8.5) stellt mit (8.6) ein System der Form (8.1) dar, dessen Vektorlösung  $y(x)$  auch die skalare Lösung  $z(x) = y_1(x)$  von (8.3) liefert.  $\square$

Einige Vorbemerkungen für dieses Kapitel:

1. Die Beschränkung auf Differenzialgleichungen 1. Ordnung erlaubt eine einheitliche Darstellung. Dadurch ist es unerlässlich, numerische Verfahren für Systeme zu formulieren, weil dies erst die Behandlung von Differenzialgleichungen höherer Ordnung erlaubt. Auch die meisten Softwaresysteme beschränken sich formal auf die numerische Lösung von Systemen 1. Ordnung. Differenzialgleichungen höherer Ordnung müssen dann für die rechnerische Lösung in ein System 1. Ordnung umformuliert werden. Da das Verständnis der Aussagen für einzelne Gleichungen meistens leicht auf die Behandlung von Systemen übertragbar ist, verzichten wir in diesem Kapitel auf die fett gedruckte Darstellung von Vektorfunktionen.
2. Für die numerische Lösung setzen wir ohne Beschränkung der Allgemeinheit Existenz und Eindeutigkeit einer stetig differenzierbaren Lösung  $y(x)$  voraus. Interessante numerische Probleme liegen aber auch bei Wegfall der Eindeutigkeit vor, es entstehen dann z.B. Verzweigungsprobleme.
3.  $\|\cdot\|$  sei eine Norm im  $\mathbb{R}^n$ ,  $\|\cdot\|$  sei dann für Matrizen eine verträgliche, submultiplikative Matrixnorm mit  $\|\mathbf{I}\| = 1$ , in der Regel wird dies die zugeordnete Norm sein.

### **Definition 8.2. Globale Lipschitz-Bedingung:**

Es gebe eine Konstante  $L \in \mathbb{R}$ , die so genannte *Lipschitz-Konstante*, so dass

$$\|f(x, y) - f(x, \tilde{y})\| \leq L \|y - \tilde{y}\| \quad \forall (x, y), (x, \tilde{y}). \tag{8.7}$$

Ohne Beweis wollen wir die folgenden Grundtatsachen aus der Theorie der Anfangswertprobleme angeben.

**Lemma 8.3.** *Das Anfangswertproblem (8.1) ist äquivalent zu dem System von Integralgleichungen*

$$y(x) = y_0 + \int_a^x f(\xi, y(\xi)) d\xi. \tag{8.8}$$

**Satz 8.4.** Die globale Lipschitz-Bedingung (8.7) sei erfüllt.

Dann hängt die Lösung von (8.1) stetig von den Anfangswerten  $y_0$  ab; oder:

Sind  $y(x)$  und  $\tilde{y}(x)$  Lösungen von

$$\begin{array}{rcl} y' & = & f(x, y) \\ y(a) & = & y_0 \end{array} \quad \left| \quad \begin{array}{rcl} \tilde{y}' & = & f(x, \tilde{y}) \\ \tilde{y}(a) & = & \tilde{y}_0, \end{array} \right.$$

so gilt unter der globalen Lipschitz-Bedingung (8.7)

$$\|y(x) - \tilde{y}(x)\| \leq e^{L|x-a|} \|y_0 - \tilde{y}_0\|. \quad (8.9)$$

(8.9) ist auch eine numerische Abschätzung über die Lösungsabweichung  $y - \tilde{y}$  bei Datenfehlern  $y_0 - \tilde{y}_0$  in den Anfangswerten. Danach pflanzt sich jede Lösungsverfälschung mit wachsendem  $x$  desto stärker fort, je größer die Lipschitz-Konstante  $L$  ist und je größer der Abstand vom Anfangspunkt ist.

### 8.1.2 Möglichkeiten numerischer Lösung

Um einen ersten Eindruck von numerischen Lösungsmöglichkeiten zu bekommen, soll die Anfangswertaufgabe

$$\begin{array}{rcl} y'(x) & = & f(x, y(x)), \\ y(a) & = & y_0, \end{array}$$

diskretisiert werden. Wir suchen Näherungen  $u_j \approx y(x_j)$  an festen, zunächst äquidistanten Stellen  $x_j$  in einem Intervall  $[a, b]$ .

$$x_j = a + jh, \quad j = 0, 1, \dots, N, \quad h = \frac{b-a}{N}. \quad (8.10)$$

Dazu können z.B. die Ableitungen durch Differenzen ersetzt werden. Wir wollen diese Vorgehensweise an einfachen Verfahren exemplarisch betrachten. Dabei sei die Existenz der für die Herleitung benötigten höheren Ableitungen jeweils vorausgesetzt.

#### Das explizite Euler- oder Polygonzug-Verfahren

Ersetze  $y'(x_{j-1})$  durch die Vorwärtsdifferenz:

$$\begin{aligned} y'(x_{j-1}) &= \frac{y(x_j) - y(x_{j-1})}{h} + O(h) \quad \Rightarrow \\ y(x_j) &= y(x_{j-1}) + h f(x_{j-1}, y(x_{j-1})) + O(h^2). \end{aligned} \quad (8.11)$$

Das ergibt mit Berücksichtigung des bekannten Anfangswertes das Verfahren

$$\begin{aligned} u_0 &= y_0 \\ u_j &= u_{j-1} + h f(x_{j-1}, u_{j-1}), \quad j = 1, 2, \dots, N. \end{aligned} \quad (8.12)$$

Mit diesem einfachsten Verfahren kann man also von links nach rechts schrittweise Lösungsnäherungen mit geringem Aufwand berechnen. Wegen (8.11) ist der Fehler  $\|u_j - y(x_j)\|$  des expliziten Euler-Verfahrens von der Größenordnung  $O(h)$ , man spricht von der Fehlerordnung  $p = 1$ , siehe Abschnitt 8.2.1.

### Das implizite Euler-Verfahren

Hier wird statt (8.11) die Rückwärtsdifferenz bei  $x_j$  genommen:

$$y'(x_j) = \frac{y(x_j) - y(x_{j-1})}{h} + O(h) \approx \frac{u_j - u_{j-1}}{h}. \quad (8.13)$$

Das liefert das Verfahren:

$$\begin{aligned} u_0 &= y_0 \\ u_j &= u_{j-1} + h f(x_j, u_j), \quad j = 1, 2, \dots, N. \end{aligned} \quad (8.14)$$

Die Lösungen  $u_j$  können hier nur über die Lösung eines im Allgemeinen nichtlinearen Gleichungssystems gewonnen werden, da die von Schritt zu Schritt zu bestimmenden Werte  $u_j$  sowohl links als auch rechts im Argument der Funktion  $f$  vorkommt. Wegen (8.13) ist die Größenordnung des Fehlers beim impliziten Euler-Verfahren wie beim expliziten Euler-Verfahren  $O(h)$ .

Beide Euler-Verfahren gewinnt man auch über eine Taylor-Reihen-Approximation.

### Die Trapezmethode

Aus (8.8) folgt, dass

$$y(x_{j+1}) = y(x_j) + \int_{x_j}^{x_{j+1}} f(x, y(x)) dx. \quad (8.15)$$

Wird dieses Integral mit der Trapezregel (7.10) integriert, so ergibt sich die *Trapezmethode*

$$u_{j+1} = u_j + \frac{h}{2} \{f(x_j, u_j) + f(x_{j+1}, u_{j+1})\}. \quad (8.16)$$

Sie ist implizit, weil jeder Integrationsschritt die Lösung eines im Allgemeinen nichtlinearen Gleichungssystems nach der unbekannten Näherung  $u_{j+1}$  verlangt. Da nach (7.15)

$$y(x_{j+1}) = y(x_j) + \frac{h}{2} \{f(x_j, y(x_j)) + f(x_{j+1}, y(x_{j+1}))\} + O(h^3) \quad (8.17)$$

gilt, ist die Trapezmethode um eine  $h$ -Potenz genauer als die Euler-Verfahren, hat also die Fehlerordnung  $p = 2$ . Darüber hinaus zeigt sie ein gutes Stabilitätsverhalten, wie wir in Beispiel 8.2 sehen und in Abschnitt 8.4.2 verstehen werden.

Da bei den Verfahren (8.12), (8.14) und (8.16) die Lösung  $u_j$  nur aus Werten  $u_{j-1}$  berechnet wird, heißen solche Verfahren *Einschrittverfahren* (ESV).

### Ein Mehrschrittverfahren: die Mittelpunktregel

Wird in der Differenzialgleichung  $y'(x)$  an der Stelle  $x_{j+1}$  durch den zentralen Differenzenquotienten (3.29) ersetzt, so ergibt sich die Approximation

$$\frac{y(x_{j+2}) - y(x_j)}{2h} = f(x_{j+1}, y(x_{j+1})) + O(h^2). \quad (8.18)$$

Das führt auf das *Zweischritt-Verfahren*

$$u_{j+2} = u_j + 2h f(x_{j+1}, u_{j+1}), \quad j = 0, 1, \dots, N-1. \quad (8.19)$$

Um dieses Verfahren zu starten, reicht der Anfangswert  $y_0$  nicht aus, da jeder Schritt zwei zurückliegende Werte benötigt. Es wird deshalb ein Einschrittverfahren zur Berechnung von  $u_1$  verwendet, z.B. das explizite Euler-Verfahren. Das ergibt

$$\begin{aligned} u_0 &= y_0, \\ u_1 &= u_0 + h f(x_0, u_0), \\ u_2 &= u_0 + 2h f(x_1, u_1), \\ &\dots \end{aligned}$$

Dieses Verfahren hätte man ebenso aus der Integraldarstellung (8.8) bekommen können, etwa für  $x_2$

$$\begin{aligned} y(x_2) &= y_0 + \int_a^{a+2h} f(\xi, y(\xi)) d\xi \\ &= y_0 + 2h f(x_1, y(x_1)) + O(h^2). \end{aligned}$$

Hier wird die Mittelpunktregel zur numerischen Integration verwendet. Das gibt dem Verfahren seinen Namen. Es hat wegen (8.18) die Fehlerordnung  $p = 2$ .

**Beispiel 8.1.** Wir wollen die Anfangswertaufgabe

$$y' = -2x y^2, \quad y(0) = 1, \tag{8.20}$$

mit den beiden Euler-Verfahren und der Mittelpunktregel und mit der Schrittweite  $h = 0.1$  behandeln. Sie hat die exakte Lösung  $y(x) = 1/(x^2 + 1)$ . Mit dem expliziten Euler-Verfahren erhalten wir

$$\begin{aligned} u_0 &= 1, \\ u_1 &= u_0 + h f(x_0, u_0) = 1 + 0.1 \cdot 0 = 1, \\ u_2 &= u_1 + h f(x_1, u_1) = 1 + 0.1 \cdot (-0.2) = 0.98, \\ &\dots \end{aligned}$$

Die Lösungspunkte sind in Abb. 8.1 durch eine gestrichelte Linie verbunden. Es ist sogar graphisch zu sehen, dass die numerische Lösung der exakten Lösung mit einem Fehler  $O(h)$  ‘hinterherläuft’.

Mit dem impliziten Euler-Verfahren erhalten wir

$$\begin{aligned} u_0 &= 1 \\ u_1 &= u_0 + 0.1 f(x_1, u_1) \\ \implies u_1 &= 1 - 0.02 u_1^2 \\ \implies u_1^2 + 50u_1 &= 50 \\ \implies u_1 &= 0.98076211, \quad \text{die andere Lösung kommt nicht in Betracht.} \\ u_2 &= u_1 + 0.1 f(x_2, u_2) \\ &\dots \end{aligned}$$

Hier haben wir die Lösungspunkte in Abb. 8.1 durch eine Strichpunktlinie verbunden. Diese wesentlich aufwändiger zu berechnende Lösung ist auch nicht besser.

Jetzt soll die Mittelpunktregel auf die Anfangswertaufgabe (8.20) angewendet werden. Für den ersten Schritt zur Berechnung von  $u_1$  wird das explizite Euler-Verfahren verwendet. Damit bekommen

wir die Werte

$$\begin{aligned} u_0 &= 1 \\ u_1 &= 1 \\ u_2 &= u_0 + 0.2 f(x_1, u_1) = 1 + 0.2 \cdot (-0.2) = 0.96 \\ &\dots \end{aligned}$$

Die in Abb. 8.1 durch eine gepunktete Linie verbundenen Lösungspunkte sind entsprechend der höheren Fehlerordnung (zumindest im betrachteten Intervall) eine bessere Näherung.  $\triangle$

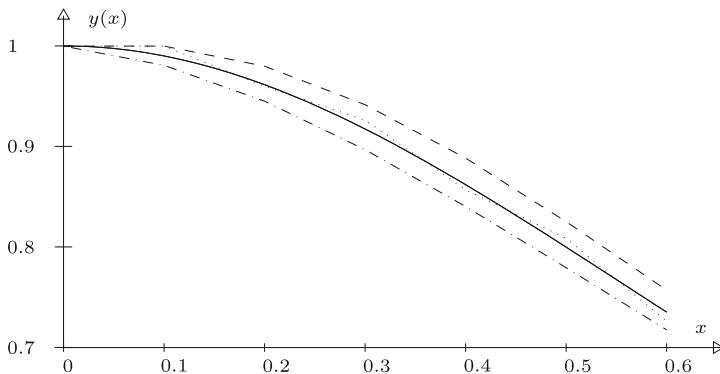


Abb. 8.1 Drei Beispielverfahren.

**Beispiel 8.2.** Um die Problematik der Stabilität von Verfahren gleicher Ordnung anzuschneiden, vergleichen wir die Trapezmethode (8.16) mit der Mittelpunktregel (8.19), indem wir beide auf das Anfangswertproblem

$$y' = -y, \quad y(0) = 1,$$

anwenden. Wir verwenden die Schrittweite  $h = 0.1$  und rechnen von  $x = 0$  bis  $x = 7$ . In Abb. 8.2 sind die Ergebnisse der beiden Regeln für  $x \in [2, 7]$  gezeichnet. Die Trapezregelwerte sind zeichnerisch kaum von der wahren Lösung  $y(x) = \exp(-x)$  zu unterscheiden, während die Mittelpunktregel mit wachsendem  $x$  immer stärker oszilliert.  $\triangle$

### Anwendung der Euler-Verfahren auf ein Modellproblem

Zur Untersuchung von Stabilitätseigenschaften ist das Modellproblem

$$y'(x) = \lambda y(x), \quad y(0) = 1, \quad \lambda \in \mathbb{R}, \tag{8.21}$$

sehr geeignet, bei dem in späteren Untersuchungen auch komplexe Werte für  $\lambda$  eine Rolle spielen werden. Es hat die exakte Lösung  $y(x) = e^{\lambda x}$ . Über den Parameter  $\lambda$  kann das Lösungsverhalten gesteuert werden, für  $\lambda < 0$  ergeben sich abklingende, für  $\lambda > 0$  anwachsende Lösungen. Der zweite Vorteil dieses einfachen Testproblems ist, dass für viele Verfahren die Näherungslösungen in ihrer Abhängigkeit von  $\lambda$  und  $h$  exakt dargestellt werden können und

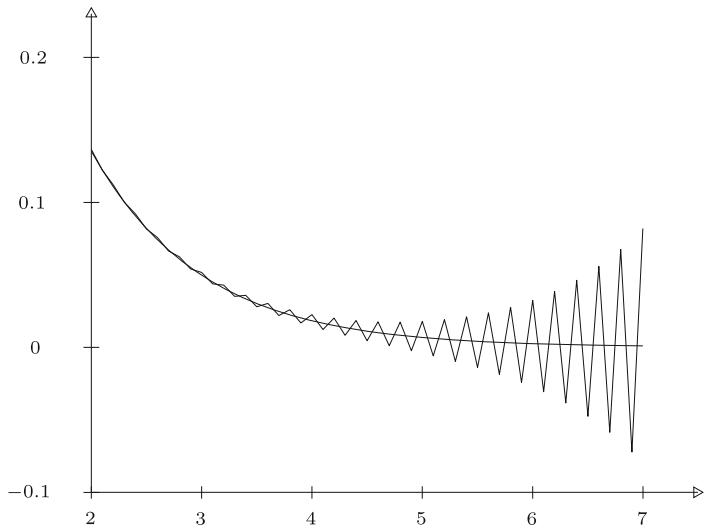


Abb. 8.2 Mittelpunktregel gegen Trapezmethode.

damit das Verhalten der Verfahren für unterschiedliche Parameterwerte untersucht werden kann. Das wollen wir für die beiden Euler-Verfahren ansehen.

Das *explizite Euler-Verfahren* (8.12) ergibt die Näherungswerte

$$\begin{aligned}
 u_0 &= 1 \\
 u_1 &= 1 + \lambda h = e^{\lambda h} + O(h^2) \\
 u_2 &= u_1 + h f(x_1, u_1) = 1 + \lambda h + h\lambda(1 + \lambda h) = 1 + 2\lambda h + (\lambda h)^2 \\
 &= (1 + \lambda h)^2 \\
 &\vdots \\
 u_j &= (1 + \lambda h)^j
 \end{aligned}$$

Hieraus können wir die Konvergenz des Verfahrens ablesen: Für ein festes  $\bar{x} \in \mathbb{R}$  lassen wir  $h \rightarrow 0$  und  $j \rightarrow \infty$  gehen, sodass immer  $\bar{x} = j \cdot h$  gilt, also  $h = \frac{\bar{x}}{j}$ . Dann gilt für den Näherungswert  $u_j$  von  $y(\bar{x})$ :

$$u_j = u_j(h) = (1 + \lambda \frac{\bar{x}}{j})^j \longrightarrow e^{\lambda \bar{x}} \quad \text{mit } j \rightarrow \infty$$

Dieses Konvergenzverhalten muss aber die Näherungslösung für festes  $h > 0$  nicht widerspiegeln. Ist z.B.  $\lambda < 0$  so stark negativ, dass  $|1 + \lambda h| > 1$ , dann wächst  $|u_j|$  oszillatorisch, während ja die exakte Lösung  $e^{\lambda h}$  abklingt, d.h. das Verfahren wird numerisch instabil, siehe Beispiel 8.3.

Hier wird die Diskrepanz deutlich zwischen der Tatsache der Konvergenz, die ja eine asymptotische Aussage darstellt, und einer numerischen Instabilität, die für gewisse Zahlenwerte  $h > 0$  auftritt.

Das *implizite Euler-Verfahren* (8.14) ergibt

$$\begin{aligned} u_0 &= 1 \\ u_1 &= 1 + h\lambda u_1 \quad \Rightarrow \quad u_1 = \frac{1}{1 - \lambda h} \\ u_2 &= u_1 + h\lambda u_2 \quad \Rightarrow \quad u_2 = \frac{1}{(1 - \lambda h)^2} \\ &\vdots \\ u_j &= \frac{1}{(1 - \lambda h)^j} \end{aligned}$$

Hier liegt Konvergenz in der gleichen Güte wie beim expliziten Euler-Verfahren vor. Darüber hinaus ist aber die numerische Lösung für endliches  $h > 0$  immer stabil, da im wichtigen Fall  $\lambda < 0$  mit einer abklingenden Lösungsfunktion auch die Werte  $u_j$  abklingen. Dieses von  $h$  unabhängige Stabilitätsverhalten werden wir in Abschnitt 8.4.2 als *absolute Stabilität* kennen lernen.

**Beispiel 8.3.** Wir demonstrieren das unterschiedliche Verhalten der beiden Euler-Verfahren an zwei Testrechnungen für das Modellproblem (8.21) mit unterschiedlichen Werten für  $\lambda$ . Als Schrittweite wählen wir in beiden Fällen  $h = 0.1$ . Der senkrechte Strich innerhalb der angegebenen Zahlen trennt die nach entsprechender Rundung korrekten von den fehlerhaften Ziffern.

$\lambda = -1$				$\lambda = -21$		
$x$	Euler explizit	Euler implizit	$y(x) = e^{-x}$	Euler explizit	Euler implizit	$y(x) = e^{-21x}$
0	1.0	1.0	1.0	1.0	1.0	1.0
0.1	0.9	0.9 09	0.904837	-1.1	0.32258	0.1224564
0.2	0.8 1	0.8 26	0.8187307	1.21	0.10406	0.015
0.3	0.7 29	0.7 51	0.74081822	-1.331	0.03357	$1.8 \cdot 10^{-3}$
0.4	0.6 56	0.6 83	0.67032005	1.4641	$1.1 \cdot 10^{-2}$	$2.2 \cdot 10^{-4}$
0.5	0.5 90	0.6 21	0.60653066	-1.6105	$3.5 \cdot 10^{-3}$	$2.8 \cdot 10^{-5}$
0.6	0.5 31	0. 564	0.54881164	1.77	$1.1 \cdot 10^{-3}$	$3.4 \cdot 10^{-6}$
0.7	0.4 78	0.5 13	0.49658530	-1.95	$3.6 \cdot 10^{-4}$	$4.1 \cdot 10^{-7}$
0.8	0.4 30	0. 467	0.44932896	2.14	$1.2 \cdot 10^{-4}$	$5.1 \cdot 10^{-8}$
0.9	0.3 87	0.4 24	0.40656966	-2.36	$3.8 \cdot 10^{-5}$	$6.2 \cdot 10^{-9}$
1.0	0. 349	0.3 85	0.36787944	2.6	$1.2 \cdot 10^{-5}$	$7.6 \cdot 10^{-10}$

△

## 8.2 Einschrittverfahren

### 8.2.1 Konsistenz, Stabilität und Konvergenz

Es seien jetzt allgemeiner als in (8.10)

$$a = x_0 < x_1 < x_2 < \cdots < x_N = b, \quad (8.22)$$

$$h_j := x_{j+1} - x_j, \quad h_{\max} = \max_j h_j.$$

Die Differenzialgleichung (8.1) wird durch ein Einschrittverfahren der Form

$$(A_h) : \begin{aligned} u_0 &= \tilde{y}_0 \\ u_{j+1} &= u_j + h_j f_h(x_j, u_j, x_{j+1}, u_{j+1}) \end{aligned} \quad (8.23)$$

gelöst.  $f_h$  heißt Verfahrensfunktion. Im Folgenden werden wir, wenn es der Zusammenhang erlaubt, oft  $h$  statt  $h_j$  oder  $h_{\max}$  verwenden.

**Definition 8.5.** *Lokaler Abschneide- oder Diskretisierungsfehler*

zum Verfahren (8.23) ist

$$\tau_h(x_j) := \frac{1}{h}(y(x_j + h) - y(x_j)) - f_h(x_j, y(x_j), x_j + h, y(x_j + h)). \quad (8.24)$$

Nach Multiplikation mit  $h$  gibt der lokale Abschneidefehler an, wie das Verfahren bei Ausführung eines Schrittes, also *lokal*, die Lösung verfälscht. Für ein explizites Verfahren ist dies leicht daran zu sehen, dass  $h \tau_h(x_j) = y(x_j + h) - u_{j+1}$ , falls  $u_j = y(x_j)$ .

**Definition 8.6.** *Konsistenz*

Das Einschrittverfahren  $(A_h)$  (8.23) heißt *konsistent* mit (A) (8.1), wenn

$$\left. \begin{aligned} u_0 &\rightarrow y_0 \\ \max_{x_j} \|\tau_h(x_j)\| &\rightarrow 0 \end{aligned} \right\} \quad \text{mit } h_{\max} \rightarrow 0. \quad (8.25)$$

**Definition 8.7.** *Konsistenzordnung*

Das Einschrittverfahren  $(A_h)$  hat die Konsistenzordnung  $p$ , falls es eine Konstante  $K$  gibt mit

$$\left. \begin{aligned} \|y_0 - \tilde{y}_0\| &\leq K h_{\max}^p \\ \max_{x_j} \|\tau_h(x_j)\| &\leq K h_{\max}^p \end{aligned} \right\} \quad \text{mit } h_{\max} \rightarrow 0. \quad (8.26)$$

**Definition 8.8.** *Globaler Abschneide- oder Diskretisierungsfehler*

$$e_h(x_j) := u_j - y(x_j). \quad (8.27)$$

**Definition 8.9.** *Konvergenz*

Das Verfahren  $(A_h)$  heißt *konvergent*, falls

$$\max_{x_j} \|u_j - y(x_j)\| \rightarrow 0 \quad \text{mit } h_{\max} \rightarrow 0. \quad (8.28)$$

Es besitzt die *Konvergenzordnung*  $p$ , falls es eine Konstante  $K$  gibt mit

$$\max_{x_j} \|u_j - y(x_j)\| \leq K h_{\max}^p \quad \text{mit } h_{\max} \rightarrow 0. \quad (8.29)$$

Wir sehen, dass die Konvergenzordnung über den globalen Abschneidefehler definiert wird. Es stellt sich aber heraus, dass im Falle der Konvergenz die Konvergenz- gleich der Konsistenzordnung ist.

*Konsistenz* ist nur eine *lokale* Aussage. *Konvergenz* ist eine *globale* Aussage, aber oft nur schwer zu zeigen. Verbindungsglied ist die *asymptotische Stabilität*, die das Verhalten der Lösung  $u_j$  als Näherung von  $y(\bar{x})$  bei festem  $\bar{x}$  mit  $h \rightarrow 0$  betrachtet; alle Standardverfahren erfüllen diese Voraussetzung, die asymptotische Stabilität ist also im Wesentlichen von theoretischem Interesse. Wir wollen sie hier nicht definieren, weil sie immer vorliegt, wenn die im Folgenden definierte Lipschitz-Bedingung (L) erfüllt ist.

**Voraussetzung 8.10.** (L): Lipschitz-Bedingung an  $f_h$ .

$$\begin{aligned} \|f_h(x_j, u_j, x_{j+1}, u_{j+1}) - f_h(x_j, \tilde{u}_j, x_{j+1}, \tilde{u}_{j+1})\| & \\ \leq L \max(\|u_j - \tilde{u}_j\|, \|u_{j+1} - \tilde{u}_{j+1}\|) & \end{aligned} \quad (8.30)$$

Der Term  $\|u_{j+1} - \tilde{u}_{j+1}\|$  auf der rechten Seite kommt nur für implizite Verfahren zum Tragen. Deshalb wird hin und wieder auf seine Berücksichtigung verzichtet ebenso wie auf die Variablen  $x_{j+1}$  und  $u_{j+1}$  in  $f_h$ .

**Satz 8.11. Konsistenz & Stabilität  $\Rightarrow$  Konvergenz**

Das durch die Verfahrensfunktion  $f_h$  definierte Verfahren sei konsistent und die Lipschitz-Bedingung (L) (8.30) sei erfüllt.

1. Dann gilt an jeder Stelle  $x_k$

$$\|u_k - y(x_k)\| \leq \left[ \|y_0 - \tilde{y}_0\| + \sum_{j=0}^{k-1} h_j \|\tau_h(x_j)\| \right] e^{L(x_k - a)}. \quad (8.31)$$

2. Es gilt Konvergenz für jeden Punkt  $\bar{x} = x_{j(h)} \in [a, b]$ :

$$\|u_j - y(\bar{x})\| \rightarrow 0 \text{ für } \begin{cases} \bar{x} = x_{j(h)} & \text{fest} \\ h_{\max} \rightarrow 0 & \\ j \rightarrow \infty & \end{cases}. \quad (8.32)$$

**Definition 8.12.** Es sei

$$E_L(x) := \begin{cases} \frac{e^{Lx} - 1}{L} & \text{für } L > 0, \\ x & \text{für } L = 0. \end{cases} \quad (8.33)$$

Dann heißt  $E_L(x)$  Lipschitz-Funktion.

**Satz 8.13.** Für das Einschrittverfahren gelte (L) und es sei konsistent mit der Ordnung  $p$ , d.h. es gibt ein  $M \geq 0$  mit

$$\|\tau_h(x_j)\| \leq M h^p \quad \forall x_j. \quad (8.34)$$

Dann gilt für den globalen Diskretisierungsfehler  $e_j := e_h(x_j)$

$$\|e_j\| = \|u_j - y(x_j)\| \leq M h_{\max}^p E_L(x_j - a) + \|\tilde{y}_0 - y_0\| e^{L(x_j - a)}. \quad (8.35)$$

Der Gesamtfehler eines numerischen Verfahrens zur Lösung eines Anfangswertproblems setzt sich aus dem Diskretisierungsfehler des Verfahrens und aus dem Rundungsfehler, der

sich von Schritt zu Schritt fortpflanzt, zusammen. Auch, wenn vom Rechenaufwand abgesehen werden kann, muss ein sehr kleines  $h$  nicht die optimale Wahl sein. Optimal ist dasjenige  $h$ , das den Gesamtfehler minimiert. In Abb. 8.3 wird die Situation für ein realistisches Verfahren und Beispiel dargestellt.

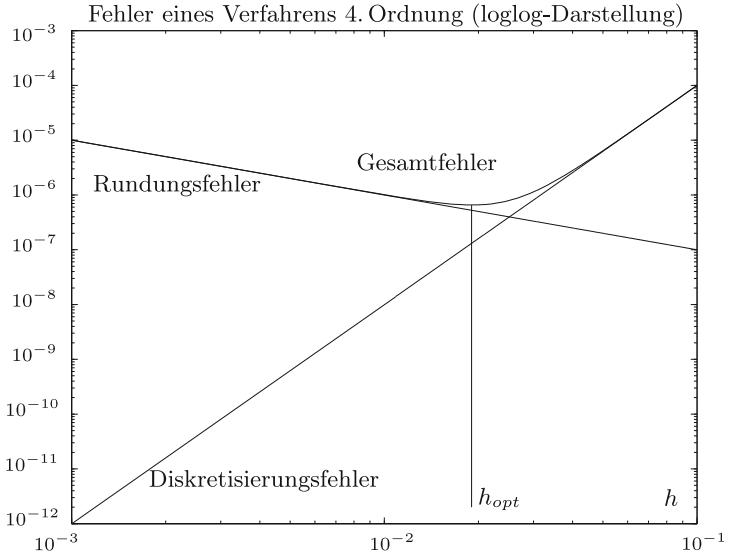


Abb. 8.3 Zusammensetzung des Gesamtfehlers.

### 8.2.2 Runge-Kutta-Verfahren

Die systematische Herleitung von Einschrittverfahren höherer Ordnung geht auf Runge und Kutta zurück, die vor mehr als hundert Jahren spezielle Methoden dritter und vieter Ordnung entwickelten, [Run 95, Kut 01]. Dabei wird die Verfahrensfunktion  $f_h(x, y)$  als Linearkombination von  $m$  Funktionen  $k_l(x, y)$  angesetzt. Es wird dementsprechend von einem  $m$ -stufigen Runge-Kutta-Verfahren gesprochen. Die  $k_l(x, y)$  stellen Funktionsauswertungen der rechten Seite  $f(x, y)$  an festzulegenden Stellen im Intervall  $[x_j, x_{j+1}]$  und den zugehörigen  $y$ -Werten dar. Damit lautet das allgemeine Runge-Kutta-Verfahren

$$\begin{aligned} u_{j+1} &= u_j + h_j \sum_{l=1}^m \gamma_l k_l(x_j, u_j) \quad \text{mit} \\ k_l(x, y) &= f \left( x + \alpha_l h_j, y + h_j \sum_{r=1}^m \beta_{lr} k_r(x, y) \right). \end{aligned} \tag{8.36}$$

Diese allgemeine Formulierung enthält die  $2m + m^2$  freien Parameter  $\gamma_l$ ,  $\alpha_l$  und  $\beta_{lr}$ . Alle auszuwertenden Funktionen  $k_l$  stehen in den Verfahrensgleichungen auf der linken und der rechten Seite. Das bedeutet, dass in jedem Schritt ein nichtlineares Gleichungssystem der Ordnung  $m \cdot n$  aufzulösen ist. Wir wollen deshalb auf die Allgemeinheit dieser Formulierung

verzichten und uns mit Familien expliziter und (halb)-impliziter Runge-Kutta-Verfahren beschäftigen, die wesentlich weniger freie Parameter enthalten.

### 8.2.3 Explizite Runge-Kutta-Verfahren

**Definition 8.14.** Ein Verfahren heißt explizites,  $m$ -stufiges Runge-Kutta-Verfahren, wenn die Verfahrensfunktion  $f_h$  eine Linearkombination von Funktionswerten  $f(x, y)$  an verschiedenen Stellen  $(x, y)$  ist

$$f_h(x, y) = \gamma_1 k_1(x, y) + \gamma_2 k_2(x, y) + \cdots + \gamma_m k_m(x, y) \quad (8.37)$$

mit

$$\begin{aligned} k_1(x, y) &= f(x, y) \\ k_2(x, y) &= f(x + \alpha_2 h, y + h\beta_{21} k_1(x, y)) \\ k_3(x, y) &= f(x + \alpha_3 h, y + h[\beta_{31} k_1(x, y) + \beta_{32} k_2(x, y)]) \\ &\vdots \\ k_m(x, y) &= f\left(x + \alpha_m h, y + h \sum_{j=1}^{m-1} \beta_{m,j} k_j(x, y)\right), \end{aligned} \quad (8.38)$$

wobei  $x = x_j$ ,  $y = u_j$  und  $h = h_j$  für Schritt  $j$ .

Das Verfahren ist also bestimmt durch die Festlegung der  $2m - 1 + m(m - 1)/2$  reellen Parameter

$$\gamma_1, \gamma_2, \dots, \gamma_m, \alpha_2, \alpha_3, \dots, \alpha_m, \beta_{21}, \beta_{31}, \beta_{32}, \beta_{41}, \dots, \beta_{m,m-1}.$$

**Beispiel 8.4.** Ein zweistufiges explizites Runge-Kutta-Verfahren ist das Heun-Verfahren:

$$u_{j+1} = u_j + h_j \left( \frac{1}{2} f(x_j, u_j) + \frac{1}{2} f(x_{j+1}, u_j + h_j f(x_j, u_j)) \right). \quad (8.39)$$

Hier sind also  $\gamma_1 = \gamma_2 = 1/2$ ,  $\alpha_2 = \beta_{21} = 1$  und damit

$$f_h(x, y) := \frac{1}{2} k_1(x, y) + \frac{1}{2} k_2(x, y) = \frac{1}{2} f(x, y) + \frac{1}{2} f(x + h, y + hf(x, y)). \quad \triangle$$

**Beispiel 8.5.** Das einstufige explizite Runge-Kutta-Verfahren:

$$f_h(x, y) = \gamma_1 k_1(x, y) = \gamma_1 f(x, y)$$

ist genau dann konsistent, wenn  $\gamma_1 = 1$  ist. Das ist das explizite Euler-Verfahren. Alle anderen Werte für  $\gamma_1$  liefern keine Konsistenz.  $\triangle$

### Allgemeine Koeffizienten-Bedingungen

Um ein Runge-Kutta-Verfahren zu konstruieren, müssen die Koeffizienten  $\gamma_l$ ,  $\alpha_l$  und  $\beta_{lr}$  so bestimmt werden, dass die Lipschitz-Bedingung (8.30) erfüllt ist, das Verfahren konsistent ist nach (8.25), und darüber hinaus eine möglichst hohe Konsistenzordnung erreicht wird. Aus diesen Forderungen ergeben sich zwei Bedingungen:

1. Für die Konsistenz muss der Gesamtzuwachs in einem Schritt  $hf$  sein, d.h.

$$\gamma_1 + \gamma_2 + \cdots + \gamma_m = 1. \quad (8.40)$$

Das werden wir in Lemma 8.15 beweisen.

2. Zum Erreichen einer möglichst hohen Konvergenzordnung sollte jedes  $k_l(x, y)$  wenigstens eine  $h^2$ -Approximation von  $y'(x + \alpha_l h)$  sein. Daraus folgt die Forderung

$$\alpha_i = \sum_{l=1}^{i-1} \beta_{il}, \quad i = 2, \dots, m. \quad (8.41)$$

Das wollen wir beispielhaft für  $i = 2$  zeigen:

$$\begin{aligned} y'(x + \alpha_2 h) &= f(x + \alpha_2 h, y(x + \alpha_2 h)) \\ &= f(x + \alpha_2 h, y(x) + \alpha_2 h f + O(h^2)) \\ &= f(x + \alpha_2 h, y + \alpha_2 h f) + O(h^2) f_y. \end{aligned}$$

Ist  $f \in C^1$  und  $\|f_y\| < L < \infty$ , dann liefert  $\beta_{21} = \alpha_2$  die gewünschte  $h^2$ -Approximation von  $y'(x + \alpha_2 h)$ .  $\|f_y\| < L < \infty$  folgt aber aus der globalen Lipschitz-Bedingung (8.7). Die restlichen Bedingungen ergeben sich entsprechend, wenn auch mit wesentlich komplexerer Rechnung.

**Lemma 8.15.** *Das  $m$ -stufige explizite Runge-Kutta-Verfahren mit der Verfahrensfunktion  $f_h$  nach (8.37) ist konsistent mit (8.1), falls (8.40) erfüllt ist.*

*Beweis.* Es gilt

$$\|f - f_h\| = \|f(x_j, y(x_j)) - \sum_{l=1}^m \gamma_l k_l(x_j, y(x_j))\| \rightarrow 0 \quad \text{mit } h_j \rightarrow 0,$$

weil mit  $h_j \rightarrow 0$  jedes  $k_l \rightarrow f(x_j, y(x_j))$  und damit wegen (8.40) auch die Linearkombination  $\sum \gamma_l k_l \rightarrow f(x_j, y(x_j))$ .

Es bleibt zu zeigen, dass aus  $\max_j \|f_h(x_j, y(x_j)) - f(x_j, y(x_j))\| \rightarrow 0$  mit  $h \rightarrow 0$  Konsistenz folgt. Dazu benutzen wir

$$\int_0^h y'(x_j + t) dt = y(x_j + h) - y(x_j) \quad \text{und} \quad \int_0^h y'(x_j) dt = y'(x_j) \int_0^h dt = h y'(x_j).$$

Daraus folgt

$$\begin{aligned} \max_j \left\| \frac{1}{h} (y(x_j + h) - y(x_j)) - y'(x_j) \right\| &= \max_j \left\| \frac{1}{h} \int_0^h [y'(x_j + t) - y'(x_j)] dt \right\| \\ &\leq \max_j \max_{0 \leq t \leq h} \|y'(x_j + t) - y'(x_j)\| \rightarrow 0 \quad \text{mit } h \rightarrow 0 \quad (y \in C^1). \end{aligned}$$

Nun ist aber

$$\begin{aligned} \max_j \|\tau_h(x_j)\| &= \max_j \left\| \frac{1}{h} (y(x_j + h) - y(x_j)) - f_h(x_j, y(x_j)) \right\| \\ &= \max_j \left\| \frac{1}{h} (y(x_j + h) - y(x_j)) \underbrace{- y'(x_j) + f(x_j, y(x_j)) - f_h(x_j, y(x_j))}_{\text{Einschub} = 0} \right\|. \end{aligned}$$

Mit dem ersten Teil des Beweises bekommt man also

$$\lim_{h \rightarrow 0} \max_j \|\tau_h(x_j)\| = \lim_{h \rightarrow 0} \max_j \|f(x_j, y(x_j)) - f_h(x_j, y(x_j))\|.$$

Daraus folgt, dass die Konvergenz von  $\max_j \|\tau_h(x_j)\|$  asymptotisch äquivalent ist mit der von  $\max_j \|f - f_h\|$ .  $\square$

Im folgenden Satz wollen wir die Konstruktion aller zweistufigen Runge-Kutta-Verfahren maximaler Konsistenzordnung herleiten. Er ist ein Beispiel für die Konstruktion von Runge-Kutta-Verfahren, das ahnen lässt, wie der Aufwand für die Bestimmung der Parameter mit wachsendem  $m$  anwächst.

Wir beschränken uns in der Darstellung auf eine skalare Differenzialgleichung. Die Aussage gilt aber ebenso für Systeme.

**Satz 8.16.** Seien  $n = 1$  und  $f \in C^3([a, b] \times \mathbb{R}, \mathbb{R})$ .

(i) Der lokale Abschneidefehler für ein explizites, zweistufiges Runge-Kutta-Verfahren lautet:

$$\begin{aligned} \tau_h(x_j) &= (1 - \gamma_1 - \gamma_2)f + \\ &\quad \frac{1}{2}h_j [(1 - 2\alpha_2 \gamma_2)f_x + (1 - 2\beta_{21} \gamma_2)ff_y] + O(h_j^2), \end{aligned} \tag{8.42}$$

d.h. es ist konsistent mit der Ordnung 2, falls

$$\gamma_1 + \gamma_2 = 1, \quad \alpha_2 = \beta_{21} = \frac{1}{2\gamma_2}, \quad \gamma_2 \neq 0. \tag{8.43}$$

Also gibt es unendlich viele explizite Runge-Kutta-Verfahren 2. Ordnung.

(ii) Es gibt kein konsistentes zweistufiges Runge-Kutta-Verfahren 3. Ordnung.

*Beweis.* Wir setzen  $x := x_j$ ,  $u := u_j$ ,  $h := h_j$ ,  $\nu := \alpha_2$  und  $\mu := \beta_{21}$ . (8.42) folgt aus einem Vergleich der Taylor-Reihen von  $f_h(x, u)$  und von  $(y(x+h) - y(x))/h$  für den lokalen Diskretisierungsfehler  $\tau_h(x)$ :

$$\begin{aligned} f_h(x, u) &= \gamma_1 f(x, u) + \gamma_2 [f + \nu h f_x + \mu h f f_y \\ &\quad + \frac{1}{2}\{\nu^2 h^2 f_{xx} + 2\nu\mu h^2 f f_{xy} + \mu^2 h^2 f^2 f_{yy}\}] + O(h^3), \end{aligned}$$

$$\begin{aligned} \frac{1}{h}(y(x+h) - y(x)) &= f(x, y) + \frac{h}{2}Df + \frac{h^2}{6}D^2f + O(h^3) \\ &= f + \frac{h}{2}(f_x + ff_y) + \frac{h^2}{6}(D(f_x + ff_y)) + O(h^3) \\ &= f + \frac{h}{2}(f_x + ff_y) + \\ &\quad \frac{h^2}{6}(f_{xx} + 2f_{xy}f + f_x f_y + ff_y^2 + f^2 f_{yy}) + O(h^3), \end{aligned}$$

also

$$\tau_h(x) = (1 - \gamma_1 - \gamma_2)f + \frac{h}{2}[(1 - 2\gamma_2\nu)f_x + (1 - 2\gamma_2\mu)ff_y]$$

$$+\frac{h^2}{6}[(1-3\nu^2\gamma_2)f_{xx}+(2-6\nu\mu\gamma_2)ff_{xy}\\ +(1-3\mu^2\gamma_2)f^2f_{yy}+f_xf_y+ff_y^2]+O(h^3).$$

Damit folgt (8.42). Es folgt auch (ii), weil als ein Faktor von  $h^2$  der Ausdruck  $(f_xf_y + ff_y^2)$  auftritt, der unabhängig von den Parametern  $\mu, \nu, \gamma_1, \gamma_2$  ist.  $\square$

**Beispiel 8.6.** Runge-Kutta-Verfahren 2. Ordnung:

1. Die verbesserte Polygonzug-Methode (Euler-Collatz-Verfahren)

$$\gamma_2 = 1 \Rightarrow \gamma_1 = 0, \quad \alpha_2 = \beta_{21} = \frac{1}{2}$$

$$\Rightarrow f_h(x, y) = f\left(x + \frac{h}{2}, y + \frac{1}{2}hf(x, y)\right)$$

2. Das Verfahren von Heun, siehe Beispiel 8.4.

3. "Optimales Verfahren": Will man möglichst viele  $h^2$ -Terme in  $\tau_h(x)$  unabhängig vom Vorzeichen verschwinden lassen, so kommt man zu

$$\gamma_2 = \frac{3}{4} \Rightarrow \gamma_1 = \frac{1}{4}, \quad \alpha_2 = \beta_{21} = \frac{2}{3}; \\ \Rightarrow f_h(x, y) = \frac{1}{4}f(x, y) + \frac{3}{4}f\left(x + \frac{2}{3}h, y + \frac{2}{3}hf(x, y)\right).$$

Es bleibt nur  $f_xf_y + ff_y^2$  im  $h^2$ -Term von  $\tau_h(x)$  übrig.  $\triangle$

Wir haben gesehen, dass man mit Stufe 2 Ordnung 2 erreichen kann; die allgemein erreichbaren Konsistenzordnungen für die Stufen  $m = 1, \dots, 10$  und die Anzahl der für die Parameter entstehenden nichtlinearen Gleichungen entnimmt man der folgenden Tabelle, [But 65]. Ab Stufe 8 verdoppelt sich die Anzahl der Bedingungen mindestens pro Stufe.

$m$	1	2	3	4	5	6	7	8	9	$m \geq 10$
Erreichbare Konsistenzordnung $p$	1	2	3	4	4	5	6	6	7	$\leq m - 2$
Anzahl Bedingungen zur Parameterbestimmung	1	2	4	8	8	17	37	37	85	$\geq 200$

Wegen des Stufensprungs zwischen den Ordnungen  $p = 4$  und  $p = 5$  werden Verfahren der Ordnung und Stufe  $m = p = 4$  besonders häufig angewendet. Für die vielen in der Literatur beschriebenen Verfahren 4. Ordnung [Gea 71, Gri 72, But 63] gilt wie für die in Beispiel 8.6 genannten Verfahren 2. Ordnung, dass Vorteile eines Verfahrens gegenüber einem anderen immer vom Beispiel abhängen. Mit anderen Worten: Alle Verfahren derselben Stufe und Ordnung sind von gleicher Güte. Wichtig ist allerdings der Gesichtspunkt der Verfahrenssteuerung, siehe Abschnitt 8.2.5.

Deswegen soll hier nur noch das so genannte klassische Runge-Kutta-Verfahren 4. Ordnung angegeben werden, das gegenüber anderen Verfahren den schönen Vorteil hat, das man es sich gut merken kann.

### Das klassische Runge-Kutta-Verfahren (1895, 1901)

$$\begin{aligned}
 k_1 &:= f(x_j, u_j) \\
 k_2 &:= f\left(x_j + \frac{h_j}{2}, u_j + \frac{h_j}{2}k_1\right) \\
 k_3 &:= f\left(x_j + \frac{h_j}{2}, u_j + \frac{h_j}{2}k_2\right) \\
 k_4 &:= f(x_j + h_j, u_j + h_j k_3) \\
 u_{j+1} &= u_j + \frac{h_j}{6}(k_1 + 2k_2 + 2k_3 + k_4)
 \end{aligned} \tag{8.44}$$

#### 8.2.4 Halbimplizite Runge-Kutta-Verfahren

Die allgemeine Form (8.36) der Runge-Kutta-Verfahren wird auch *voll implizit* genannt. Da solche Verfahren schwierig herzuleiten sind und ihre Durchführung sehr aufwändig ist, werden sie nur bei Problemen angewendet, die besondere Stabilitätsvorkehrungen erfordern, z.B. bei steifen Systemen, auf die wir in Abschnitt 8.4.4 eingehen werden. Hier wollen wir nur kurz halbimplizite Runge-Kutta-Verfahren definieren und ein Beispiel geben.

Das Verfahren (8.36) heißt *halbimplizit*, wenn  $\beta_{lr} = 0$  für  $r > l$ :

$$\begin{aligned}
 u_{j+1} &= u_j + h_j \sum_{l=1}^m \gamma_l k_l(x_j, u_j) \quad \text{mit} \\
 k_l(x, y) &= f\left(x + \alpha_l h, y + h \sum_{r=1}^l \beta_{lr} k_r(x, y)\right)
 \end{aligned} \tag{8.45}$$

Dann zerfällt das nichtlineare Gleichungssystem, das in jedem Schritt zu lösen ist, in  $m$  Systeme  $n$ -ter Ordnung, die sukzessiv gelöst werden können.

**Beispiel 8.7.** (Gauß-Form mit  $m = 1$ )  $\alpha_1 = \beta_{11} = \frac{1}{2}$  und  $\gamma_1 = 1$  liefern

$$u_{j+1} = u_j + h_j k_1, \quad k_1 = f\left(x_j + \frac{h_j}{2}, u_j + \frac{h_j}{2} k_1\right). \tag{8.46}$$

Der Beweis des folgenden Lemmas ist eine Übungsaufgabe.

**Lemma 8.17.** 1. Das Verfahren (8.46) hat die Konsistenzordnung  $p = 2$ .

2.  $f(x, y)$  sei zweimal stetig differenzierbar mit  $\|f_y\| \leq L$ . Weiter gelte  $L h/2 < 1$ .

Dann genügt ein Schritt des Newton-Verfahrens zum Erhalt der Konsistenzordnung  $p = 2$ , wenn für die Lösung des nichtlinearen Gleichungssystems (8.46) für  $k_1$  der Startwert  $k_1^{(0)} = f(x_j, u_j)$  gewählt wird, d.h. dann ist  $k_1^{(1)} - k_1 = O(h^2)$ .

Wir wollen die für (8.46) notwendige Vorgehensweise an einem konkreten Beispiel durchspielen. Auf die Differenzialgleichung

$$y' = y^2, \quad y(0) = -1 \quad \text{mit der exakten Lösung } y(x) = \frac{-1}{1+x}$$

soll ein Schritt des Verfahrens (8.46) mit  $h = h_0 = 0.1$  angewendet werden:

$$\begin{aligned} u_0 &= -1, \\ u_1 &= -1 + hk_1 \quad \text{mit } k_1 = (u_0 + \frac{h}{2}k_1)^2. \end{aligned} \quad (8.47)$$

Dann ergibt die Auflösung von (8.47) nach Division durch 0.05<sup>2</sup>

$$g(k_1) := k_1^2 - 440k_1 + 400 = 0 \Rightarrow k_1 = \begin{cases} 0.910977 \\ 439.1 \end{cases}.$$

Natürlich muss die obere Lösung genommen werden. Das ergibt

$$\begin{aligned} u_1 &= -1 + 0.0910977 = -0.908|9023 \quad \text{im Vergleich zu} \\ y(x_1) &= 0.9090. \end{aligned}$$

Was würde ein Schritt Newton-Verfahren ergeben?

$$\begin{aligned} g(k_1) &= 0, \quad g'(k_1) = 2k_1 - 440, \\ k_1^{(0)} &= f(x_0, u_0) = (-1)^2 = 1 \\ \Rightarrow k_1^{(1)} &= 1 - \frac{1^2 - 440 + 400}{2 - 440} = 0.91096 \\ \Rightarrow u_1 &= -1 + 0.091096 = -0.908|904. \end{aligned}$$

Also liefert hier ein Schritt Newton-Verfahren für (8.47) die gleiche Genauigkeit für  $u_1$  wie die exakte Lösung der nichtlinearen Gleichung (8.47).  $\triangle$

Die beim impliziten Runge-Kutta-Verfahren in der allgemeinen Form (8.36) entstehenden nichtlinearen Gleichungen können auch mit einem *Einzelschrittverfahren*

$$\begin{aligned} k_i^{(0)} &\quad \text{beliebig (siehe aber unten),} \\ \text{für } r &= 1, 2, \dots : \\ k_i^{(r)} &= f \left( x + \alpha_i h_j, y + h_j \sum_{l=1}^{i-1} \beta_{il} k_l^{(r)} + h_j \sum_{l=i}^m \beta_{il} k_l^{(r-1)} \right), \quad i = 1, \dots, m, \end{aligned} \quad (8.48)$$

gelöst werden. Dies entspricht dem Gauß-Seidel-Verfahren bei der iterativen Lösung von linearen Gleichungssystemen, siehe Kapitel 11. Unter milden Voraussetzungen konvergiert  $k_i^{(r)} \rightarrow k_i$  mit  $r \rightarrow \infty$  für  $i = 1, \dots, m$ , und es genügen  $p$  Schritte des Verfahrens, um die Konsistenzordnung  $p$  des Runge-Kutta-Verfahrens zu erhalten, [Gri 72]. Aber diese theoretische Aussage erweist sich praktisch als brauchbar nur für sehr kleine  $h$  und vernünftige Startwerte  $k_i^{(0)}$ . Deshalb ist meistens das Newton-Verfahren vorzuziehen.

### 8.2.5 Eingebettete Verfahren und Schrittweitensteuerung

Im Verlauf der numerischen Behandlung eines Differenzialgleichungssystems kann sich der Diskretisierungsfehler mit wachsendem  $x$  stark ändern. Deshalb spielt eine dem Problem angepasste Wahl der Schrittweite  $h_j$  eine wichtige Rolle. Grundsätzlich soll die Schrittweite über eine Fehlerschätzung gesteuert werden. Da die Abschätzungen des globalen Fehlers wie (8.35) oder (8.31) schwer zu bestimmen sind und darüber hinaus den tatsächlichen Fehler

meistens stark überschätzen, ist eine Schätzung

$$T(x, h) \doteq |\tau_h(x)| \quad (8.49)$$

des Betrags des lokalen Fehlers günstiger. Dafür gibt es unterschiedliche Methoden:

1. Schätzungen  $T$  für  $\tau_h$ , die vom speziellen Verfahren abhängen.

So geben z.B. Ceschino und Kuntzmann in [Ces 66] eine Schätzung des lokalen Abschneidefehlers für beliebige Runge-Kutta-Verfahren 4. Ordnung an.

2. Parallelrechnung mit zwei Schrittweiten  $h$  und  $qh$  und Schätzung des lokalen Diskretisierungsfehler mit Hilfe der beiden Ergebnisse

$$T(x, h) := \left| \frac{u(qh) - u(h)}{q^p - 1} \right|. \quad (8.50)$$

Hier bietet sich die zusätzliche Möglichkeit, das Ergebnis durch Extrapolation zu verbessern etwa wie in Abschnitt 3.1.6 dargestellt, siehe auch [Sto 02].

3. Parallelrechnung mit zwei Verfahren verschiedener Ordnung oder Stufe. Zur Aufwandsersparnis werden dabei *eingebettete Verfahren* verwendet. Das sind Verfahren, bei denen die Koeffizienten gemeinsamer Stufe übereinstimmen, sodass dann die Werte  $k_l$  nur einmal berechnet werden müssen.

Wir wollen zunächst auf die letzte Möglichkeit näher eingehen. Eine vierstufige Runge-Kutta-Methode 4. Ordnung, die in eine sechsstufigen Methode 5. Ordnung eingebettet wird, wurde von Englund [Eng 69] vorgeschlagen. Fehlberg [Feh 69a] hat diese Idee zu mehreren besseren Varianten weiterentwickelt, bei der der lokale Diskretisierungsfehler gegenüber Englund deutlich verringert werden kann. Eine dieser von Fehlberg vorgeschlagenen Methoden mit einfachen Zahlenwerten lautet

$$\begin{aligned} k_1 &= f(x_j, u_j) \\ k_2 &= f\left(x_j + \frac{2}{9}h, u_j + \frac{2}{9}hk_1\right) \\ k_3 &= f\left(x_j + \frac{1}{3}h, u_j + \frac{1}{12}hk_1 + \frac{1}{4}hk_2\right) \\ k_4 &= f\left(x_j + \frac{3}{4}h, u_j + \frac{69}{128}hk_1 - \frac{243}{128}hk_2 + \frac{135}{64}hk_3\right) \\ k_5 &= f\left(x_j + h, u_j - \frac{17}{12}hk_1 + \frac{27}{4}hk_2 - \frac{27}{5}hk_3 + \frac{16}{15}hk_4\right) \\ u_{j+1} &= u_j + h \left\{ \frac{1}{9}k_1 + \frac{9}{20}k_3 + \frac{16}{45}k_4 + \frac{1}{12}k_5 \right\} \end{aligned} \quad (8.51)$$

Dieses fünfstufige Verfahren 4. Ordnung wird eingebettet mit der Erweiterung

$$\begin{aligned} k_6 &= f\left(x_j + \frac{5}{6}h, u_j + \frac{65}{432}hk_1 - \frac{5}{16}hk_2 + \frac{13}{16}hk_3 + \frac{4}{27}hk_4 + \frac{5}{144}hk_5\right) \\ \hat{u}_{j+1} &= u_j + h \left\{ \frac{47}{450}k_1 + \frac{12}{25}k_3 + \frac{32}{225}k_4 + \frac{1}{30}k_5 + \frac{6}{25}k_6 \right\}. \end{aligned} \quad (8.52)$$

Schätzwert des lokalen Fehlers der Methode (8.51) ist der Betrag der Differenz  $\hat{u}_{j+1} - u_{j+1}$

$$T(x_{j+1}, h) = \frac{h}{300} |-2k_1 + 9k_3 - 64k_4 - 15k_5 + 72k_6|. \quad (8.53)$$

In [But 87, Feh 64, Feh 68, Feh 69b, Feh 70] sind weitere Kombinationen expliziter Runge-Kutta-Verfahren verschiedener Ordnung mit eingebauter Schrittweitensteuerung angegeben. Eine anders geartete Methode der automatischen Schrittweitensteuerung stammt von Zonneveld [Str 74, Zon 79]. Der lokale Diskretisierungsfehler wird mit Hilfe einer weiteren Funktionsauswertung so abgeschätzt, dass aus den berechneten  $k_i$ -Werten eine geeignete Linearkombination gebildet wird. Eine weitere interessante Idee besteht auch darin, den lokalen Fehler auf Grund einer eingebetteten Methode niedrigerer Ordnung zu schätzen [Dor 78, Dor 80, Hai 93].

Die Schätzung  $T$  des lokalen Fehlers kann auf unterschiedliche Weise zur Schrittweitensteuerung benutzt werden. Es wird eine Strategie benötigt, wann die Schrittweite beibehalten, vergrößert oder verkleinert wird. Eine mögliche Strategie haben wir in Abb. 8.4 veranschaulicht. Steuerungsparameter sind neben der Schätzung  $T$  die Fehlertoleranzen  $\varepsilon$  und  $\varepsilon/20$  sowie der Faktor (hier 2), um den die Schrittweite vergrößert bzw. verkleinert wird.

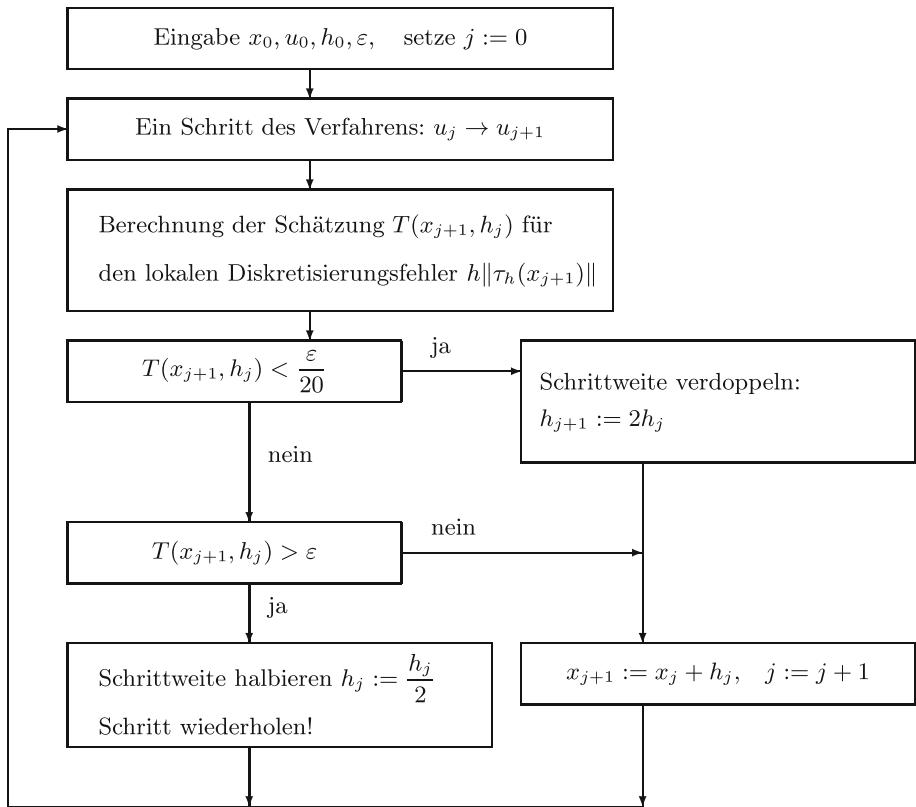


Abb. 8.4 Schema einer möglichen Strategie zur Schrittweitensteuerung.

Alternativ gibt es die Möglichkeit, die Schrittweite auf Grund des Fehlerschätzers nach jedem Schritt neu zu berechnen. Muss im Strategie-Schema der Schritt wiederholt werden, so wird die Schrittweite nicht halbiert wie in Abb. 8.4, sondern nach der Schätzformel neu

festgelegt. Konkreter schlägt Stoer [Sto 02] folgendes Vorgehen vor: Der Fehler wird durch Parallelrechnung mit  $h$  und  $h/2$  nach (8.50) geschätzt. Im Fall der Schrittweiterholung wird die neue Schrittweite festgelegt als

$$h_{\text{neu}} = \frac{h_{\text{alt}} \sqrt[p+1]{\varepsilon}}{\sqrt[p+1]{2^p T}} \quad (8.54)$$

mit der vorgegebenen Toleranz  $\varepsilon$  und der Fehlerschätzung  $T$ . Der Schrittweiten-Schätzer (8.54) beruht auf einer Schätzung des globalen Fehlers, siehe [Sto 02].

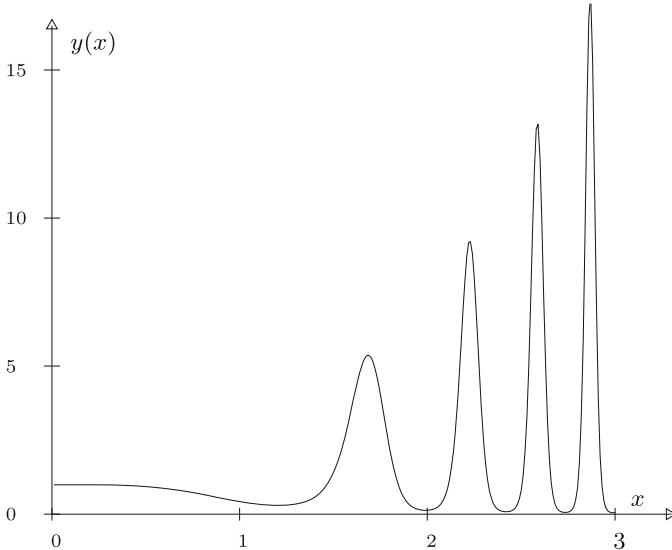


Abb. 8.5 Schwingung mit wachsender Amplitude und Frequenz.

**Beispiel 8.8.** Es soll die lineare Differenzialgleichung

$$y' = -(\sin(x^3) + 3x^3 \cos(x^3)) y, \quad y(0) = 1 \quad (8.55)$$

mit dem klassischen Runge-Kutta-Verfahren (8.44) und der gerade beschriebenen Schrittweitensteuerung nach Stoer [Sto 02] gelöst werden. Die analytische Lösung ist

$$y(x) = \exp(-x \sin(x^3)).$$

Sie ähnelt einem glatten Einschwingvorgang, dem eine in Frequenz und Amplitude wachsende Schwingung folgt, siehe Abb. 8.5.

Wir führen jetzt mit einfacher Genauigkeit, also mit etwa acht wesentlichen Dezimalstellen, die folgenden drei Rechnungen durch:

- (a) Schrittweitensteuerung nach (8.50)/(8.54) mit der kleinen Toleranz  $\varepsilon = 1 \cdot 10^{-7}$ . Es werden 4368 Funktionsauswertungen benötigt.
- (b) Wir verbrauchen dieselbe Zahl von 4368 Funktionsauswertungen, rechnen aber mit äquidistanter Schrittweite.
- (c) Wir rechnen wieder mit äquidistanter Schrittweite, machen die Schrittweite aber so klein, dass der Fehler am Endpunkt etwa gleich dem aus der ersten Rechnung ist.

Die Ergebnisse haben wir in der folgenden Tabelle zusammengefasst. Sie zeigen deutlich, dass man ohne Schrittweitensteuerung bei einem Beispiel wie diesem entweder den 20-fachen Fehler oder den doppelten Aufwand in Kauf nehmen muss.

Methode	# Fkts.ausw.	$ u_{x=3} - y(3) $	$h_{\min}$	$h_{\max}$
(a)	4368	$0.145 \cdot 10^{-4}$	$2.7 \cdot 10^{-3}$	0.5
(b)	4368	$3.170 \cdot 10^{-4}$	$8.2 \cdot 10^{-3}$	$8.2 \cdot 10^{-3}$
(c)	8784	$0.144 \cdot 10^{-4}$	$4.1 \cdot 10^{-3}$	$4.1 \cdot 10^{-3}$

△

## 8.3 Mehrschrittverfahren

In Abschnitt 8.1.1 haben wir mit der Mittelpunktregel  $u_{j+2} = u_j + 2hf_{j+1}$  schon ein Mehrschrittverfahren kennen gelernt. Jetzt sollen diese systematischer behandelt werden. Dabei werden wir zunächst auf eine Familie von expliziten und impliziten Methoden eingehen, bevor wir zur allgemeinen Konvergenztheorie und Verfahrenskonstruktion kommen.

Allgemein betrachten wir  $m$ -Schritt-Verfahren. Dabei ist  $m$  die Anzahl der Werte  $u_{j+i}$ ,  $i = 0, 1, \dots, m-1$ , die an den als äquidistant vorausgesetzten Stellen  $x_{j+i}$  vorliegen müssen. Sie werden zur Berechnung des neuen Wertes  $u_{j+m}$  verwendet.  $u_{j+m}$  stellt eine Näherung für  $y(x_{j+m})$  dar.

### 8.3.1 Verfahren vom Adams-Typ

Zur Herleitung einer wichtigen Klasse von Mehrschrittverfahren, den Verfahren vom Adams-Typ, wird folgende Integraldarstellung der exakten Lösung benutzt:

$$y(x_{j+m}) = y(x_{j+m-1}) + \int_{x_{j+m-1}}^{x_{j+m}} f(x, y(x)) dx. \quad (8.56)$$

Das Integral in (8.56) wird numerisch approximiert, indem  $f$  (komponentenweise) durch ein Polynom  $P$  vom Höchstgrad  $r$  interpoliert wird

$$P(x_{j+k}) = f(x_{j+k}, u_{j+k}), \quad k = 0, 1, \dots, r,$$

und dieses exakt integriert wird

mit  $r = m \implies$  implizites oder Interpolationsverfahren,  
oder  $r = m - 1 \implies$  explizites oder Extrapolationsverfahren.

Die so gewonnenen expliziten Verfahren bilden die Familie der *Adams-Bashforth-Methoden*, die impliziten die der *Adams-Moulton-Methoden*.

#### Adams-Bashforth für $m = 2$

Das Polynom  $P(x)$  entsteht durch lineare Interpolation der beiden Punkte  $(x_j, f_j)$  und  $(x_{j+1}, f_{j+1})$ . Wir transformieren das Intervall  $x \in [x_j, x_{j+2}]$  auf  $t \in [0, 2]$ . Dann ergibt sich

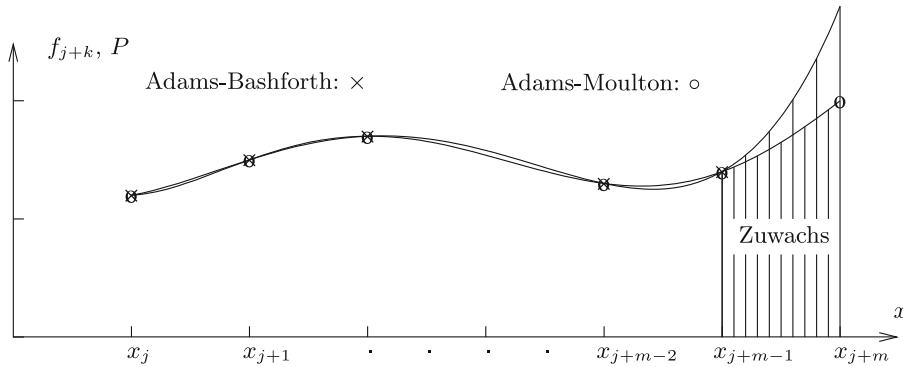


Abb. 8.6 Konstruktion der Adams-Methoden.

mit der Vorwärtsdifferenz  $\Delta f_j = f_{j+1} - f_j$

$$P(x) = P(x_j + th) = f_j + t\Delta f_j. \quad (8.57)$$

Integration dieses Polynoms

$$\int_{x_{j+1}}^{x_{j+2}} P(x) dx = h \int_1^2 P(x_j + th) dt = h \left[ t f_j + \frac{t^2}{2} \Delta f_j \right]_1^2 = h \left[ \frac{3}{2} f_{j+1} - \frac{1}{2} f_j \right]$$

ergibt mit (8.56) das Verfahren

$$u_{j+2} = u_{j+1} + \frac{h}{2} (3f_{j+1} - f_j). \quad (8.58)$$

Bevor wir uns im Abschnitt 8.3.2 genauer mit der Theorie der Mehrschrittverfahren auseinander setzen, wollen wir für das einfache Verfahren (8.58) den lokalen Diskretisierungsfehler (8.24) berechnen.

$$\begin{aligned} \tau_h(x_{j+2}) &:= \frac{1}{h} \{y(x_{j+2}) - y(x_{j+1})\} - \frac{1}{2} \{3f(x_{j+1}, y(x_{j+1})) - f(x_j, y(x_j))\} \\ &= \frac{1}{h} (y(x_{j+2}) - y(x_{j+1})) - \frac{1}{2} \{3y'(x_{j+1}) - y'(x_j)\} \\ &= y'(x_{j+1}) + \frac{h}{2} y''(x_{j+1}) + \frac{h^2}{6} y^{(3)}(x_{j+1}) + O(h^3) \\ &\quad - \frac{1}{2} \left\{ 3y'(x_{j+1}) - \left[ y'(x_{j+1}) - hy''(x_{j+1}) + \frac{h^2}{2} y^{(3)}(x_{j+1}) + O(h^3) \right] \right\} \\ &= \frac{5}{12} h^2 y^{(3)}(x_{j+1}) + O(h^3) \\ &=: C_3 h^2 y^{(3)}(x_{j+1}) + O(h^3) \end{aligned} \quad (8.59)$$

Damit ist gezeigt, dass das Verfahren (8.58) die Ordnung  $p = 2$  hat. Der Faktor  $C_3 = 5/12$  heißt *Fehlerkonstante* des Verfahrens, siehe Abschnitt 8.3.2.

**Adams-Moulton für  $m = 2$** 

$P(x)$  entsteht jetzt durch Interpolation von  $(x_j, f_j)$ ,  $(x_{j+1}, f_{j+1})$ ,  $(x_{j+2}, f_{j+2})$ . Das Newtonsche Interpolationsschema ergibt wegen der äquidistanten Stützstellen

$$P(x) = f_j + (x - x_j)\Delta f_j + (x - x_j)(x - x_{j+1})\frac{\Delta^2 f_j}{2}$$

oder

$$P(x_j + th) = f_j + t\Delta f_j + \frac{t(t-1)}{2}\Delta^2 f_j. \quad (8.60)$$

Der Zuwachs berechnet sich wieder durch Integration dieses Polynoms

$$\begin{aligned} \int_{x_{j+1}}^{x_{j+2}} P(x) dx &= h \int_1^2 P(x_j + th) dt = h \left[ f_j t + \Delta f_j \frac{t^2}{2} + \Delta^2 f_j \left( \frac{t^3}{6} - \frac{t^2}{4} \right) \right]_1^2 \\ &= h \left[ f_j + \frac{3}{2}(f_{j+1} - f_j) + \left( \frac{7}{6} - \frac{3}{4} \right) (f_{j+2} - 2f_{j+1} + f_j) \right] \\ &= \frac{h}{12} [-f_j + 8f_{j+1} + 5f_{j+2}]. \end{aligned}$$

Das ergibt das Verfahren

$$u_{j+2} = u_{j+1} + \frac{h}{12} [-f_j + 8f_{j+1} + 5f_{j+2}]. \quad (8.61)$$

Seinen lokalen Diskretisierungsfehler rechnen wir ganz analog zu (8.59) aus.

$$\begin{aligned} \tau_h(x_{j+2}) &:= \frac{1}{h} \{y(x_{j+2}) - y(x_{j+1})\} - \frac{1}{12} \{-y'(x_j) + 8y'(x_{j+1}) + 5y'(x_{j+2})\} \\ &= y'(x_{j+1}) + \frac{h}{2} y''(x_{j+1}) + \frac{h^2}{6} y^{(3)}(x_{j+1}) + \frac{h^3}{24} y^{(4)}(x_{j+1}) + O(h^4) \\ &\quad - \frac{1}{12} \left\{ -y'(x_{j+1}) + hy''(x_{j+1}) - \frac{h^2}{2} y^{(3)}(x_{j+1}) + \frac{h^3}{6} y^{(4)}(x_{j+1}) \right. \\ &\quad \left. + 8y'(x_{j+1}) \right. \\ &\quad \left. + 5[y'(x_{j+1}) + hy''(x_{j+1}) + \frac{h^2}{2} y^{(3)}(x_{j+1}) + \frac{h^3}{6} y^{(4)}(x_{j+1})] + O(h^4) \right\} \\ &= -\frac{h^3}{24} y^{(4)}(x_{j+1}) + O(h^4) \end{aligned} \quad (8.62)$$

Das Adams-Moulton-Verfahren mit  $m = 2$  hat also die Ordnung  $p = m + 1 = 3$  und die Fehlerkonstante  $C_3 = -\frac{1}{24}$ .

**Adams-Verfahren höherer Ordnung**

Diese ergeben sich völlig analog, wobei die Interpolation wegen der äquidistanten Stützstellen mit Vorwärtsdifferenzen besonders einfach durchzuführen ist. Wir geben noch die expliziten Drei- bis Sechsschrittverfahren an.

$$u_{j+3} = u_{j+2} + \frac{h}{12} \{23f_{j+2} - 16f_{j+1} + 5f_j\}, \quad (8.63)$$

$$u_{j+4} = u_{j+3} + \frac{h}{24} \{55f_{j+3} - 59f_{j+2} + 37f_{j+1} - 9f_j\}, \quad (8.64)$$

$$u_{j+5} = u_{j+4} + \frac{h}{720} \{1901f_{j+4} - 2774f_{j+3} + 2616f_{j+2} - 1274f_{j+1} + 251f_j\}, \quad (8.65)$$

$$\begin{aligned} u_{j+6} = u_{j+5} + \frac{h}{1440} & \{4277f_{j+5} - 7923f_{j+4} + 9982f_{j+3} \\ & - 7298f_{j+2} + 2877f_{j+1} - 475f_j\}. \end{aligned} \quad (8.66)$$

Jedes explizite  $m$ -Schrittverfahren vom Typ Adams-Bashforth hat die Ordnung  $p = m$ . Dabei erfordern alle diese Verfahren pro Schritt nur eine Funktionsauswertung. Zudem erlaubt die Kombination von zwei Verfahren verschiedener Ordnung eine praktisch kostenlose Schätzung des lokalen Diskretisierungsfehlers. So ergibt sich beispielsweise für das Vierschrittverfahren (8.64) in Kombination mit (8.65) der Schätzwert

$$T(x_{j+4}, h) = \frac{h}{720} \{251f_{j+3} - 1004f_{j+2} + 1506f_{j+1} - 1004f_j + 251f_{j-1}\}, \quad (8.67)$$

der zur Schrittweitensteuerung verwendet werden kann. Der lokale Diskretisierungsfehler einer Adams-Bashforth-Methode ist stets bedeutend größer als der eines Runge-Kutta-Verfahrens gleicher Ordnung.

Als implizite Drei-, Vier- und Fünfschrittverfahren vom Typ Adams-Moulton ergeben sich

$$u_{j+3} = u_{j+2} + \frac{h}{24} \{9f(x_{j+3}, u_{j+3}) + 19f_{j+2} - 5f_{j+1} + f_j\} \quad (8.68)$$

$$\begin{aligned} u_{j+4} = u_{j+3} + \frac{h}{720} & \{251f(x_{j+4}, u_{j+4}) \\ & + 646f_{j+3} - 264f_{j+2} + 106f_{j+1} - 19f_j\} \end{aligned} \quad (8.69)$$

$$\begin{aligned} u_{j+5} = u_{j+4} + \frac{h}{1440} & \{475f(x_{j+5}, u_{j+5}) + 1427f_{j+4} - 798f_{j+3} \\ & + 482f_{j+2} - 173f_{j+1} + 27f_j\}. \end{aligned} \quad (8.70)$$

Jedes implizite  $m$ -Schrittverfahren vom Typ Adams-Moulton hat die Ordnung  $p = m + 1$ .

Die Berechnung von  $u_{j+m}$  aus der impliziten Gleichung in jedem Integrationsschritt wird mit der Prädiktor-Korrektor-Technik vermieden. Eine Startnäherung  $u_{j+m}^{(0)}$  für eine implizite  $m$ -Schrittmetode von Adams-Moulton wird mit der expliziten  $m$ -Schrittmetode von Adams-Bashforth bestimmt. Die implizite Formel wird nur dazu verwendet, diese Startnäherung mit einem einzigen Schritt der Fixpunkt-Iteration zu verbessern. Eine solche Kombination von zwei Dreischrittverfahren zu einer *Prädiktor-Korrektor-Verfahren* lautet

$$\begin{aligned} u_{j+3}^{(P)} = u_{j+2} + \frac{h}{12} & \{23f_{j+2} - 16f_{j+1} + 5f_j\}, \\ u_{j+3} = u_{j+2} + \frac{h}{24} & \{9f(x_{j+3}, u_{j+3}^{(P)}) + 19f_{j+2} - 5f_{j+1} + f_j\}. \end{aligned} \quad (8.71)$$

Dieses *Adams-Bashforth-Moulton-Verfahren*, kurz als A-B-M-Verfahren bezeichnet, besitzt die Ordnung  $p = 4$ . Es kann gezeigt werden, dass eine Prädiktor-Korrektor-Methode, die durch Kombination eines expliziten  $m$ -Schritt-Prädiktors der Ordnung  $m$  mit einem impliziten  $m$ -Schritt-Korrektor der Ordnung  $m + 1$  erklärt ist, die Ordnung  $p = m + 1$  besitzt

[Gri 72, Lam 91]. Die entsprechende Analyse zeigt, dass der Koeffizient  $C_{m+2}^{(PC)}$  von  $h^{m+2}$  des Hauptanteils des lokalen Diskretisierungsfehlers als Linearkombination der Konstanten  $C_{m+1}^{(P)}$  und  $C_{m+2}^{(C)}$  der beiden Verfahren gegeben ist, wobei problembedingt verschiedene Ableitungen von  $y(x)$  zur Bildung von  $C_{m+2}^{(PC)}$  auftreten. Da die Koeffizienten  $C_{m+1}^{(AB)}$  der Adams-Bashforth-Verfahren betragsmäßig bedeutend größer als die Konstanten  $C_{m+2}^{(AM)}$  der Adams-Moulton-Methoden sind (vgl. Tab. 8.3), wird die Größe des lokalen Diskretisierungsfehlers der Prädiktor-Korrektor-Methode im Wesentlichen durch den Hauptanteil der expliziten Prädiktor-Methode bestimmt.

Die Situation wird verbessert, wenn als Prädiktorformel eine Adams-Bashforth-Methode mit der gleichen Ordnung wie die Korrektorformel verwendet wird. Wir kombinieren somit zwei Verfahren mit verschiedener Schrittzahl. Als Beispiel formulieren wir das Prädiktor-Korrektor-Verfahren, welches aus der expliziten Vierschrittmetode von Adams-Bashforth (8.64) als Prädiktor und aus der impliziten Dreischrittmetode von Adams-Moulton (8.68) besteht.

$$\begin{aligned} u_{j+4}^{(P)} &= u_{j+3} + \frac{h}{24} \{55f_{j+3} - 59f_{j+2} + 37f_{j+1} - 9f_j\} \\ u_{j+4} &= u_{j+3} + \frac{h}{24} \{9f(x_{j+4}, u_{j+4}^{(P)}) + 19f_{j+3} - 5f_{j+2} + f_{j+1}\} \end{aligned} \quad (8.72)$$

Das Integrationsverfahren (8.72) besitzt wie (8.71) die Ordnung  $p = 4$ . Die Analyse des Diskretisierungsfehlers zeigt aber, dass der Koeffizient des Hauptanteils  $C_5^{(PC)} = -\frac{19}{720} = C_5^{(AM)}$  ist. Bei solchen Kombinationen von Prädiktor-Korrektor-Verfahren ist stets der Koeffizient des Hauptanteils der Korrektorformel maßgebend [Gri 72, Lam 91]. Der lokale Diskretisierungsfehler der Methode (8.72) ist deshalb kleiner als derjenige von (8.71). Die praktische Anwendung der Prädiktor-Korrektor-Methode (8.72) erfordert selbstverständlich vier Startwerte  $u_0, u_1, u_2, u_3$ , weil dem Prädiktor ein Vierschrittverfahren zu Grunde liegt.

Tab. 8.1 Runge-Kutta-Methode und A-B-M-Verfahren.

$x_j$	$y(x_j)$	Runge-Kutta $u_j$	A-B-M	(8.71)	A-B-M	(8.72)
			$u_j^{(P)}$	$u_j$	$u_j^{(P)}$	$u_j$
1.0	0	0				
1.1	0.2626847	0.2626829				
1.2	0.5092384	0.5092357				
1.3	0.7423130	0.7423100				
1.4	0.9639679	0.9639648	0.9641567	0.9639538	0.9638945	0.9639751
1.5	1.1758340	1.1758309	1.1759651	1.1758147	1.1757894	1.1758460
1.6	1.3792243	1.3792213	1.3793138	1.3792027	1.3792028	1.3792382
1.7	1.5752114	1.5752087	1.5752742	1.5751894	1.5751995	1.5752258
1.8	1.7646825	1.7646799	1.7647268	1.7646610	1.7646775	1.7646965
1.9	1.9483792	1.9483769	1.9484106	1.9483588	1.9483785	1.9483924
2.0	2.1269280	2.1269258	2.1269503	2.1269089	2.1269298	2.1269401

**Beispiel 8.9.** Wir behandeln die Anfangswertaufgabe  $y' = xe^{x-y}$ ,  $y(1) = 0$ , mit den beiden A-B-M-Methoden (8.71) und (8.72) und wählen die Schrittweite  $h = 0.1$ . Um die beiden Verfahren unter gleichen Bedingungen vergleichen zu können, werden in beiden Fällen die drei Startwerte  $u_1, u_2, u_3$  mit der klassischen Runge-Kutta-Methode (8.44) berechnet. Tab. 8.1 enthält die Näherungswerte, die sowohl mit dem Runge-Kutta-Verfahren vierter Ordnung (8.44) als auch mit den beiden Prädiktor-Korrektor-Methoden erhalten worden sind, wobei für diese Verfahren auch die Prädiktorwerte  $u_j^{(P)}$  angegeben sind. Die Mehrschrittmethoden arbeiten etwas ungenauer als das Runge-Kutta-Verfahren.  $\triangle$

### 8.3.2 Konvergenztheorie und Verfahrenskonstruktion

**Definition 8.18.** ( $A_h$ ): *Lineares Mehrschrittverfahren (MSV)*

(i) Zur Lösung von (8.1) im Intervall  $[a, b]$  sei eine konstante Schrittweite  $h > 0$  gegeben mit

$$x_j = a + jh, \quad j = 0, 1, \dots, N, \quad h = \frac{b - a}{N}.$$

(ii) Start- oder Anlaufrechnung

Es sei ein besonderes Verfahren (meistens ein Einschrittverfahren) gegeben, das die Werte

$$u_0, u_1, \dots, u_{m-1}$$

an den Stellen  $x_0, x_1, \dots, x_{m-1}$  berechnet.

(iii) Es seien reelle Zahlen  $a_0, a_1, \dots, a_m$  und  $b_0, b_1, \dots, b_m$  gegeben mit  $a_m = 1$ . Bekannt seien die Werte  $u_j, u_{j+1}, \dots, u_{j+m-1}$ .

Gesucht ist die Lösung  $u_{j+m}$  der Gleichungssysteme

$$\begin{aligned} \frac{1}{h} \sum_{k=0}^m a_k u_{j+k} &= f_h(x_j, u_j, u_{j+1}, \dots, u_{j+m}), \quad j = 0, 1, \dots, N-m, \\ \text{mit } f_h &:= \sum_{k=0}^m b_k f(x_{j+k}, u_{j+k}) =: \sum_{k=0}^m b_k f_{j+k}. \end{aligned} \tag{8.73}$$

(iv) Das Mehrschrittverfahren heißt *explizit*, falls  $b_m = 0$ , und *implizit*, falls  $b_m \neq 0$ .

*Explizite Mehrschrittverfahren* lassen sich nach  $u_{j+m}$  auflösen:

$$u_{j+m} = - \sum_{k=0}^{m-1} a_k u_{j+k} + h \sum_{k=0}^{m-1} b_k f_{j+k}. \tag{8.74}$$

Bei *impliziten Mehrschrittverfahren* muß man wie bei Einschrittverfahren das i.a. nichtlineare Gleichungssystem

$$u_{j+m} = g(u_{j+m})$$

lösen mit

$$g(u_{j+m}) := - \sum_{k=0}^{m-1} a_k u_{j+k} + h \sum_{k=0}^m b_k f_{j+k}.$$

Dazu reichen oft wenige Schritte des Einzelschrittverfahrens (wie (8.48)) oder der sukzessiven Iteration aus:

$$u_{j+m}^{(0)} = u_{j+m-1},$$

$$u_{j+m}^{(l)} = g(u_{j+m}^{(l-1)}), \quad l = 1, 2, \dots$$

Diese Folge konvergiert, wenn

$$q := h|b_m|L < 1,$$

wo  $L$  Lipschitz-Konstante von  $f$  bezüglich  $y$  ist. Die Anzahl der benötigten Schritte kann vermindert werden, wenn statt des Startwertes  $u_{j+m}^{(0)} = u_{j+m-1}$  ein expliziter Euler-Schritt von  $u_{j+m-1}$  zu  $u_{j+m}^{(0)}$  vollzogen wird, also

$$u_{j+m}^{(0)} = u_{j+m-1} + hf(x_{j+m-1}, u_{j+m-1}).$$

**Definition 8.19.** 1. Sei  $(A_h)$  ein Mehrschrittverfahren mit eindeutig bestimmter Lösung.

Der *lokale Abschneidefehler* wird in Analogie zu (8.24) definiert als

$$\begin{aligned} \tau_h(x_{j+m}) &:= \frac{1}{h} \sum_{k=0}^m a_k y(x_{j+k}) - \sum_{k=0}^m b_k f(x_{j+k}, y(x_{j+k})), \\ j &= 0, 1, \dots, N-m. \end{aligned} \quad (8.75)$$

mit den Startwerten

$$\tau_h(x_j) := u_j - y(x_j) \text{ für } j = 0, 1, \dots, m-1.$$

2.  $(A_h)$  heißt *konsistent* mit  $(A)$ , wenn

$$\max_j \|\tau_h(x_j)\| \rightarrow 0 \quad \text{für } h \rightarrow 0, \quad j = 0, 1, \dots, N. \quad (8.76)$$

3.  $(A_h)$  besitzt die *Konsistenzordnung*  $p$ , wenn es ein  $K \in \mathbb{R}$  gibt mit

$$\max_j \|\tau_h(x_j)\| \leq Kh^p \text{ mit } h \rightarrow 0, \quad j = 0, 1, \dots, N.$$

4. 1. und 2. erzeugendes oder charakteristisches Polynom des Mehrschrittverfahrens  $(A_h)$  sind

$$\begin{aligned} \varrho(\zeta) &:= \sum_{j=0}^m a_j \zeta^j, \\ \sigma(\zeta) &:= \sum_{j=0}^m b_j \zeta^j. \end{aligned} \quad (8.77)$$

**Satz 8.20.**  $(A_h)$  hat mindestens die Konsistenzordnung  $p = 1$ , falls  $f \in C^1$ ,  $u_j \rightarrow y_0$  mit  $h \rightarrow 0$  für  $j = 0, 1, \dots, m-1$  und

$$\varrho(1) = 0, \quad \varrho'(1) = \sigma(1). \quad (8.78)$$

*Beweis.* Taylor-Entwicklung von  $\tau_h$  um  $x_j$ :

$$\begin{aligned} \tau_h(x_{j+m}) &= \frac{1}{h} \sum_{k=0}^m a_k y(x_{j+k}) - \sum_{k=0}^m b_k f(x_{j+k}, y(x_{j+k})) \\ &= \frac{1}{h} \sum_{k=0}^m a_k y(x_{j+k}) - \sum_{k=0}^m b_k y'(x_{j+k}) \\ &= \frac{1}{h} \sum_{k=0}^m a_k \left[ y(x_j) + k h y'(x_j) + \frac{1}{2}(k h)^2 y''(\xi_{j+k}) \right] \end{aligned}$$

$$\begin{aligned}
& - \sum_{k=0}^m b_k [y'(x_j) + k h y''(\eta_{j+k})] \\
& \text{mit Zwischenstellen } \xi_j, \dots, \xi_{j+m}, \eta_j, \dots, \eta_{j+m} \in [a, b] \\
& = \frac{1}{h} y(x_j) \sum_{k=0}^m a_k + y'(x_j) \left[ \sum_{k=0}^m k a_k - \sum_{k=0}^m b_k \right] \\
& \quad + h \underbrace{\left[ \sum_{k=0}^m \frac{k^2}{2} a_k y''(\xi_{j+k}) - \sum_{k=0}^m k b_k y''(\eta_{j+k}) \right]}_{(*)} \\
& = \frac{1}{h} y(x_j) \varrho(1) + y'(x_j) [\varrho'(1) - \sigma(1)] + O(h) \\
& = O(h).
\end{aligned}$$

(\*): Dieser Ausdruck ist beschränkt, weil  $y''$  mit  $f \in C^1$  im kompakten Intervall  $[a, b]$  beschränkt ist.  $\square$

Eine Verallgemeinerung dieser Vorgehensweise liefert:

**Lemma 8.21.** 1.  $(A_h)$  besitzt genau die Konsistenzordnung  $p$  für alle  $f \in C^p$ , wenn die sog. Fehlerkonstanten

$$\begin{aligned}
C_0 &= a_0 + a_1 + \dots + a_m, \\
C_1 &= a_1 + 2a_2 + \dots + m a_m - (b_0 + b_1 + b_2 + \dots + b_m) \quad \text{und} \\
C_i &= \frac{1}{i!} (a_1 + 2^i a_2 + \dots + m^i a_m) \\
&\quad - \frac{1}{(i-1)!} (b_1 + 2^{i-1} b_2 + \dots + m^{i-1} b_m), \quad i = 2, \dots, p,
\end{aligned} \tag{8.79}$$

verschwinden, aber

$$C_{p+1} = \frac{1}{(p+1)!} \left( \sum_{k=0}^m a_k k^{p+1} - (p+1) \sum_{k=0}^m b_k k^p \right) \neq 0. \tag{8.80}$$

2.  $C_i = 0 \iff D_i(t) = 0$ ,  $t \in \mathbb{R}$ , wo  $D_0 = C_0$  und

$$\begin{aligned}
D_i(t) &= \frac{1}{i!} ((-t)^i a_0 + (1-t)^i a_1 + \dots + (m-t)^i a_m) \\
&\quad - \frac{1}{(i-1)!} \{ (-t)^{i-1} b_0 + (1-t)^{i-1} b_1 + \dots + (m-t)^{i-1} b_m \}.
\end{aligned} \tag{8.81}$$

*Beweis.* (8.79) ist die rechnerische Weiterentwicklung der Taylor-Reihen-Betrachtung aus Satz 8.20.  $D_i$  sind die Taylor-Koeffizienten von  $\tau_h(x_j + th)$ . Deshalb muss für (8.81) die Taylor-Reihe um  $x_j + t h$  entwickelt werden. Den ausführlichen Beweis findet man wie den des nächsten Lemmas in [Lam 91].  $\square$

**Lemma 8.22.** Der lokale Diskretisierungsfehler eines Mehrschrittverfahrens mit der Fehlerkonstante  $C_{p+1}$  nach (8.80) ist

$$\tau_h(x_{j+m}) = C_{p+1} h^p y^{(p+1)}(x_j) + O(h^{p+1}). \tag{8.82}$$

Die Näherungswerte  $u_j$  einer linearen  $m$ -Schrittmethode (8.73) erfüllen für das Modellproblem (8.21) die lineare, homogene *Differenzengleichung*  $m$ -ter Ordnung

$$\sum_{k=0}^m (a_k - h\lambda b_k) u_{j+k} = 0. \quad (8.83)$$

Für  $m = 3$  lautet sie

$$(a_0 - h\lambda b_0)u_j + (a_1 - h\lambda b_1)u_{j+1} + (a_2 - h\lambda b_2)u_{j+2} + (a_3 - h\lambda b_3)u_{j+3} = 0. \quad (8.84)$$

Für eine feste Schrittweite  $h$  sind ihre Koeffizienten konstant. Der folgenden Betrachtung legen wir die Differenzengleichung 3-ter Ordnung (8.84) zu Grunde. Ihre allgemeine Lösung bestimmt man mit dem Potenzansatz  $u_{j+k} = \zeta^k$ ,  $\zeta \neq 0$ . Nach seiner Substitution in (8.84) erhalten wir für  $\zeta$  die algebraische Gleichung dritten Grades

$$(a_0 - h\lambda b_0) + (a_1 - h\lambda b_1)\zeta + (a_2 - h\lambda b_2)\zeta^2 + (a_3 - h\lambda b_3)\zeta^3 = 0, \quad (8.85)$$

welche sich mit den beiden charakteristischen Polynomen (8.77) in der allgemeinen Form

$$\phi(\zeta) := \varrho(\zeta) - h\lambda\sigma(\zeta) = 0 \quad (8.86)$$

schreiben lässt. (8.85) und (8.86) bezeichnet man als die charakteristische Gleichung der entsprechenden Differenzengleichung. (8.85) besitzt drei Lösungen  $\zeta_1, \zeta_2, \zeta_3$ , die wir zunächst paarweise verschieden voraussetzen wollen. Dann bilden  $\zeta_1^k, \zeta_2^k$  und  $\zeta_3^k$  ein Fundamentalsystem von unabhängigen Lösungen der Differenzengleichung (8.84). Auf Grund der Linearität von (8.84) lautet ihre allgemeine Lösung

$$u_{j+k} = c_1 \zeta_1^k + c_2 \zeta_2^k + c_3 \zeta_3^k, \quad c_1, c_2, c_3 \text{ beliebig.} \quad (8.87)$$

Die Konstanten ergeben sich aus den drei Startwerten  $u_0, u_1, u_2$ , die ja für die Anwendung einer 3-Schrittmethode bekannt sein müssen. Die Bedingungsgleichungen lauten

$$\begin{aligned} c_1 &+ c_2 + c_3 = u_0 \\ \zeta_1 c_1 &+ \zeta_2 c_2 + \zeta_3 c_3 = u_1 \\ \zeta_1^2 c_1 &+ \zeta_2^2 c_2 + \zeta_3^2 c_3 = u_2 \end{aligned} \quad (8.88)$$

Unter der getroffenen Annahmen  $\zeta_i \neq \zeta_j$  für  $i \neq j$  ist das lineare Gleichungssystem (8.88) eindeutig lösbar, da seine *Vandermondesche Determinante* ungleich null ist. Mit den so bestimmten Koeffizienten  $c_i$  erhalten wir eine explizite Darstellung für die Näherungswerte  $u_j$  ( $j \geq 3$ ) des Modellproblems (8.21). Speziell kann das qualitative Verhalten der Werte  $u_j$  auf Grund von (8.87) für  $h \rightarrow 0$  und  $j \rightarrow \infty$  diskutiert werden. Für den wichtigen Fall  $\lambda < 0$  in (8.21) ist die Lösung  $y(x)$  exponentiell abklingend. Dies soll auch für die berechneten Werte  $u_j$  zutreffen, ganz besonders für beliebig kleine Schrittweiten  $h > 0$ . Wegen (8.87) ist dies genau dann erfüllt, wenn alle Lösungen  $\zeta_i$  der charakteristischen Gleichung (8.86) betragsmäßig kleiner als Eins sind. Da für  $h \rightarrow 0$  die Lösungen  $\zeta_i$  von  $\phi(\zeta) = 0$  wegen (8.86) in die Nullstellen des ersten charakteristischen Polynoms  $\varrho(\zeta)$  übergehen, dürfen dieselben folglich nicht außerhalb des abgeschlossenen Einheitskreises der komplexen Ebene liegen. Diese notwendige Bedingung für die Brauchbarkeit eines Mehrschrittverfahrens zur Integration des Modellproblems gilt bis jetzt nur unter der Voraussetzung paarweise verschiedener Lösungen  $\zeta_i$  von (8.86).

Falls beispielsweise für ein Sechs-Schrittverfahren die Lösungen  $\zeta_1, \zeta_2 = \zeta_3, \zeta_4 = \zeta_5 = \zeta_6$  sind, dann hat die allgemeine Lösung der Differenzengleichung die Form [Hen 62, Hen 72]

$$u_{j+k} = c_1 \zeta_1^k + c_2 \zeta_2^k + c_3 k \zeta_2^k + c_4 \zeta_4^k + c_5 k \zeta_4^k + c_6 k(k-1) \zeta_4^k. \quad (8.89)$$

Aus dieser Darstellung schließt man wieder, dass  $u_{j+k}$  genau dann mit zunehmendem  $k$  abklingt, wenn alle Lösungen  $\zeta_i$  betragsmäßig kleiner Eins sind. Diese Feststellungen führen zu folgender

**Definition 8.23.** 1. Ein lineares Mehrschrittverfahren erfüllt die *Wurzelbedingung (P)*, wenn für die Nullstellen  $\zeta_j, j = 1, \dots, m$ , des 1. erzeugenden Polynoms  $\varrho(\zeta)$

$$|\zeta_j| \leq 1 \quad \text{für alle } j = 1, \dots, m \quad (8.90)$$

gilt und die *wesentlichen Wurzeln*  $\zeta_j$  mit  $|\zeta_j| = 1$  einfach sind.

2. Ein lineares Mehrschrittverfahren heißt *Lipschitz-stabil* (L-stabil, asymptotisch stabil, Null-stabil, Dahlquist-stabil), falls (P) erfüllt ist.

Einige Tatsachen wollen wir ohne Beweis anmerken:

1. Es kann wie bei den Einschrittverfahren gezeigt werden [Hen 82, Gri 72], dass aus Konsistenz und L-Stabilität Konvergenz folgt.

2. Es ist auch wieder die Konvergenzordnung gleich der Konsistenzordnung.

3. Für ein konsistentes, stabiles  $m$ -Schrittverfahren ( $A_h$ ) mit der Konsistenzordnung  $p$  gilt

$$p \leq m+1, \quad \text{falls } m \text{ ungerade,}$$

$$p \leq m+2, \quad \text{falls } m \text{ gerade. Dann gilt für } i = 1, \dots, m :$$

$$|\zeta_i| = 1, \quad \text{falls } p = m+2. \quad (8.91)$$

### Beispiel 8.10. Konstruktion eines Vierschritt-Verfahrens der Ordnung 6

Wegen (8.91) muss  $\varrho(\zeta)$  vier Nullstellen mit  $|\zeta_i| = 1$  haben. Als Ansatz versuchen wir

$$\zeta_1 = 1, \zeta_2 = -1, \zeta_3 = e^{i\varphi}, \zeta_4 = e^{-i\varphi}, \quad 0 < \varphi < \pi. \quad (8.92)$$

$$\Rightarrow \varrho(\zeta) = (\zeta - 1)(\zeta + 1)(\zeta - e^{i\varphi})(\zeta - e^{-i\varphi}) \text{ wegen } a_4 = 1$$

$$\Rightarrow \varrho(\zeta) = \zeta^4 - 2 \cos \varphi \zeta^3 + 2 \cos \varphi \zeta - 1.$$

Setze  $\mu := \cos \varphi$ . Dann sind  $a_4 = 1$  (wegen Def. 8.18 (iii)),  $a_3 = -2\mu$ ,  $a_2 = 0$ ,  $a_1 = 2\mu$ ,  $a_0 = -1$ .  $p = 6$  erfüllen wir mit Hilfe von (8.81) durch Lösung von  $D_i(t) = 0$  mit  $t = 2$ . Damit nutzen wir die vorliegende Symmetrie aus.

$$D_0(2) = a_0 + a_1 + a_2 + a_3 + a_4 = 0 \quad (\text{ist erfüllt, s.o.}),$$

$$D_1(2) = -2a_0 - a_1 + a_3 + 2a_4 - (b_0 + b_1 + b_2 + b_3 + b_4) \stackrel{!}{=} 0,$$

$$D_2(2) = \frac{1}{2} \underbrace{(4a_0 + a_1 + a_3 + 4a_4)}_{= 0 \text{ für alle geraden } i} - (-2b_0 - b_1 + b_3 + 2b_4) \stackrel{!}{=} 0,$$

$$D_3(2) = \frac{1}{6}(-8a_0 - a_1 + a_3 + 8a_4) - \frac{1}{2!}(4b_0 + b_1 + b_3 + 4b_4),$$

$$= \frac{8}{3} - \frac{2}{3}\mu - \frac{1}{2}(4b_0 + b_1 + b_3 + 4b_4) \stackrel{!}{=} 0,$$

$$D_4(2) = -\frac{1}{3!}(-8b_0 - b_1 + b_3 + 8b_4) \stackrel{!}{=} 0,$$

$$\begin{aligned} D_5(2) &= \frac{2}{5!}(2^5 - 2\mu) - \frac{1}{4!}(16b_0 + b_1 + b_3 + 16b_4) \stackrel{!}{=} 0, \\ D_6(2) &= -\frac{1}{5!}(-32b_0 - b_1 + b_3 + 32b_4) \stackrel{!}{=} 0. \end{aligned}$$

Die Gleichungen reduzieren sich, wenn man zur Symmetrie-Ausnutzung fordert

$$b_0 = b_4, \quad b_1 = b_3 \quad \Rightarrow \quad D_2 = D_4 = D_6 = 0.$$

Dann ergeben sich aus  $D_3 \stackrel{!}{=} 0$  und  $D_5 \stackrel{!}{=} 0$

$$4b_0 + b_1 = \frac{1}{3}(8 - 2\mu), \quad 16b_0 + b_1 = \frac{2}{5!} \cdot \frac{4!}{2}(32 - 2\mu).$$

Daraus folgt

$$b_0 = \frac{1}{45}(14 + \mu) = b_4, \quad b_1 = \frac{1}{45}(64 - 34\mu) = b_3.$$

Damit folgt aus  $D_1 \stackrel{!}{=} 0$

$$b_2 = -\frac{1}{45}(28 + 2\mu + 128 - 68\mu) + 4 - 4\mu = \frac{1}{15}(8 - 38\mu).$$

Damit sind für unseren Ansatz (8.92) alle Bedingungen für ein Verfahren der Ordnung  $p = 6$  erfüllt, und der Parameter  $\varphi$  ist sogar noch frei wählbar. Die einfachste Wahl ist  $\varphi = \pi/2$  mit  $\cos \varphi = 0 = \mu$ . Das ergibt die Methode

$$u_{j+4} = u_j + \frac{h}{45}(14f_j + 64f_{j+1} + 24f_{j+2} + 64f_{j+3} + 14f_{j+4}) \quad (8.93)$$

mit der Fehlerkonstante

$$\begin{aligned} C_{p+1} = D_{p+1}(2) &= \frac{1}{7!}(-2^7 a_0 - a_1 + a_3 + 2^7 a_4) - \frac{1}{6!}(2^6 b_0 + b_1 + b_3 + 2^6 b_4) \\ &= \frac{2 \cdot 2^7}{7!} - \frac{2}{6!}(2^6 \cdot \frac{14}{45} + \frac{64}{45}) = -\frac{16}{1890}. \end{aligned} \quad (8.94)$$

Dieses Verfahren erhalten wir auch, wenn wir das Integral in

$$y(x_{j+4}) = y(x_j) + \int_{x_j}^{x_{j+4}} f(x, y(x)) dx \quad (8.95)$$

numerisch integrieren, indem wir  $f$  durch das Interpolationspolynom durch die Punkte  $(x_j, f_j)$ ,  $(x_{j+1}, f_{j+1})$ ,  $(x_{j+2}, f_{j+2})$ ,  $(x_{j+3}, f_{j+3})$ ,  $(x_{j+4}, f_{j+4})$  ersetzen und dieses exakt integrieren.  $\triangle$

**Beispiel 8.11.** Das explizite lineare Zweischrittverfahren

$$u_{j+2} = (1 + \alpha)u_{j+1} - \alpha u_j + \frac{h}{2}[(3 - \alpha)f_{j+1} - (1 + \alpha)f_j] \quad (8.96)$$

ist nach Satz 8.20 konsistent für alle  $\alpha \in \mathbb{R}$ , denn es sind  $\varrho(\zeta) = \zeta^2 - (1 + \alpha)\zeta + \alpha$ , also  $\varrho(1) = 0$ ,  $\sigma(\zeta) = \frac{1}{2}((3 - \alpha)\zeta - 1 - \alpha)$  und damit  $\varrho'(1) = 1 - \alpha = \sigma(1)$ . Das Anfangswertproblem

$$y' = 4x\sqrt{y} \text{ in } [a, b] = [0, 2], \quad y(0) = 1,$$

hat die exakte Lösung  $y(x) = (1 + x^2)^2$ . Wir nehmen als Startwerte  $u_0 = y_0 = 1$  und  $u_1 = y(h) = (1 + h^2)^2$ . Wir betrachten (aus [Lam 91]) numerische Lösungen für  $\alpha = 0$  und  $\alpha = -5$  mit  $h = 0.1$  in Tab. 8.2 mit auf sechs wesentliche Stellen gerundeten Zahlenwerten. Die Lösung für  $\alpha = -5$  ist auch für kleinere Werte von  $h$  völlig falsch trotz der exakten Startwerte. Das liegt an der Verletzung der Wurzelbedingung (8.90) für  $\alpha = -5$ . Die Konsistenz allein ist eben nicht hinreichend für die Konvergenz.

$\triangle$

Tab. 8.2 Ergebnisse der Methode (8.96).

$x$	$y(x)$	$u_j(\alpha = 0)$	$u_j(\alpha = -5)$
0	1.00000	1.00000	1.00000
0.1	1.02010	1.02010	1.02010
0.2	1.08160	1.08070	1.08120
0.3	1.18810	1.18524	1.18924
0.4	1.34560	1.33963	1.38887
0.5	1.56250	1.55209	1.59299
$\vdots$	$\vdots$	$\vdots$	$\vdots$
1.0	4.00000	3.94069	-68.6398
1.1	4.88410	4.80822	+367.263
$\vdots$	$\vdots$	$\vdots$	$\vdots$
2.0	25.0000	24.6325	$-6.96 \cdot 10^8$

**Beispiel 8.12.** Wir wollen jetzt eine implizite Dreischrittmethode maximaler Ordnung mit  $b_0 = b_1 = b_2 = 0$  konstruieren. Die vier verbleibenden Parameter  $a_0, a_1, a_2$  und  $b_3$  werden erlauben, eine Methode der Ordnung drei zu konstruieren, falls sie die Gleichungen

$$\begin{aligned} C_0 &= a_0 + a_1 + a_2 + 1 = 0 \\ C_1 &= a_1 + 2a_2 - b_3 + 3 = 0 \\ 2C_2 &= a_1 + 4a_2 - 6b_3 + 9 = 0 \\ 6C_3 &= a_1 + 8a_2 - 27b_3 + 27 = 0 \end{aligned}$$

erfüllen. Als Lösung ergeben sich  $a_0 = -2/11$ ,  $a_1 = 9/11$ ,  $a_2 = -18/11$ ,  $b_3 = 6/11$ , und die resultierende implizite Dreischrittmethode lautet

$$u_{j+3} = \frac{18}{11}u_{j+2} - \frac{9}{11}u_{j+1} + \frac{2}{11}u_j + \frac{6}{11}hf(x_{j+3}, u_{j+3}).$$

Sie wird nach Multiplikation mit 11/6 üblicherweise wie folgt geschrieben.

$$\frac{11}{6}u_{j+3} - 3u_{j+2} + \frac{3}{2}u_{j+1} - \frac{1}{3}u_j = hf(x_{j+3}, u_{j+3})$$

(8.97)

Das Verfahren (8.97) ist ein Repräsentant aus der Klasse der *Rückwärtsdifferenzationsmethoden*, die kurz BDF-Methoden (backward differentiation formulae) genannt werden [Gea 71, Hai 93, Hen 62]. Der Name erklärt sich so, dass die linke Seite von (8.97) das  $h$ -fache einer Formel der numerischen Differenziation für die erste Ableitung von  $y(x)$  an der Stelle  $x_{j+3}$  ist (vgl. Abschnitt 3.1.6). Die Differenzialgleichung  $y' = f(x, y)$  wird unter Verwendung einer Differenzationsformel an der Stelle  $x_{j+3}$  mit Hilfe von zurückliegenden Funktionswerten approximiert. Diese speziellen impliziten  $m$ -Schrittverfahren können auch direkt auf diese Weise hergeleitet werden, sie sind allgemein durch  $b_0 = b_1 = \dots = b_{m-1} = 0$  charakterisiert und besitzen die Ordnung  $p = m$ . Die BDF-Methoden der Ordnungen  $p = m = 1$  (das ist gerade das implizite Euler-Verfahren (8.14)) bis  $p = m = 6$  sind besonders zur Lösung mild steifer Differenzialgleichungssysteme geeignet (Abschnitt 8.4.4 und [Hai 96]). BDF-Methoden mit  $m > 6$  sind unbrauchbar, siehe auch Abschnitt 8.4.3.  $\triangle$

Für die Verfahren vom Adams-Typ gilt, dass ein  $m$ -stufiges implizites Verfahren trotz gleicher Ordnung besser ist als ein  $(m+1)$ -stufiges explizites Verfahren, da es eine kleinere Fehlerkonstante (Tab. 8.3) und größere Stabilität (Abschnitt 8.4.3) besitzt.

Tab. 8.3 Ordnungen und Fehlerkonstanten der Adams-Methoden.

	Adams-Bashforth (explizit)					Adams-Moulton (implizit)			
$m$	2	3	4	5	6	2	3	4	5
$p$	2	3	4	5	6	3	4	5	6
$C_{p+1}$	$\frac{5}{12}$	$\frac{3}{8}$	$\frac{251}{720}$	$\frac{95}{288}$	$\frac{19087}{60480}$	$-\frac{1}{24}$	$-\frac{19}{720}$	$-\frac{3}{160}$	$-\frac{863}{60480}$

### Start- oder Anlaufrechnung

Um ein  $m$ -Schrittverfahren überhaupt anwenden zu können, sind neben dem Anfangswert  $y_0 = u_0$  noch  $(m-1)$  weitere Startwerte  $u_1, u_2, \dots, u_{m-1}$  nötig. Die einfachste Möglichkeit zu ihrer Berechnung ist die Verwendung eines Einschrittverfahrens derselben Ordnung.

Eine bessere Möglichkeit ergibt sich, wenn vom Startpunkt  $(x_0, y_0)$  ausgehend die Möglichkeit besteht, mit negativer Schrittweite zu rechnen. Es können dann mit den Schrittweiten  $h$  und  $-h$   $m$  Startwerte symmetrisch zu  $x_0$

$$\dots, u_{-2}, u_{-1}, u_0 = y_0, u_1, u_2, \dots$$

berechnet werden, und das Mehrschrittverfahren startet ‘weiter links’ im Intervall  $[a, b]$ .

Eine noch bessere Möglichkeit ist der sukzessive Aufbau der Startwerte mit Mehrschrittverfahren wachsender Ordnung [Gri 72].

### Schrittweitensteuerung

Wie bei Einschrittverfahren, siehe Abschnitt 8.2.5, sollte auch bei Mehrschrittverfahren die Schrittweite variabel gesteuert werden, am einfachsten wieder, indem parallel mit zwei Verfahren unterschiedlicher Ordnung gerechnet wird und dann die Differenz als Schätzer wie (8.67) gewählt wird. Problematischer als bei Einschrittverfahren ist der Wechsel der Schrittweite. Ein Einschrittverfahren benötigt nur den letzten  $u$ -Wert, während ein  $m$ -Schrittverfahren  $m$  zurück liegende Werte zur Berechnung von  $u_{j+m}$  benötigt.

Bei einer *Schrittweiten-Verdoppelung* müssen  $m-2$  zusätzliche  $u$ - und  $f$ -Werte gespeichert bleiben. Nach einer Verdoppelung müssen  $m-1$  Schritte ohne erneute Verdoppelung durchgeführt werden.

Bei einer *Schrittweiten-Halbierung* müssen  $\left[\frac{m+1}{2}\right]$  Zwischenwerte berechnet werden, die mindestens die Genauigkeit des Verfahrens besitzen. Zur Berechnung dieser Zwischenwerte kommen zwei Methoden in Betracht:

1. Es wird das Verfahren der Anlaufrechnung verwendet.

2. Die Zwischenwerte werden durch Interpolation berechnet, wobei sehr genaue Interpolationsmethoden verwendet werden müssen, um die Ordnung des Mehrschrittverfahrens zu erhalten. Speziell für Vierschritt-Verfahren besitzen die Interpolationsformeln [Kei 56]

$$\begin{aligned} u_{j+m-\frac{3}{2}} &= \frac{1}{256}(80u_{j+3} + 135u_{j+2} + 40u_{j+1} + u_j) \\ &\quad + \frac{h}{256}(-15f_{j+3} + 90f_{j+2} + 15f_{j+1}), \\ u_{j+m-\frac{5}{2}} &= \frac{1}{256}(12u_{j+3} + 135u_{j+2} + 108u_{j+1} + u_j) \\ &\quad + \frac{h}{256}(-3f_{j+3} - 54f_{j+2} + 27f_{j+1}) \end{aligned} \tag{8.98}$$

die Genauigkeit  $O(h^7)$ .

Es gibt noch zwei wesentlich verfeinerte Methoden zur Genauigkeitssteuerung.

1. Verfahren mit *variabler* Schrittweite nach Krogh, bei denen für jeden Schritt die Koeffizienten des Verfahrens neu berechnet werden [Hai 93].
2. Verfahren variabler Ordnung und Schrittweite wie die BDF-Verfahren, bei denen beide Größen kombiniert gesteuert werden.

## 8.4 Stabilität

Bei der Wahl eines bestimmten Verfahrens zur genäherten Lösung eines Differenzialgleichungssystems erster Ordnung sind die Eigenschaften der gegebenen Differenzialgleichungen und der resultierenden Lösungsfunktionen zu berücksichtigen. Tut man dies nicht, können die berechneten Näherungslösungen mit den exakten Lösungsfunktionen sehr wenig zu tun haben oder schlicht sinnlos sein. Es geht im Folgenden um die Analyse von Instabilitäten, die bei unsachgemäßer Anwendung von Verfahren auftreten können, und die es zu vermeiden gilt [Dah 85, Rut 52].

### 8.4.1 Inhärente Instabilität

Wir untersuchen die Abhängigkeit der Lösung  $y(x)$  vom Anfangswert  $y(x_0) = y_0$  anhand einer Klasse von Differenzialgleichungen, deren Lösungsmenge geschlossen angegeben werden kann. Die Anfangswertaufgabe lautet

$$y'(x) = \lambda\{y(x) - F(x)\} + F'(x), \quad y(x_0) = y_0, \tag{8.99}$$

wo  $F(x)$  mindestens einmal stetig differenzierbar sei. Da  $y_{\text{hom}}(x) = Ce^{\lambda x}$  die allgemeine Lösung der homogenen Differenzialgleichung und  $y_{\text{part}}(x) = F(x)$  eine partikuläre Lösung der inhomogenen Differenzialgleichung ist, lautet die Lösung von (8.99)

$$y(x) = \{y_0 - F(x_0)\}e^{\lambda(x-x_0)} + F(x). \tag{8.100}$$

Für den speziellen Anfangswert  $y_0 = F(x_0)$  ist  $y(x) = F(x)$ , und der Anteil der Exponentialfunktion ist nicht vorhanden. Für den leicht geänderten Anfangswert  $\hat{y}_0 = F(x_0) + \varepsilon$ , wo

$\varepsilon$  eine betragsmäßig kleine Größe darstellt, lautet die Lösungsfunktion

$$\hat{y}(x) = \varepsilon e^{\lambda(x-x_0)} + F(x). \quad (8.101)$$

Ist  $\lambda \in \mathbb{R}, \lambda > 0$ , nimmt der erste Summand in  $\hat{y}(x)$  mit zunehmendem  $x$  exponentiell zu, so dass sich die benachbarte Lösung  $\hat{y}(x)$  von  $y(x)$  mit zunehmendem  $x$  immer mehr entfernt. Es besteht somit eine starke Empfindlichkeit der Lösung auf kleine Änderungen  $\varepsilon$  des Anfangswertes. Die Aufgabenstellung kann als schlecht konditioniert bezeichnet werden. Da die Empfindlichkeit der Lösungsfunktion durch die gegebene Anfangswertaufgabe bedingt ist, bezeichnet man diese Phänomen als *inhärente Instabilität*. Sie ist unabhängig von der verwendeten Methode zur genäherten Lösung von (8.99). Sie äußert sich so, dass sich die berechneten Näherungswerte  $u_n$  entsprechend zu (8.101) von den exakten Werten  $y(x_n) = F(x_n)$  in exponentieller Weise entfernen. Wenn überhaupt, dann ist die inhärente Instabilität nur so in den Griff zu bekommen, dass mit Methoden hoher Fehlerordnung und mit hoher Rechengenauigkeit gearbeitet wird, um sowohl die Diskretisierungs- als auch die Rundungsfehler genügend klein zu halten.

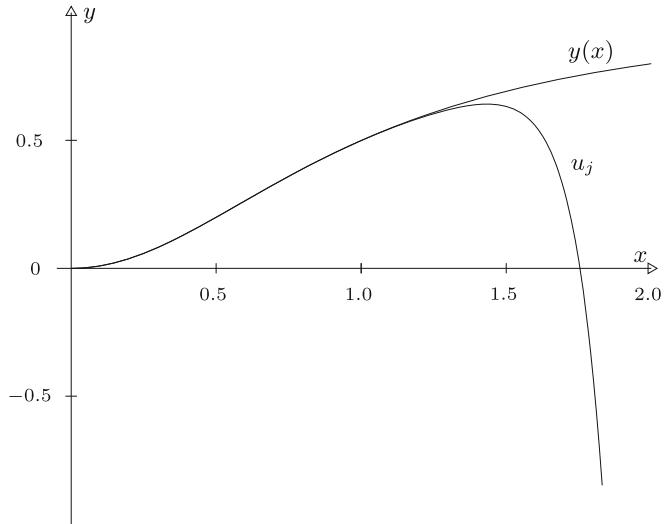


Abb. 8.7 Zur inhärenten Instabilität.

**Beispiel 8.13.** Wir betrachten die Anfangswertaufgabe

$$y'(x) = 10 \left\{ y(x) - \frac{x^2}{1+x^2} \right\} + \frac{2x}{(1+x^2)^2}, \quad y(0) = y_0 = 0,$$

vom Typus (8.99) mit der Lösung  $y(x) = x^2/(1+x^2)$ . Mit dem klassischen Runge-Kutta-Verfahren vierter Ordnung (8.44) ergeben sich bei einer Schrittweite  $h = 0.01$  die Näherungen  $u_j$ , welche zusammen mit  $y(x)$  in Abb. 8.7 dargestellt sind. Die inhärente Instabilität wird sehr deutlich.  $\triangle$

### 8.4.2 Absolute Stabilität bei Einschrittverfahren

Während wir im letzten Abschnitt gesehen haben, dass ein Differenzialgleichungsproblem instabil sein kann mit den entsprechenden Auswirkungen auf die numerische Lösung, wollen wir uns jetzt der so genannten *Stabilität für endliche h* zuwenden. Dabei geht es um die Vermeidung von numerischer Instabilität, die wir schon in den einleitenden Beispielen 8.1 und 8.2 kennen gelernt haben. Um die Stabilität für endliche  $h$  zu charakterisieren, sind zahlreiche Begriffe und Definitionen eingeführt worden. Wir wollen uns auf den wichtigsten Begriff, den der *absoluten Stabilität*, beschränken.

Die Betrachtungen werden wieder am linearen Modellproblem

$$y'(x) = \lambda y(x), \quad y(0) = 1, \quad \lambda \in \mathbb{R} \text{ oder } \lambda \in \mathbb{C} \quad (8.102)$$

mit der bekannten Lösung  $y(x) = e^{\lambda x}$  durchgeführt. Als typischen Vertreter der Einschrittmethoden betrachten wir das klassische Runge-Kutta-Verfahren (8.44) vierter Ordnung und bestimmen seine Wirkungsweise für die Anfangswertaufgabe (8.102). Wir erhalten sukzessive

$$\begin{aligned} k_1 &= \lambda u_j \\ k_2 &= \lambda \left( u_j + \frac{1}{2} h k_1 \right) = \left( \lambda + \frac{1}{2} h \lambda^2 \right) u_j \\ k_3 &= \lambda \left( u_j + \frac{1}{2} h k_2 \right) = \left( \lambda + \frac{1}{2} h \lambda^2 + \frac{1}{4} h^2 \lambda^3 \right) u_j \\ k_4 &= \lambda \left( u_j + h k_3 \right) = \left( \lambda + h \lambda^2 + \frac{1}{2} h^2 \lambda^3 + \frac{1}{4} h^3 \lambda^4 \right) u_j \\ u_{j+1} &= u_j + \frac{h}{6} \{k_1 + 2k_2 + 2k_3 + k_4\} \\ &= \left( 1 + h \lambda + \frac{1}{2} h^2 \lambda^2 + \frac{1}{6} h^3 \lambda^3 + \frac{1}{24} h^4 \lambda^4 \right) u_j \end{aligned} \quad (8.103)$$

Nach (8.103) entsteht  $u_{j+1}$  aus  $u_j$  durch Multiplikation mit dem Faktor

$$F(h\lambda) := 1 + h\lambda + \frac{1}{2} h^2 \lambda^2 + \frac{1}{6} h^3 \lambda^3 + \frac{1}{24} h^4 \lambda^4, \quad (8.104)$$

der vom Produkt  $h\lambda$  abhängt und offensichtlich gleich dem Beginn der Taylor-Reihe für  $e^{h\lambda}$  ist mit einem Fehler  $O(h^5)$ . Für die Lösung  $y(x)$  gilt ja  $y(x_{j+1}) = y(x_j + h) = e^{h\lambda} y(x_j)$ , und somit steht die letzte Feststellung im Einklang damit, dass der lokale Diskretisierungsfehler der klassischen Runge-Kutta-Methode von der Ordnung  $O(h^4)$  ist. Der Multiplikator  $F(h\lambda)$  (8.104) stellt für betragskleine Werte  $h\lambda$  sicher eine gute Approximation für  $e^{h\lambda}$  dar. Für den wichtigen Fall abklingender Lösung  $\lambda < 0$  sollen die Näherungswerte  $u_j$  mit  $y(x_j)$  abklingen. Das ist aber nur der Fall, wenn  $|F(h\lambda)| < 1$  ist. Für das Polynom vierten Grades  $F(h\lambda)$  gilt  $\lim_{h\lambda \rightarrow -\infty} F(h\lambda) = +\infty$ . Deshalb kann  $|F(h\lambda)| < 1$  nicht für alle negativen Werte von  $h\lambda$  erfüllt sein.

Systeme von Differenzialgleichungen besitzen oft auch oszillierende, exponentiell abklingende Komponenten, welche komplexen Werten von  $\lambda$  entsprechen. Für die jetzt komplexewertige Lösung  $y(x)$  gilt wiederum  $y(x_{j+1}) = e^{h\lambda} y(x_j)$ . Der komplexe Faktor  $e^{h\lambda}$  ist im

interessierenden Fall mit  $\operatorname{Re}(\lambda) < 0$  betragsmäßig kleiner Eins. Damit die berechneten Näherungswerte  $u_j$  wenigstens wie  $y(x_j)$  dem Betrag nach abnehmen, muss wiederum die notwendige und hinreichende Bedingung  $|F(h\lambda)| < 1$  erfüllt sein.

Analoge Bedingungen gelten für alle expliziten Runge-Kutta-Verfahren. Eine einfache Rechnung zeigt, dass bei Anwendung eines expliziten  $p$ -stufigen Runge-Kutta-Verfahrens der Fehlerordnung  $p \leq 4$  auf das Modellproblem (8.21) der Faktor  $F(h\lambda)$  stets gleich den ersten  $(p+1)$  Termen der Taylor-Reihe von  $e^{h\lambda}$  ist. Runge-Kutta-Verfahren höherer Ordnung  $p$  erfordern aber  $m > p$  Stufen, so dass  $F(h\lambda)$  ein Polynom vom Grad  $m$  wird, welches in den ersten  $(p+1)$  Termen mit der Taylor-Reihe von  $e^{h\lambda}$  übereinstimmt. Die Koeffizienten bis zur Potenz  $m$  sind vom Verfahren abhängig. Die notwendige und hinreichende Bedingung erfasst man mit der

**Definition 8.24.** 1. Für ein Einschrittverfahren, welches für das Modellproblem (8.21) auf  $u_{j+1} = F(h\lambda)u_j$  führt, heißt die Menge

$$B := \{\mu \in \mathbb{C} \mid |F(\mu)| < 1\} \quad (8.105)$$

mit  $\mu = h\lambda$  Gebiet der absoluten Stabilität.

2. Ein Verfahren, für das das Gebiet  $B$  der absoluten Stabilität die gesamte linke Halbebene umfasst, heißt absolut stabil.

Die Schrittweite  $h$  ist so zu wählen, dass für  $\operatorname{Re}(\lambda) < 0$  stets  $h\lambda \in B$  gilt. Andernfalls liefert das Verfahren unsinnige Ergebnisse, es arbeitet instabil. Diese *Stabilitätsbedingung* ist speziell bei der Integration von Differenzialgleichungssystemen zu beachten, denn die Schrittweite  $h$  ist so zu bemessen, dass für alle  $\lambda_j$  mit  $\operatorname{Re}(\lambda_j) < 0$  (die sog. Abklingkonstanten) die Bedingungen  $h\lambda_j \in B$  erfüllt sind.

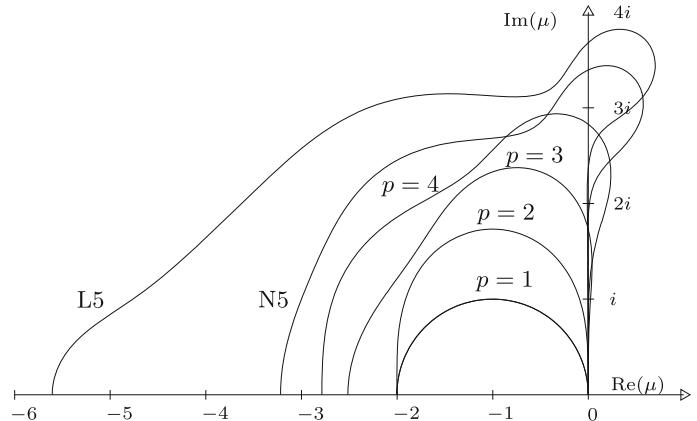


Abb. 8.8 Stabilitätsgebiete für explizite Runge-Kutta-Methoden.

In Abb. 8.8 sind die Berandungen der Gebiete der absoluten Stabilität für explizite Runge-Kutta-Verfahren der Ordnungen  $p = 1, 2, 3, 4, 5$  für die obere Hälfte der komplexen Ebene dargestellt, denn die Gebiete sind symmetrisch bezüglich der reellen Achse. Im Fall  $p = 5$  ist

die spezielle Runge-Kutta-Methode von *Nyström* [Gri 72] zu Grunde gelegt mit der Eigenschaft, dass  $F(h\lambda)$  mit den ersten sechs Termen der Taylor-Reihe von  $e^{h\lambda}$  übereinstimmt. Die Randkurve wurde mit N5 beschriftet.

Die Stabilitätsgebiete werden mit zunehmender Ordnung größer. Das Stabilitätsgebiet der Methode von Euler ist das Innere des Kreises vom Radius Eins mit Mittelpunkt  $\mu = -1$ . Ein Maß für die Größe des Stabilitätsgebietes ist das *Stabilitätsintervall* für reelle negative Werte  $\mu$ . Tab. 8.4 enthält die Angaben über die Stabilitätsintervalle der Methoden der Ordnungen  $p = 1, 2, 3, 4, 5$ , wobei im letzten Fall wieder die Methode von Nyström angenommen ist.

Tab. 8.4 Stabilitätsintervalle von Runge-Kutta-Verfahren.

$p =$	1	2	3	4	5
Intervall	$(-2.0, 0)$	$(-2.0, 0)$	$(-2.51, 0)$	$(-2.78, 0)$	$(-3.21, 0)$

Lawson [Law 66] hat ein sechsstufiges Runge-Kutta-Verfahren fünfter Ordnung mit dem besonders großen Stabilitätsintervall  $(-5.6, 0)$  angegeben. Der Rand des zugehörigen Stabilitätsgebiets wurde in Abb. 8.8 mit L5 beschriftet.

Nun betrachten wir die impliziten Einschrittverfahren, zu denen die Trapezmethode und die impliziten Runge-Kutta-Verfahren gehören. Die Trapezmethode (8.16) ergibt für das Modellproblem (8.21) die explizite Rechenvorschrift

$$\begin{aligned} u_{j+1} &= u_j + \frac{h}{2} \{ \lambda u_j + \lambda u_{j+1} \}, \quad \text{also} \\ u_{j+1} &= \frac{1 + h\lambda/2}{1 - h\lambda/2} u_j =: F(h\lambda) u_j. \end{aligned} \tag{8.106}$$

Die Funktion  $F(h\lambda)$  ist jetzt gebrochen rational mit der Eigenschaft

$$|F(\mu)| = \left| \frac{2 + \mu}{2 - \mu} \right| < 1 \text{ für alle } \mu \text{ mit } \operatorname{Re}(\mu) < 0, \tag{8.107}$$

denn der Realteil des Zählers ist für  $\operatorname{Re}(\mu) < 0$  betragsmäßig stets kleiner als der Realteil des Nenners, während die Imaginärteile entgegengesetzt gleich sind. Das Gebiet der absoluten Stabilität der Trapezmethode umfasst somit die ganze linke Halbebene, sie ist also absolut stabil; es ist keine Grenze für die Schrittweite  $h$  zu beachten. Das Gleiche gilt für das implizite Euler-Verfahren (8.14). Auch das einstufige halbimplizite Runge-Kutta-Verfahren (8.46), dessen zugehörige Funktion  $F(h\lambda)$  mit der der Trapezmethode übereinstimmt, ist absolut stabil. Auf die problemgerechte Wahl von  $h$  werden wir im Zusammenhang mit steifen Differenzialgleichungssystemen in Abschnitt 8.4.4 eingehen.

#### 8.4.3 Absolute Stabilität bei Mehrschrittverfahren

Das Problem der *Stabilität für endliche  $h$*  stellt sich auch bei den linearen Mehrschrittverfahren. Nach den Überlegungen von Abschnitt 8.3.2, welche zur Definition der Wurzelbedingung bzw. der L-Stabilität führten, erfüllen die Näherungswerte  $u_{j+k}$  für das Modellproblem

(8.21) einer allgemeinen Mehrschrittmetode (8.73) die lineare Differenzengleichung (8.83)  $m$ -ter Ordnung. Ihre allgemeine Lösung ergibt sich mit den  $m$  Wurzeln  $\zeta_1, \zeta_2, \dots, \zeta_m$  der charakteristischen Gleichung  $\phi(\zeta) = \varrho(\zeta) - h\lambda\sigma(\zeta) = 0$  als

$$u_{j+k} = c_1 \zeta_1^k + c_2 \zeta_2^k + \dots + c_m \zeta_m^k, \quad (8.108)$$

wobei die  $\zeta_i$  vereinfachend als paarweise verschieden angenommen sind. Die allgemeine Lösung (8.108) klingt im allein interessierenden Fall  $\operatorname{Re}(\lambda) < 0$  genau dann ab, wenn alle Wurzeln  $\zeta_i$  betragsmäßig kleiner als Eins sind. Dies gilt auch dann, wenn mehrfache  $\zeta_i$  vorkommen, vgl. (8.89).

**Definition 8.25.** 1. Zu einer linearen Mehrschrittmetode (8.73) heißt die Menge der komplexen Werte  $\mu = h\lambda$ , für welche die charakteristische Gleichung  $\phi(\zeta) = \varrho(\zeta) - \mu\sigma(\zeta) = 0$  nur Lösungen  $\zeta_j \in \mathbb{C}$  im Innern des Einheitskreises besitzt, *Gebiet B der absoluten Stabilität*.

2. Ein Mehrschrittverfahren, für welches das Gebiet  $B$  der absoluten Stabilität die gesamte linke Halbebene umfasst, heißt *absolut stabil*.

Die explizite Vierschrittmetode von *Adams-Basforth* (8.64) hat die charakteristische Gleichung (nach Multiplikation mit 24)

$$24\phi(\zeta) = 24\zeta^4 - (24 + 55\mu)\zeta^3 + 59\mu\zeta^2 - 37\mu\zeta + 9\mu = 0. \quad (8.109)$$

Der Rand des Stabilitätsgebietes für Mehrschrittmethoden kann als geometrischer Ort der Werte  $\mu \in \mathbb{C}$  ermittelt werden, für welchen  $|\zeta| = 1$  ist. Dazu genügt es, mit  $\zeta = e^{i\theta}, 0 \leq \theta \leq 2\pi$  den Einheitskreis zu durchlaufen und die zugehörigen Werte  $\mu$  zu berechnen. Im Fall der charakteristischen Gleichung (8.109) führt dies für  $\mu$  auf die explizite Darstellung

$$\mu = \frac{24\zeta^4 - 24\zeta^3}{55\zeta^3 - 59\zeta^2 + 37\zeta - 9} = \frac{\varrho(\zeta)}{\sigma(\zeta)}, \quad \zeta = e^{i\theta}, \quad 0 \leq \theta \leq 2\pi.$$

Der Rand des Stabilitätsgebietes ist offensichtlich symmetrisch zur reellen Achse. Deshalb ist er für die Vierschrittmetode von Adams-Basforth (8.64) in Abb. 8.9 nur für die obere Halbebene dargestellt und mit AB4 gekennzeichnet. Das Stabilitätsintervall  $(-0.3, 0)$  ist im Vergleich zu den Runge-Kutta-Verfahren vierter Ordnung etwa neunmal kleiner. Die expliziten Adams-Basforth-Methoden besitzen allgemein sehr kleine Gebiete absoluter Stabilität.

Die zur impliziten Vierschrittmetode von *Adams-Moulton* (8.69) gehörende charakteristische Gleichung ist nach Multiplikation mit 720

$$(720 - 251\mu)\zeta^4 - (720 + 646\mu)\zeta^3 + 264\mu\zeta^2 - 106\mu\zeta + 19\mu = 0. \quad (8.110)$$

Der zugehörige Rand des Gebietes der absoluten Stabilität ist in Abb. 8.9 eingezeichnet und mit AM4 beschriftet. Im Vergleich zur expliziten Vierschrittmetode ist das Stabilitätsgebiet bedeutend größer, das Stabilitätsintervall ist  $(-1.836, 0)$ . Obwohl die Methode implizit ist, ist das Stabilitätsgebiet endlich, und das Verfahren ist nicht absolut stabil.

Das Gebiet der absoluten Stabilität der impliziten Dreischrittmetode von Adams-Moulton (8.68) mit der Ordnung  $p = 4$ , dessen Rand in Abb. 8.9 mit AM3 bezeichnet ist, ist noch größer. Das Stabilitätsintervall  $(-3.0, 0)$  ist sogar größer als dasjenige des klassischen Runge-Kutta-Verfahrens gleicher Ordnung.

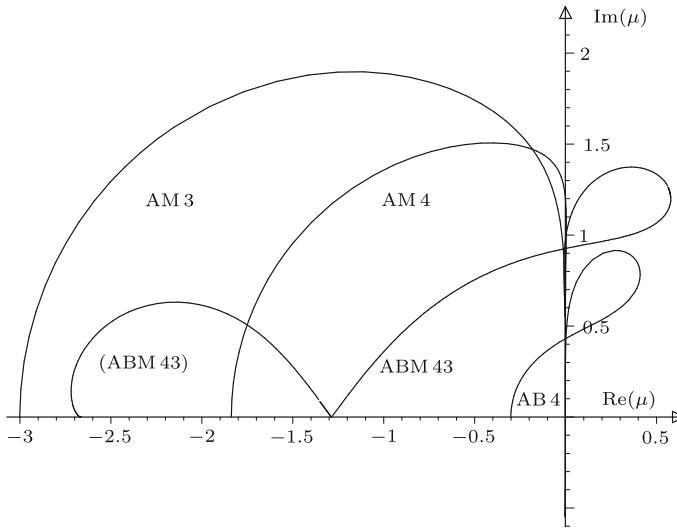


Abb. 8.9 Gebiete absoluter Stabilität für verschiedene Mehrschrittmethoden.

Oft wird die Adams-Moulton-Methode in Verbindung mit der Adams-Bashforth-Methode als Prädiktor-Korrektor-Verfahren verwendet. Das Verfahren (8.72) liefert für das Modellproblem den folgenden Prädiktor- und Korrektowert

$$\begin{aligned} u_{j+4}^{(P)} &= u_{j+3} + \frac{h\lambda}{24} \{55u_{j+3} - 59u_{j+2} + 37u_{j+1} - 9u_j\} \\ u_{j+4} &= u_{j+3} + \frac{h\lambda}{24} \left[ 9 \left\{ u_{j+3} + \frac{h\lambda}{24} (55u_{j+3} - 59u_{j+2} + 37u_{j+1} - 9u_j) \right\} \right. \\ &\quad \left. + 19u_{j+3} - 5u_{j+2} + u_{j+1} \right]. \end{aligned}$$

Durch Addition von  $9u_{j+4}$  und anschließender Subtraktion desselben Wertes in der eckigen Klammer erhalten wir die Differenzengleichung

$$\begin{aligned} u_{j+4} - u_{j+3} - \frac{h\lambda}{24} \{9u_{j+4} + 19u_{j+3} - 5u_{j+2} + u_{j+1}\} \\ + \frac{9h\lambda}{24} \left\{ u_{j+4} - u_{j+3} - \frac{h\lambda}{24} (55u_{j+3} - 59u_{j+2} + 37u_{j+1} - 9u_j) \right\} = 0. \end{aligned} \quad (8.111)$$

In (8.111) erscheinen die Koeffizienten der ersten und zweiten charakteristischen Polynome  $\varrho_{\text{AM}}(\zeta)$ ,  $\sigma_{\text{AM}}(\zeta)$ , bzw.  $\varrho_{\text{AB}}(\zeta)$ ,  $\sigma_{\text{AB}}(\zeta)$  der beiden zu Grunde liegenden Verfahren. Die zu (8.111) gehörende charakteristische Gleichung lautet

$$\phi_{\text{ABM}}(\zeta) = \zeta[\varrho_{\text{AM}}(\zeta) - \mu\sigma_{\text{AM}}(\zeta)] + b_3^{(\text{AM})}\mu\{\varrho_{\text{AB}}(\zeta) - \mu\sigma_{\text{AB}}(\zeta)\} = 0. \quad (8.112)$$

$b_3^{(\text{AM})} = 9/24$  bedeutet den Koeffizienten  $b_3$  der impliziten Dreischrittmetode von Adams-Moulton. Die charakteristische Gleichung (8.112) ist typisch für alle Prädiktor-Korrektor-Methoden. Sie kann wie oben zur Bestimmung des Randes des Gebietes absoluter Stabilität verwendet werden mit dem Unterschied, dass sie für einen Wert  $\zeta = e^{i\theta}$  eine quadratische

Gleichung für  $\mu$  mit zwei Lösungen darstellt. In Abb. 8.9 ist der Rand des Stabilitätsgebiets für das A-B-M-Verfahren (8.72) wiedergegeben. Er ist mit ABM43 bezeichnet um zu verdeutlichen, dass das explizite Vierschrittverfahren von Adams-Bashforth mit dem impliziten Dreischrittverfahren von Adams-Moulton kombiniert ist. Das Stabilitätsgebiet ist gegenüber der Adams-Moulton-Methode (AM3) kleiner, da der Prädiktor-Korrekturwert anstelle der exakten Lösung  $u_{k+1}$  der impliziten Gleichung verwendet wird. Das Stabilitätsintervall des Verfahrens (8.72) ist  $(-1.28, 0)$ .

Die Gebiete absoluter Stabilität einiger Rückwärtsdifferentiationsmethoden wie beispielsweise (8.97) sind für bestimmte Anwendungen recht bedeutungsvoll. Die einfachste Einschrittmethode aus dieser Klasse ist das implizite oder *Rückwärts-Euler-Verfahren*, das ja absolut stabil ist. Die Zweischrittmethode der BDF-Verfahren lautet

$$\boxed{\frac{3}{2}u_{j+2} - 2u_{j+1} + \frac{1}{2}u_j = hf(x_{j+2}, u_{j+2})}. \quad (8.113)$$

Für sie führt das Modellproblem auf die charakteristische Gleichung  $\phi(\zeta) = (3/2 - \mu)\zeta^2 - 2\zeta + 1/2 = 0$ . Der Rand des zugehörigen Gebietes absoluter Stabilität ist gegeben durch

$$\mu = \frac{3\zeta^2 - 4\zeta + 1}{2\zeta^2}, \quad \zeta = e^{i\Theta}, \quad 0 \leq \Theta \leq 2\pi,$$

und liegt in der rechten komplexen Halbebene. Die Randkurve ist in Abb. 8.10 aus Symmetriegründen nur in der oberen Halbebene wiedergegeben und ist mit BDF2 gekennzeichnet. Da weiter die beiden Nullstellen der charakteristischen Gleichung  $\phi(\zeta) = 0$  für alle  $\mu \in \mathbb{R}$  mit  $\mu < 0$  betragsmäßig kleiner als Eins sind, gehört die ganze linke komplexe Halbebene zum Gebiet absoluter Stabilität. Die Zweischritt-BDF-Methode (8.113) ist also auch absolut stabil.

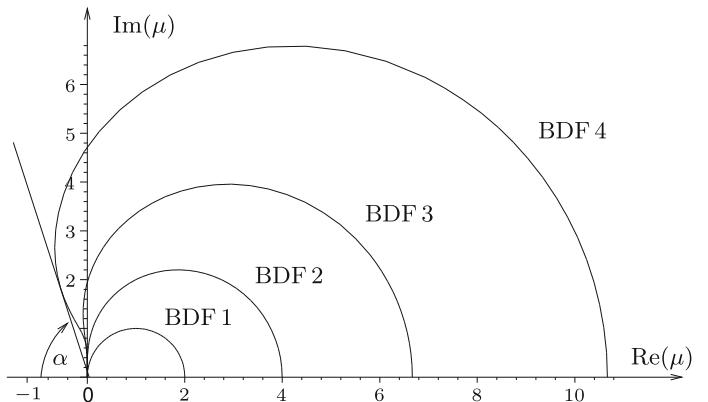


Abb. 8.10 Gebiete absoluter Stabilität von BDF-Methoden.

Für die Dreischritt-BDF-Methode (8.97) wird der Rand des Gebietes absoluter Stabilität gegeben durch

$$\mu = \frac{11\zeta^3 - 18\zeta^2 + 9\zeta - 2}{6\zeta^3}, \quad \zeta = e^{i\Theta}, \quad 0 \leq \Theta \leq 2\pi.$$

Die Randkurve, welche in Abb. 8.10 mit BDF3 bezeichnet ist, verläuft teilweise in der linken komplexen Halbebene. Da aber für  $\mu \in \mathbb{R}$  mit  $\mu < 0$  die drei Nullstellen der charakteristischen Gleichung  $\phi(\zeta) = (11 - 6\mu)\zeta^3 - 18\zeta^2 + 9\zeta - 2 = 0$  betragsmäßig kleiner als Eins sind, umfasst das Gebiet der absoluten Stabilität doch fast die ganz linke komplexe Halbebene. Dieser Situation wird so Rechnung getragen, dass man einen maximalen Winkelbereich mit dem halben Öffnungswinkel  $\alpha > 0$  definiert, dessen Spitze im Nullpunkt liegt, der Teilbereich des Gebietes absoluter Stabilität ist, und bezeichnet ein Mehrschrittverfahren mit dieser Eigenschaft als *A( $\alpha$ )-stabil*. Der Winkel der BDF-Methode (8.97) beträgt etwa  $88^\circ$ , so dass das Verfahren *A( $88^\circ$ )-stabil* ist. Die Vierschritt-BDF-Methode lautet

$$\boxed{\frac{25}{12}u_{j+4} - 4u_{j+3} + 3u_{j+2} - \frac{4}{3}u_{j+1} + \frac{1}{4}u_j = hf(x_{j+4}, u_{j+4})}. \quad (8.114)$$

Die Randkurve des Gebietes absoluter Stabilität ist in Abb. 8.10 eingezeichnet und mit BDF4 gekennzeichnet. Der Winkelbereich besitzt jetzt einen halben Öffnungswinkel von etwa  $72^\circ$ , so dass die Methode (8.114) noch *A( $72^\circ$ )-stabil* ist.

Die BDF-Methoden fünfter und sechster Ordnung besitzen noch kleinere Winkelbereiche innerhalb des Gebietes absoluter Stabilität, und BDF-Methoden noch höherer Ordnung sind nicht mehr L-stabil und somit für den praktischen Einsatz unbrauchbar [Gea 71, Gri 72].

#### 8.4.4 Steife Differenzialgleichungen

Die Lösungsfunktionen von Differenzialgleichungssystemen, welche physikalische, chemische oder biologische Vorgänge beschreiben, haben oft die Eigenschaft, dass sie sich aus stark verschiedenen rasch exponentiell abklingenden Anteilen zusammensetzen. Wird ein Verfahren angewendet, dessen Gebiet absoluter Stabilität nicht die ganze linke komplexe Halbebene umfasst, so ist die Schrittweite  $h$  auf jeden Fall so zu wählen, dass die komplexen Werte  $\mu$  als Produkte von  $h$  und den Abklingkonstanten  $\lambda_j$  dem Gebiet absoluter Stabilität angehören, um damit eine stabile Integration sicherzustellen.

**Beispiel 8.14.** Die Situation und Problematik zeigen wir auf am System von drei linearen und homogenen Differenzialgleichungen

$$\begin{aligned} y'_1 &= -0.5 y_1 + 32.6 y_2 + 35.7 y_3 \\ y'_2 &= \quad -48 \quad y_2 + \quad 9 \quad y_3 \\ y'_3 &= \quad \quad 9 \quad y_2 - 72 \quad y_3 \end{aligned} \quad (8.115)$$

mit den Anfangsbedingungen  $y_1(0) = 4$ ,  $y_2(0) = 13$ ,  $y_3(0) = 1$ . Der Lösungsansatz

$$y_1(x) = a_1 e^{\lambda x}, \quad y_2(x) = a_2 e^{\lambda x}, \quad y_3(x) = a_3 e^{\lambda x}$$

führt nach Substitution in (8.115) auf das Eigenwertproblem

$$\begin{aligned} (-0.5 - \lambda) a_1 + \quad 32.6 a_2 + \quad 35.7 a_3 &= 0 \\ (-48 - \lambda) a_2 + \quad \quad 9 a_3 &= 0 \\ 9 a_2 + (-72 - \lambda) a_3 &= 0. \end{aligned} \quad (8.116)$$

Daraus ist ein nichttriviales Wertetripel  $(a_1, a_2, a_3)^T =: \mathbf{a}$  als Eigenvektor der Koeffizientenmatrix  $\mathbf{A}$  des Differenzialgleichungssystems (8.115) zu bestimmen. Die drei Eigenwerte von (8.116) sind  $\lambda_1 = -0.5$ ,  $\lambda_2 = -45$ ,  $\lambda_3 = -75$ . Zu jedem Eigenwert gehört eine Lösung von (8.115), und die

allgemeine Lösung stellt sich als Linearkombination dieser drei Basislösungen dar. Nach Berücksichtigung der Anfangsbedingung lauten die Lösungsfunktionen

$$\begin{aligned} y_1(x) &= 15e^{-0.5x} - 12e^{-45x} + e^{-75x} \\ y_2(x) &= \quad\quad\quad 12e^{-45x} + e^{-75x} \\ y_3(x) &= \quad\quad\quad 4e^{-45x} - 3e^{-75x} \end{aligned} \quad (8.117)$$

Die stark unterschiedlichen Abklingkonstanten der Lösungskomponenten sind durch die Eigenwerte  $\lambda_j$  gegeben. Zur numerischen Integration von (8.115) soll die klassische Runge-Kutta-Methode (8.44) vierter Ordnung verwendet werden. Um die am raschesten exponentiell abklingende Komponente  $e^{-75x}$  mit mindestens vierstelliger Genauigkeit zu erfassen, ist mit einer Schrittweite  $h_1 = 0.0025$  zu arbeiten. Diese Schrittweite bestimmt sich aus der Forderung, dass  $e^{-75h_1}$  mit  $F(-75h_1)$  gemäß (8.104) auf fünf Stellen übereinstimmt. Integrieren wir (8.115) über 60 Schritte bis zur Stelle  $x_1 = 0.150$ , dann ist  $e^{-75 \cdot 0.150} = e^{-11.25} \doteq 0.000013$  gegenüber  $e^{-45 \cdot 0.150} = e^{-6.75} \doteq 0.001171$  bedeutend kleiner. Diese rasch abklingende und bereits kleine Komponente braucht ab dieser Stelle nicht mehr so genau integriert zu werden, und wir können die Schrittweite vergrößern. Damit jetzt die Komponente  $e^{-45x}$  mit einer vergleichbaren Genauigkeit behandelt wird, müssen wir die Schrittweite  $h_2 = 0.005$  wählen. Nach weiteren 30 Schritten erreichen wir  $x_2 = x_1 + 30h_2 = 0.300$ . Jetzt ist auch  $e^{-45 \cdot 0.300} = e^{-13.5} \doteq 0.0000014$  sehr klein geworden, so dass wir die Schrittweite nochmals vergrößern können. Betrachten wir die sehr langsam abklingende Komponente  $e^{-0.5x}$  für sich allein, so würde das Runge-Kutta-Verfahren (8.44) diese mit einer Schrittweite  $\tilde{h} = 0.4$  mit der geforderten Genauigkeit wiedergeben. Doch verletzt diese Schrittweite wegen  $\tilde{\mu} = -75\tilde{h} = -30$  die Bedingung bei weitem, dass  $\tilde{\mu}$  im Intervall der absoluten Stabilität  $(-2.78, 0)$  liegen muss! Die maximal verwendbare Schrittweite  $h^*$  muss der Ungleichung  $h^* \leq 2.78/75 \doteq 0.037$  genügen. Mit  $h_3 = 0.035$  sind für eine Integration von (8.115) bis  $x = 24$ , wo  $|y_1(x)| \leq 0.0001$  wird, weitere 678 Integrationschritte nötig. In Abb. 8.11 ist die euklidische Norm des globalen Fehlers  $\|e_j\| = \|u_j - y(x_j)\|$  mit logarithmischem Maßstab in Abhängigkeit von  $x$  im Intervall  $[0, 1.2]$  dargestellt, falls die oben beschriebenen Schrittweiten gewählt werden. Jede Schrittweitenvergrößerung bewirkt einen vorübergehenden sprunghaften Anstieg der Norm des globalen Fehlers, da jeweils eine betreffende Lösungskomponente mit einem größeren Fehler integriert wird. Das langsame, im logarithmischen Maßstab praktisch lineare Anwachsen von  $\|e_j\|$  ab etwa  $x = 0.6$  entspricht dem Fehlerwachstum (8.35) für den globalen Fehler. Wird ab  $x_2 = 0.3$  statt  $h_3$  die Schrittweite  $\tilde{h}_3 = 0.045$  gewählt, welche der Bedingung der absoluten Stabilität wegen  $-75 \cdot 0.045 = -3.35 < -2.78$  nicht genügt, zeigt die Norm des globalen Fehlers die Instabilität des Runge-Kutta-Verfahrens an.  $\triangle$

Ein lineares, inhomogenes Differenzialgleichungssystem

$$y'(x) = \mathbf{A}y(x) + b(x), \quad \mathbf{A} \in \mathbb{R}^{n,n}, y, b \in \mathbb{R}^n \quad (8.118)$$

heißt *steif*, falls die Eigenwerte  $\lambda_j, j = 1, 2, \dots, n$ , der Matrix  $\mathbf{A}$  sehr unterschiedliche negative Realteile aufweisen. Als Maß der *Steifheit*  $S$  des Differenzialgleichungssystems (8.118) gilt der Quotient der Beträge der absolut größten und kleinsten Realteile der Eigenwerte

$$S := \max_j |Re(\lambda_j)| / \min_j |Re(\lambda_j)| \quad \text{für alle } \lambda_j \text{ mit } Re(\lambda_j) < 0. \quad (8.119)$$

Für das Differenzialgleichungssystem (8.115) ist  $S = 150$ . Es ist nicht besonders steif, denn in manchen Fällen erreicht  $S$  Werte zwischen  $10^3$  und  $10^6$ . Um solche Systeme mit einer nicht allzu kleinen Schrittweite integrieren zu können, kommen nur Verfahren in Betracht, deren Gebiet der absoluten Stabilität entweder die ganze linke Halbebene  $Re(\mu) < 0$  umfasst oder aber zumindest  $A(\alpha)$ -stabil sind. Absolut stabil sind die Trapezmethode (8.16) und die impliziten Runge-Kutta-Verfahren (8.14) und (8.46), während die impliziten  $m$ -Schritt-

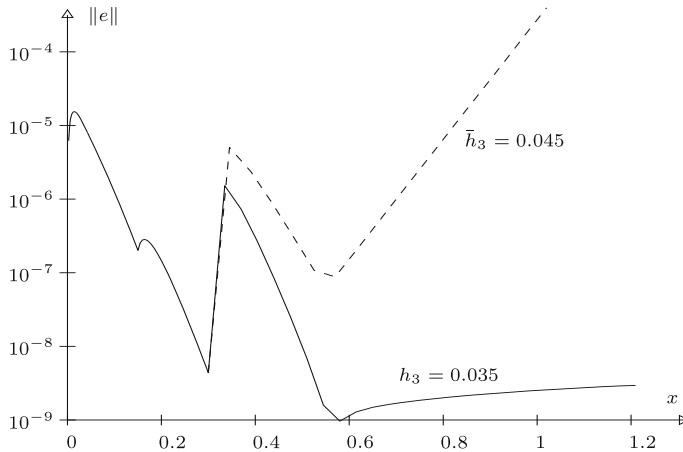


Abb. 8.11 Stabile und instabile Integration eines Differenzialgleichungssystems.

BDF-Methoden für  $m = 1$  bis  $m = 6$  wenigstens  $A(\alpha)$ -stabil sind. Alle diese Methoden erfordern aber in jedem Integrationsschritt die Lösung eines Gleichungssystems nach den Unbekannten.

Das Problem der Steifheit existiert ausgeprägt bei nichtlinearen Differenzialgleichungssystemen für  $n$  Funktionen

$$y'(x) = f(x, y(x)), \quad y(x) \in \mathbb{R}^n. \quad (8.120)$$

Hier kann die Steifheit nur *lokal* mittels einer Linearisierung zu erfassen versucht werden. Dabei übernimmt die *Funktional-* oder *Jacobi-Matrix*

$$\mathbf{J}(x, y) := f_y = \begin{pmatrix} \frac{\partial f_1}{\partial y_1} & \frac{\partial f_1}{\partial y_2} & \dots & \frac{\partial f_1}{\partial y_n} \\ \frac{\partial f_2}{\partial y_1} & \frac{\partial f_2}{\partial y_2} & \dots & \frac{\partial f_2}{\partial y_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_n}{\partial y_1} & \frac{\partial f_n}{\partial y_2} & \dots & \frac{\partial f_n}{\partial y_n} \end{pmatrix} \in \mathbb{R}^{n,n} \quad (8.121)$$

die Rolle der Matrix  $\mathbf{A}$  im linearen Problem (8.118). Das nichtlineare Differenzialgleichungssystem (8.120) wird als *steif* bezeichnet, falls die Eigenwerte  $\lambda_i$  der Jacobi-Matrix  $\mathbf{J}(x, y)$  (8.121) sehr unterschiedliche negative Realteile haben und somit der Wert  $S$  (8.119) groß ist. Das Maß der so definierten Steifheit von (8.120) ist jetzt abhängig von  $x$  und  $y$ , so dass sich  $S$  im Verlauf der Integration mit  $x_k$  und der aktuellen berechneten Lösung  $u_k$  und abhängig von den Anfangsbedingungen stark ändern kann. An dieser Stelle sei aber darauf hingewiesen, dass Beispiele von Differenzialgleichungssystemen und Lösungen konstruiert werden können, für welche die Eigenwerte der Jacobi-Matrix (8.121) irreführende Informationen bezüglich der Steifheit liefern [Aik 85, Dek 84, Lam 91].

**Beispiel 8.15.** Wir betrachten das nichtlineare Anfangswertproblem

$$\begin{aligned}\dot{y}_1 &= -0.1y_1 + 100y_2y_3, & y_1(0) &= 4, \\ \dot{y}_2 &= 0.1y_1 - 100y_2y_3 - 500y_2^2, & y_2(0) &= 2, \\ \dot{y}_3 &= 500y_2^2 - 0.5y_3, & y_3(0) &= 0.5.\end{aligned}\tag{8.122}$$

Tab. 8.5 Integration eines steifen Differenzialgleichungssystems.

$t_j$	$u_{1,j}$	$u_{2,j}$	$u_{3,j}$	$\lambda_1$	$\lambda_2$	$\lambda_3$	$S$	$h$
0	4.0000	2.0000	0.5000	-0.00025	-219.06	-1831.5	$7.4 \cdot 10^6$	0.0002
0.001	4.1379	0.9177	1.4438	-0.00054	-86.950	-975.74	$1.8 \cdot 10^6$	
0.002	4.2496	0.5494	1.6996	-0.00090	-45.359	-674.62	$7.5 \cdot 10^5$	
0.003	4.3281	0.3684	1.8013	-0.00133	-26.566	-522.56	$3.9 \cdot 10^5$	
0.004	4.3846	0.2630	1.8493	-0.00184	-16.588	-431.91	$2.4 \cdot 10^5$	
0.005	4.4264	0.1952	1.8743	-0.00243	-10.798	-372.48	$1.5 \cdot 10^5$	0.0010
0.006	4.4581	0.1489	1.8880	-0.00310	-7.2503	-331.06	$1.1 \cdot 10^5$	
0.008	4.5016	0.0914	1.9001	-0.00465	-3.5318	-278.45	$6.0 \cdot 10^4$	
0.010	4.5287	0.0588	1.9038	-0.00620	-1.9132	-247.81	$4.0 \cdot 10^4$	0.0025
0.020	4.5735	0.0097	1.8985	-0.00444	-0.5477	-199.60	$4.5 \cdot 10^4$	
0.030	4.5795	0.0035	1.8892	-0.00178	-0.5063	-192.48	$1.1 \cdot 10^5$	
0.040	4.5804	0.0026	1.8799	-0.00134	-0.5035	-190.65	$1.4 \cdot 10^5$	
0.050	4.5805	0.0025	1.8705	-0.00128	-0.5032	-189.60	$1.5 \cdot 10^5$	0.0050
0.10	4.5803	0.0025	1.8245	-0.00134	-0.5034	-185.03	$1.4 \cdot 10^5$	
0.15	4.5800	0.0025	1.7796	-0.00140	-0.5036	-180.60	$1.3 \cdot 10^5$	
0.20	4.5798	0.0026	1.7358	-0.00147	-0.5039	-176.29	$1.2 \cdot 10^5$	
0.25	4.5796	0.0027	1.6931	-0.00154	-0.5042	-172.08	$1.1 \cdot 10^5$	0.010
0.50	4.5782	0.0030	1.4951	-0.00196	-0.5060	-152.63	$7.8 \cdot 10^4$	
0.75	4.5765	0.0034	1.3207	-0.00247	-0.5086	-135.56	$5.5 \cdot 10^4$	
1.00	4.5745	0.0038	1.1670	-0.00311	-0.5124	-120.63	$3.9 \cdot 10^4$	0.020
2.00	4.5601	0.0061	0.7177	-0.00710	-0.5482	-77.88	$1.1 \cdot 10^4$	
5.00	4.3899	0.0134	0.2590	-0.01749	-0.9863	-38.92	$2.2 \cdot 10^3$	0.050
10.00	3.8881	0.0141	0.2060	-0.01854	-1.1115	-34.14	$1.8 \cdot 10^3$	

Das System (8.122) beschreibt die kinetische Reaktion von drei chemischen Substanzen  $Y_1$ ,  $Y_2$ ,  $Y_3$  nach dem Massenwirkungsgesetz, wobei die drei unbekannten Funktionen  $y_1(t)$ ,  $y_2(t)$ ,  $y_3(t)$  die entsprechenden Konzentrationen der Substanzen zum Zeitpunkt  $t$  bedeuten. Die Reaktionen laufen mit sehr unterschiedlichen Zeitkonstanten ab, was in den verschiedenen großen Koeffizienten in (8.122) zum Ausdruck kommt. Die Jacobi-Matrix  $\mathbf{J}(t, y)$  des Systems (8.122) lautet

$$\mathbf{J}(t, y) = \begin{pmatrix} -0.1 & 100y_3 & 100y_2 \\ 0.1 & -100y_3 - 1000y_2 & -100y_2 \\ 0 & 1000y_2 & -0.5 \end{pmatrix}.\tag{8.123}$$

Die Matrixelemente von  $\mathbf{J}$  sind vom Ablauf der chemischen Reaktion abhängig; deshalb sind die

Eigenwerte  $\lambda_i$  von  $\mathbf{J}(t, y)$  zeitabhängig. Zur Startzeit  $t = 0$  sind die Eigenwerte von

$$\mathbf{J}(0, y_0) = \begin{pmatrix} -0.1 & 50 & 200 \\ 0.1 & -2050 & -200 \\ 0 & 2000 & -0.5 \end{pmatrix}$$

$\lambda_1 \doteq -0.000249$ ,  $\lambda_2 \doteq -219.0646$ ,  $\lambda_3 \doteq -1831.535$ . Folglich ist  $S \doteq 7.35 \cdot 10^6$ , und das Differenzialgleichungssystem (8.122) ist zum Zeitpunkt  $t = 0$  sehr steif. Um das Problem der Steifheit zu illustrieren, integrieren wir (8.122) mit Hilfe der klassischen Runge-Kutta-Methode (8.44) vierter Ordnung. Der absolut größte, negative Eigenwert  $\lambda_3$  verlangt eine kleine Schrittweite  $h = 0.0002$ , damit die zugehörige rasch abklingende Lösungskomponente mit einem lokalen, relativen Diskretisierungsfehler von etwa  $10^{-4}$  integriert wird. Nach 25 Integrationsschritten hat die Steifheit des Systems abgenommen (vgl. Tab. 8.5), und da jetzt  $\lambda_3 \doteq -372.48$  ist, kann die Schrittweite auf  $h = 0.001$  vergrößert werden, denn die rasch abklingende und bereits mit einem kleinen Anteil beteiligte Komponente kann schon mit geringerer (relativer) Genauigkeit behandelt werden. Dasselbe gilt auch für die weitere Integration, deren Resultat in Tab. 8.5 zusammengestellt ist. Angegeben sind auszugsweise zu ausgewählten Zeitpunkten die berechneten Näherungswerte der drei Lösungsfunktionen, die aus der Jacobi-Matrix  $\mathbf{J}$  resultierenden Eigenwerte  $\lambda_1$ ,  $\lambda_2$ ,  $\lambda_3$ , das Maß  $S$  der Steifheit und die verwendeten Schrittweiten. Nach einer raschen Abnahme nimmt  $S$  vorübergehend wieder etwas zu, um dann mit wachsender Zeit  $t$  monoton abzunehmen. Ab  $t = 0.25$  wird die Schrittweite  $h$  durch das Stabilitätsintervall der verwendeten expliziten Runge-Kutta-Methode beschränkt.

△

## 8.5 Anwendung: Lotka-Volterras Wettbewerbsmodell

Das Lotka–Volterrascche System besteht aus autonomen<sup>1</sup> Differenzialgleichungen mit quadratischer, also nichtlinearer rechter Seite. Es beschreibt sowohl das Wettbewerbs- als auch das Räuber-Beute-Modell [Heu 09], für die wir zwei Fälle diskutieren wollen.

### Kampf um Ressourcen

Wenn eine Population  $P = P(t)$  von einer beschränkten Ressource  $R$  lebt, so entwickelt sie sich nach dem Wachstumsgesetz

$$\dot{P} = \alpha P - \beta P^2, \quad \alpha, \beta > 0. \quad (8.124)$$

Konkurrieren zwei Populationen  $P_1(t)$  und  $P_2(t)$  um die gleiche beschränkte Ressource, so findet ein Wettbewerb statt; jede Population behindert das Wachstum der anderen. Ihre Bestände entwickeln sich dann aus den Anfangsbeständen  $P_{10}$  und  $P_{20}$  nach dem System

$$\dot{P}_k = \alpha_k P_k - \beta_k P_k^2 - \gamma_k P_1 P_2, \quad \alpha_k, \beta_k, \gamma_k > 0, \quad (k = 1, 2). \quad (8.125)$$

Der hinzugekommene gemischte Term  $\gamma_k P_1 P_2$  stellt also den Einfluss der Konkurrenz dar. Wir wollen das konkrete System

$$\begin{aligned} \dot{P}_1 &= 0.004 P_1 (50 - P_1 - 0.75 P_2) \\ \dot{P}_2 &= 0.001 P_2 (100 - P_2 - 3 P_1) \end{aligned} \quad (8.126)$$

---

<sup>1</sup>Die rechte Seite ist nicht explizit von  $t$  abhängig.

untersuchen. Hier ist es zunächst interessant, die so genannten stationären Punkte zu bestimmen. Das sind die Gleichgewichtszustände der Populationen, in denen die Lösung der Differenzialgleichung zeitunabhängig ist, also konstant bleibt.

Ist  $P_2 = 0$ , so gilt  $\dot{P}_1 = 0.004 P_1 (50 - P_1)$ , d.h., die erste Population wächst oder schrumpft bis zum Grenzwert  $P_1 = 50$  (oder 0) und bleibt dann konstant.

Ist  $P_1 = 0$ , so gilt  $\dot{P}_2 = 0.001 P_2 (100 - P_2)$ , und der entsprechende Grenzwert für die zweite Population ist  $P_2 = 100$  (oder 0).

Ein stationärer Punkt ist offensichtlich auch der Trivialzustand  $(P_1, P_2) = (0, 0)$ . Einen weiteren stationären Punkt findet man, indem man die Klammerausdrücke der rechten Seite von (8.126) gleich null setzt:

$$\begin{aligned} P_1 + 0.75 P_2 &= 50, \\ 3 P_1 + P_2 &= 100. \end{aligned}$$

Das ergibt den Punkt  $(20, 40)$ . Es ist wichtig zu wissen, ob diese stationären Punkte lokal stabil sind, d.h., ob sie die Lösung anziehen oder abstoßen (instabil). Lokal stabil ist ein stationärer Punkt, wenn die Eigenwerte der Jacobi-Matrix  $J(P_1, P_2)$  (8.121) negative Realteile besitzen. Im symmetrischen Fall müssen also alle Eigenwerte negativ sein. Es ist hier

$$J(P_1, P_2) = \begin{pmatrix} 0.2 - 0.008P_1 - 0.003P_2 & -0.003P_1 \\ -0.003P_1 & 0.1 - 0.003P_1 - 0.002P_2 \end{pmatrix}.$$

Zur Überprüfung der negativen Definitheit in den stationären Punkten müssen wir also vier kleine Eigenwertprobleme – quadratische Gleichungen – lösen. Wir wollen nur die Ergebnisse wiedergeben:

Punkt	$\lambda_1$	$\lambda_2$	lokal stabil?
$(0,0)$	0.2	0.1	Nein
$(0,100)$	-0.1	-0.1	Ja
$(20,40)$	0.027	-0.14	Nein
$(50,0)$	-0.2	-0.05	Ja

Diese Verhältnisse lassen sich folgendermaßen interpretieren:

Es werden weder beide Populationen aussterben noch beide überleben, weil sowohl der stationäre Punkt  $(0,0)$  als auch  $(20,40)$  instabil sind. Welche der beiden Populationen überlebt, d.h. welcher der stabilen stationären Punkte von der Lösung des Systems angesteuert wird, hängt von den Anfangswerten – hier den Anfangspopulationen  $P_{10}$  und  $P_{20}$  – ab.

In Abb. 8.12 haben wir die stationären Punkte durch kleine Vollkreise gekennzeichnet, und wir haben die “Bahnenlinien” (Phasenkurven) mehrerer Lösungen mit verschiedenen Anfangswerten (+) eingezeichnet. Sie laufen in die beiden lokalen stabilen stationären Punkte  $(0,100)$  oder  $(50,0)$ . Die Trennungslinie (Separatrix) zwischen Anfangswerten, die in den einen oder anderen der beiden Gleichgewichtszustände laufen, ist gut abzulesen. Auch Lösungen, die sich dem instabilen Gleichgewichtszustand  $(20, 40)$  annähern, werden von diesem abgestoßen und knicken scharf in Richtung eines stabilen Gleichgewichtszustands ab.

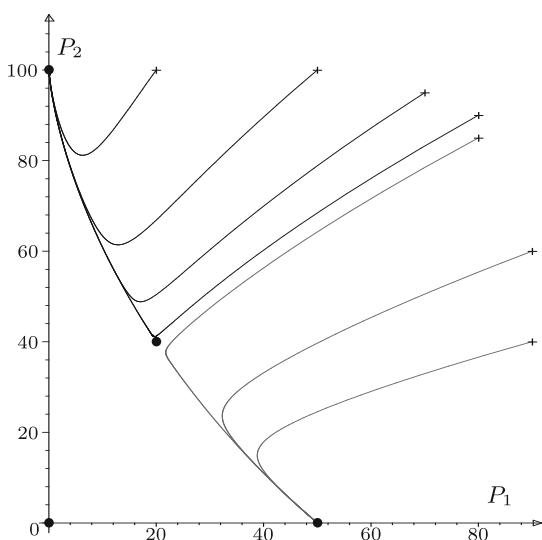


Abb. 8.12 Wettbewerbsmodell: Abhängig vom Anfangszustand stirbt eine der Populationen aus.

### Räuber und Ressourcen

Wir wollen jetzt die biologischen Verhältnisse gegenüber (8.124) und (8.125) etwas ändern.  $P_1$  repräsentiert jetzt Beutetiere, die von der begrenzten Ressource  $R$  leben. Erbeutet werden sie von Räubern, die durch  $P_2$  repräsentiert werden und ausschließlich von den Beutetieren  $P_1$  leben. Dabei kann man z.B. an Hasen und Füchse denken. Jetzt ergibt sich mit geeigneten Zahlen das Modellsystem

$$\begin{aligned}\dot{P}_1 &= 2P_1(1 - 0.3P_1 - P_2), \\ \dot{P}_2 &= P_2(P_1 - 1).\end{aligned}\tag{8.127}$$

Wird dieses System entsprechend der Vorgehensweise oben untersucht, so ergeben sich die stationären Punkte und zugehörigen Eigenwerte der Jacobi-Matrix wie folgt:

Punkt	$\lambda_1$	$\lambda_2$	lokal stabil?
(0,0)	2	-1	Nein
(0,1)	-1	0	Nein
(1,0)	0.8	0	Nein
(1,0.7)	$-0.3 + i\sqrt{1.49}$	$-0.3 - i\sqrt{1.49}$	Ja

Hier liegt also die biologisch glücklichere Situation vor, dass beide Populationen überleben können. Das System hat an der Stelle (1, 0.7) den einzigen stabilen Strudelpunkt. In diesem Gleichgewichtszustand landet es nach einiger Zeit von jedem Anfangswertpaar aus, das nicht genau in einem der instabilen stationären Punkte liegt. In Abb. 8.13 sehen wir zwei

Phasenkurven, die in den Anfangswerten  $(2, 1)$  bzw.  $(0.2, 1.5)$  gestartet wurden.

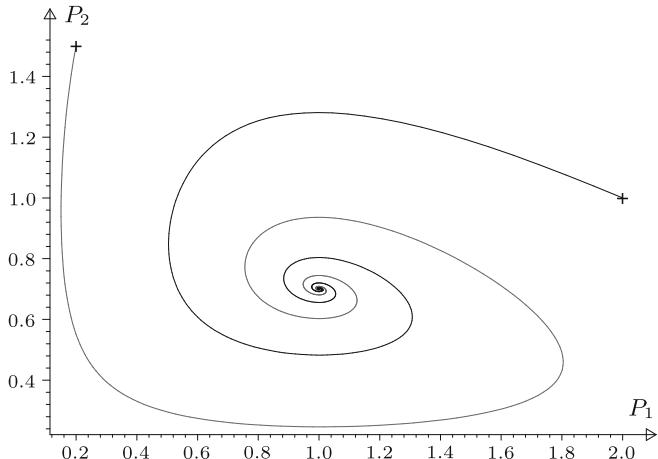


Abb. 8.13 Räuber-Beute-Modell: Beide Populationen überleben.

## 8.6 Software

Einige Pakete zur numerischen Lösung von Anfangswertproblemen bei gewöhnlichen Differentialgleichungen enthalten Routinen sowohl zu Anfangs- als auch zu Randwertproblemen (siehe Kapitel 9). Andererseits wird bei den Routinen zu Anfangswertproblemen zwischen steifen und wenig oder gar nicht steifen (engl. *non-stiff and mildly stiff*) Problemen unterschieden, so z.B. bei der Klassifizierung des GAMS.

Die NAG-FORTRAN-Bibliothek enthält im Kapitel D02 insgesamt 62 Routinen zur Lösung von Anfangs- und Randwertproblemen, davon in den Unterkapiteln D02M und D02N allein 19 Routinen für steife Probleme. Mit den Routinen des Unterkapitels D02L können auch Systeme von Differentialgleichungen zweiter Ordnung gelöst werden. Dazu werden spezielle Runge-Kutta-Nyström-Verfahren angewendet. Die NAG-C-Bibliothek enthält dagegen im gleichen Kapitel nur 20 Routinen.

MATLAB verfügt über sieben Routinen zur Lösung von Anfangswertproblemen, zwei davon benutzen explizite Runge-Kutta-Verfahren, eine ein Prädiktor-Korrektor-Schema mit Adams-Verfahren variabler Ordnung, die im Allgemeinen eine höhere Genauigkeit liefern. Für steife oder mild steife Probleme gibt es weitere vier Routinen, von denen zwei auch differentiell-algebraische Probleme lösen können; das sind Systeme, bei denen Differentialgleichungen und nichtlineare Gleichungen gekoppelt vorkommen. MATLAB erlaubt neben einer Reihe von wichtigen Optionen auch eine Ereignis-(Event-)Steuerung, die es ermöglicht, den Lösungsprozess beim Eintritt gewisser Bedingungen zu beenden. Zudem bietet MATLAB besondere Graphik-Routinen zum Zeichnen der Lösungen von Differentialgleichungen an.

Es gibt spezielle Pakete zur Lösung von gewöhnlichen Differentialgleichungen wie ODE und ODEPACK, die überwiegend die gleichen Methoden anwenden oder sogar nahezu identische Routinen vertreiben wie die großen Bibliotheken NAG und IMSL.

Einige Autoren haben ihren Büchern über die numerische Lösung von Differentialgleichungen Programme hinzugefügt, die via Internet frei verfügbar sind. Hier möchten wir [Deu 08a]<sup>2</sup> und [Hai 93, Hai 96]<sup>3</sup> erwähnen.

Unsere Problemlöseumgebung PAN (<http://www.upb.de/SchwarzKoeckler/>) verfügt über drei Programme zur Lösung von Anfangswertproblemen mit einem Runge-Kutta-Verfahren 4. Ordnung, einem Mehrschrittverfahren mit einem Adams-Verfahren variabler Ordnung und Schrittweite und einem ebensolchen BDF-Verfahren.

## 8.7 Aufgaben

**Aufgabe 8.1.** Man bestimme die exakte Lösung der Anfangswertaufgabe

$$y' = \frac{2x}{y^2}, \quad y(0) = 1$$

und berechne im Intervall  $[0, 3]$  die Näherungslösungen

- a) nach der Methode von Euler mit den Schrittweiten  $h = 0.1, 0.01, 0.001$ ;
- b) nach der verbesserten Polygonzug-Methode und der Methode von Heun mit den Schrittweiten  $h = 0.1, 0.05, 0.025, 0.01$ ;
- c) nach je einem Runge-Kutta-Verfahren der Ordnung drei und vier mit den Schrittweiten  $h = 0.2, 0.1, 0.05, 0.025$ .

Mit den an den Stellen  $x_j = 0.2j$ ,  $j = 1, 2, \dots, 15$ , berechneten globalen Fehlern verifizierte man die Fehlerordnungen der Methoden.

**Aufgabe 8.2.** Die Anfangswertaufgaben

$$\text{a)} \quad y' = \frac{1}{1+4x^2} - 8y^2, \quad y(0) = 0; \quad x \in [0, 4]$$

$$\text{b)} \quad y' = \frac{1}{1+4x^2} + 0.4y^2, \quad y(0) = 0; \quad x \in [0, 4]$$

$$\text{c)} \quad y' = \frac{1-x^2-y^2}{1+x^2+xy}, \quad y(0) = 0; \quad x \in [0, 10]$$

sollen mit einer Runge-Kutta-Methode zweiter und vierter Ordnung und mit einer Schrittweitensteuerung in den angegebenen Intervallen näherungsweise so gelöst werden, dass der Schätzwert des lokalen Diskretisierungsfehlers betragsmäßig höchstens  $\varepsilon$  ( $\varepsilon = 10^{-4}, 10^{-6}, 10^{-8}$ ) ist. Für die Steuerung der Schrittweite experimentiere man mit verschiedenen Strategien.

**Aufgabe 8.3.** Die kleine Auslenkung  $x(t)$  eines schwingenden Pendels mit Reibung wird durch die Differentialgleichung zweiter Ordnung

$$\ddot{x}(t) + 0.12\dot{x}(t) + 2x(t) = 0$$

---

<sup>2</sup><http://www.zib.de/de/numerik/software/codelib.html>

<sup>3</sup><http://www.unige.ch/~hairer/software.html>

beschrieben. Die Anfangsbedingung sei  $x(0) = 1$ ,  $\dot{x}(0) = 0$ . Das zugehörige System von Differenzialgleichungen erster Ordnung ist mit der klassischen Runge-Kutta-Methode vierter Ordnung mit drei verschiedenen Schrittweiten  $h$  näherungsweise zu lösen, und die Näherungslösung  $x_j$  soll graphisch dargestellt werden. Zudem ist der globale Fehler mit Hilfe der exakten Lösung zu berechnen und sein Verhalten zu studieren. Erfolgt die genäherte Integration der beiden komplexen Lösungsanteile amplituden- und phasentreu?

**Aufgabe 8.4.** Das Differenzialgleichungssystem

$$\dot{x} = 1.2x - x^2 - \frac{xy}{x + 0.2}$$

$$\dot{y} = \frac{1.5xy}{x + 0.2} - y$$

beschreibt ein Räuber-Beute-Modell der Biologie, wobei  $x(t)$  eine Maßzahl für die Anzahl der Beutetiere und  $y(t)$  eine Maßzahl für die Anzahl der Raubtiere bedeuten. Für die zwei verschiedenen Anfangsbedingungen  $x(0) = 1$ ,  $y(0) = 0.75$  und  $\bar{x}(0) = 0.75$ ,  $\bar{y}(0) = 0.25$  ist das System mit dem klassischen Runge-Kutta-Verfahren vierter Ordnung im Intervall  $0 \leq t \leq 30$  mit der Schrittweite  $h = 0.1$  näherungsweise zu lösen. Die Lösung soll in der  $(x, y)$ -Phasenebene dargestellt und das gefundene Ergebnis interpretiert werden.

Als Variante löse man die Aufgabe mit automatischer Schrittweitensteuerung nach der Methode von Fehlberg.

**Aufgabe 8.5.** Man leite das explizite Dreischrittverfahren von Adams-Basforth und das implizite Dreischrittverfahren von Adam-Moulton auf Grund der Integralgleichung und als Spezialfall eines allgemeinen Mehrschrittverfahrens her. Sodann zeige man, dass die Ordnung der Verfahren drei bzw. vier ist, und man verifiziere die Fehlerkonstanten der lokalen Diskretisierungsfehler in Tab. 8.3 für diese beiden Methoden.

**Aufgabe 8.6.** Die Differenzialgleichungen der Aufgaben 8.1 und 8.2 sind mit den Adams-Basforth-Moulton-Methoden ABM33 (8.71) und ABM43 (8.72) näherungsweise mit der Schrittweite  $h = 0.1$  zu lösen. Um die Resultate der beiden Verfahren fair vergleichen zu können, sollen mit dem klassischen Runge-Kutta-Verfahren vierter Ordnung drei Startwerte  $u_1, u_2, u_3$  bestimmt werden.

**Aufgabe 8.7.** Welches sind die Gebiete der absoluten Stabilität der folgenden Mehrschrittmethoden?

a) AB3:  $u_{j+1} = u_j + \frac{h}{12}[23f_j - 16f_{j-1} + 5f_{j-2}]$ ;

b) AM2:  $u_{j+1} = u_j + \frac{h}{12}[5f(x_{j+1}, u_{j+1}) + 8f_j - f_{j-1}]$ ;

c) Prädiktor-Korrektor-Methode ABM32.

Für die ABM32-Methode zeige man, dass der Koeffizient des Hauptanteils des lokalen Diskretisierungsfehlers gleich demjenigen des AM2-Verfahrens ist.

**Aufgabe 8.8.** Zur Problematik der stabilen Integration betrachten wir das lineare homogene Differenzialgleichungssystem [Lam 91]

$$\begin{aligned} y'_1 &= -21y_1 + 19y_2 - 20y_3, & y_1(0) &= 1, \\ y'_2 &= 19y_1 - 21y_2 + 20y_3, & y_2(0) &= 0, \\ y'_3 &= 40y_1 - 40y_2 - 40y_3, & y_3(0) &= -1. \end{aligned}$$

Die Eigenwerte der Matrix des Systems sind  $\lambda_1 = -2$ ,  $\lambda_{2,3} = -40 \pm 40i$ . Das System soll mit der Trapezmethode und dem klassischen Runge-Kutta-Verfahren vierter Ordnung näherungsweise gelöst werden. Welche Schrittweiten  $h$  sind zu wählen, um mit den beiden Methoden im Intervall  $[0, 0.3]$  eine vierstellige Genauigkeit der Näherungen zu garantieren? Dazu ist  $e^{\lambda h}$  mit dem entsprechenden  $F(\lambda h)$  zu vergleichen. Mit welchen Schrittweiten kann anschließend im Intervall  $[0.3, 5]$  weiter integriert werden? Man überprüfe die Richtigkeit der Aussagen, indem das System numerisch gelöst wird. Welche maximale Schrittweite ist im Fall der ABM43-Methode (8.72) möglich?

**Aufgabe 8.9.** Das nichtlineare Differenzialgleichungssystem

$$\begin{aligned} \dot{y}_1 &= -0.01y_1 + 0.01y_2 \\ \dot{y}_2 &= y_1 - y_2 - y_1y_3 \\ \dot{y}_3 &= y_1y_2 - 100y_3 \end{aligned}$$

ist für die Anfangsbedingung  $y_1(0) = 0$ ,  $y_2(0) = 1$ ,  $y_3(0) = 1$  in Abhängigkeit von  $t$  auf Steifheit zu untersuchen.

**Aufgabe 8.10.** Man zeige, dass die implizite zweistufige Runge-Kutta-Methode

$$\begin{aligned} k_1 &= f\left(x_j, u_j + \frac{1}{4}hk_1 - \frac{1}{4}hk_2\right) \\ k_2 &= f\left(x_j + \frac{2}{3}h, u_j + \frac{1}{4}hk_1 + \frac{5}{12}hk_2\right) \\ u_{j+1} &= u_j + \frac{h}{4}(k_1 + 3k_2) \end{aligned}$$

die Ordnung drei besitzt und absolut stabil ist. Die Funktion  $F(\mu)$ , welche für das Modellproblem (8.102) resultiert, hat im Gegensatz zu derjenigen der Trapezmethode die Eigenschaft, dass  $\lim_{\mu \rightarrow -\infty} F(\mu) = 0$  gilt. Welche Konsequenzen ergeben sich daraus für rasch abklingende Komponenten von steifen Systemen? Man wende diese Methode zur numerischen Integration des Differenzialgleichungssystems von Aufgabe 8.8 an unter Verwendung einer konstanten Schrittweite  $h$ . Was kann festgestellt werden?

## 9 Rand- und Eigenwertprobleme bei gewöhnlichen Differenzialgleichungen

Randwertprobleme bei gewöhnlichen Differenzialgleichungen beschreiben stationäre Systeme wie die Durchbiegung eines belasteten Balkens. Eigenwertprobleme beschreiben charakteristische Eigenschaften solcher Systeme wie seine Eigenfrequenzen. In beiden Fällen müssen mindestens zwei *Randwerte* bekannt sein.

Für Rand- und Eigenwertprobleme gibt es keine so einheitliche Theorie wie für Anfangswertaufgaben. Aber auch sie spielen in den Anwendungen eine bedeutende Rolle.

Wir wollen zunächst auf die Problematik der Existenz und Eindeutigkeit von Lösungen eingehen, die auch für die numerische Behandlung von Bedeutung sind. Sie sind für Rand- und Eigenwertprobleme wesentlich schwieriger zu klären als bei Anfangswertproblemen, die wir in Kapitel 8 behandelt haben.

Nachdem wir analytische Methoden für lineare Randwertaufgaben betrachtet haben, kommen wir zu zwei wichtigen numerischen Verfahrensfamilien, die auch auf nichtlineare Probleme angewendet werden können: die Schießverfahren und die Differenzenverfahren. Für eingehendere Darstellungen verweisen wir auf [Asc 95, Kel 92, Sto 05].

### 9.1 Problemstellung und Beispiele

Bei Randwertaufgaben (RWA) werden neben einer zu lösenden Differenzialgleichung

$$y^{(n)} = f(x, y, y', \dots, y^{(n-1)}) \quad \text{für } x \in (a, b) \quad (9.1)$$

Bedingungen an den beiden Randpunkten des betrachteten Lösungsintervalls vorgegeben:

$$\begin{aligned} r_1(y(a), y'(a), \dots, y^{(n-1)}(a), y(b), y'(b), \dots, y^{(n-1)}(b)) &= 0, \\ &\vdots && \vdots && \vdots \\ r_n(y(a), y'(a), \dots, y^{(n-1)}(a), y(b), y'(b), \dots, y^{(n-1)}(b)) &= 0. \end{aligned} \quad (9.2)$$

Zu lösen ist also normalerweise eine einzelne Differenzialgleichung höherer als erster Ordnung. Für die Behandlung in Software-Paketen oder Programm-Bibliotheken muss diese Differenzialgleichung oft in ein äquivalentes System 1. Ordnung umgeformt werden, siehe Lemma 8.1. Besonders wichtig für die Anwendungen sind Randwertaufgaben 2. Ordnung. Sie treten mit Abstand am häufigsten auf. Wir beschränken uns deshalb hauptsächlich auf die Betrachtung dieser Aufgaben.

### Randwertaufgabe 2. Ordnung mit linearen Randbedingungen

Gegeben seien eine stetige Funktion  $f(x, y, z)$  und reelle Zahlen  $\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1$  und  $\gamma_2$ . Gesucht ist eine im Intervall  $(a, b)$  zweimal stetig differenzierbare Funktion  $y(x)$ , für die gilt

$$y'' = f(x, y, y'), \quad (9.3)$$

$$\begin{aligned} r_1(y(a), y'(a)) &:= \alpha_1 y(a) + \alpha_2 y'(a) = \gamma_1, \quad (\alpha_1, \alpha_2) \neq (0, 0), \\ r_2(y(b), y'(b)) &:= \beta_1 y(b) + \beta_2 y'(b) = \gamma_2, \quad (\beta_1, \beta_2) \neq (0, 0). \end{aligned} \quad (9.4)$$

Über die Existenz einer Lösung kann man für dieses allgemein formulierte Problem ohne Weiteres keine Aussage machen.

### Lineare Randwertaufgabe 2. Ordnung

Ist in (9.3) auch die Differenzialgleichung linear und von der Form

$$y'' + a_1(x) y' + a_0(x) y = f(x), \quad (9.5)$$

mit den Randbedingungen (9.4), dann können die Voraussetzungen für die Existenz einer eindeutigen Lösung leicht angegeben werden. Sei  $(y_1(x), y_2(x))$  ein Fundamentalsystem der zugehörigen homogenen Differenzialgleichung  $y'' + a_1(x)y' + a_0(x)y = 0$ , dann ist das Randwertproblem eindeutig lösbar, falls

$$\begin{vmatrix} r_1(y_1(a), y'_1(a)) & r_1(y_2(a), y'_2(a)) \\ r_2(y_1(b), y'_1(b)) & r_2(y_2(b), y'_2(b)) \end{vmatrix} \neq 0. \quad (9.6)$$

### Sturmsche Randwertaufgabe

(Charles Sturm (1803–1855))

Ist in (9.5) die Funktion  $a_1$  stetig, so kann die Differenzialgleichung immer auf die Form

$$-(p(x)y')' + q(x)y = f(x) \quad (9.7)$$

gebracht werden mit einer in  $[a, b]$  stetig differenzierbaren Funktion  $p(x) > 0$ . Sind die Funktionen  $q$  und  $f$  stetig, die Randbedingungen von der Form (9.4) und die Voraussetzung (9.6) erfüllt, dann existiert wieder eine eindeutige Lösung. Man nennt diese Form *selbstadjungiert* oder auch Sturmsche Randwertaufgabe. Diese Form ist auch für die numerische Lösung vorteilhaft, siehe etwa Seite 406.

Weitere Einzelheiten findet man in [Heu 09]; siehe auch Abschnitt 9.2.

Auf die Frage nach der Existenz und Eindeutigkeit von Lösungen kommen wir unten im Spezialfall linearer Probleme und bei Beispielen zurück.

### Eigenwertaufgabe 2. Ordnung

Eigenwertprobleme sind Randwertprobleme, die zusätzlich einen Parameter  $\lambda$  in der Problemstellung enthalten. Gesucht sind Werte von  $\lambda$  (*Eigenwerte*), für die die Randwertaufgabe lösbar ist, und die zugehörigen Lösungen (*Eigenfunktionen*).

Gegeben seien stetige Funktionen  $f(x, y, z)$  und  $g(x, y, z)$ . Gesucht sind Eigenwerte  $\lambda$  und zugeordnete, in  $(a, b)$  zweimal stetig differenzierbare Funktionen  $y(x) \not\equiv 0$ , für die gilt:

$$\begin{aligned} y'' + f(x, y, y') &= \lambda g(x, y, y'), \\ y(a) = 0, \quad y(b) &= 0. \end{aligned} \tag{9.8}$$

Unter gewissen Voraussetzungen bilden die Lösungen ein abzählbar unendliches System  $(\lambda_i, y_i(x))$  aus Eigenwerten und Eigenfunktionen, das *Eigenwertproblem* heißt.

### Sturm-Liouville'sche Eigenwertaufgabe

(Joseph Liouville (1809–1882))

Sind bei gleichen Bezeichnungen die Voraussetzungen der Sturmschen Randwertaufgabe erfüllt und ist zusätzlich  $v(x)$  positiv und stetig, dann hat das Eigenwertproblem

$$\begin{aligned} -(p(x)y')' + q(x)y &= \lambda v(x)y, \\ \alpha_1 y(a) + \alpha_2 y'(a) &= 0, \quad (\alpha_1, \alpha_2) \neq (0, 0), \\ \beta_1 y(b) + \beta_2 y'(b) &= 0, \quad (\beta_1, \beta_2) \neq (0, 0), \end{aligned} \tag{9.9}$$

reelle, positive und voneinander verschiedene Eigenwerte

$$0 < \lambda_1 < \lambda_2 < \lambda_3 < \dots \rightarrow \infty$$

mit Eigenfunktionen  $y_k(x)$ , die  $k - 1$  Nullstellen in  $(a, b)$  haben (Oszillationseigenschaft).

Komplexere Problemstellungen ergeben sich durch

- höhere Ableitungen bzw. Systeme von Differenzialgleichungen,
- nichtlineare Randbedingungen,
- Differenzialgleichungen, die implizit in der höchsten Ableitung und nicht direkt nach dieser auflösbar sind.

Wir wollen mögliche Schwierigkeiten und die harmlose Normalsituation bei der Lösung von Randwertproblemen an einigen Beispielen verdeutlichen.

### Beispiel 9.1. Keine oder unendlich viele Lösungen

Die Randwertaufgabe

$$y'' + \pi^2 y = 0, \quad y(0) = 0, \quad y(1) = 1, \tag{9.10}$$

hat keine Lösung, während für

$$y'' + \pi^2 y = 0, \quad y(0) = 0, \quad y(1) = 0, \tag{9.11}$$

unendlich viele Lösungen existieren.

*Beweis.* (9.10) ist eine homogene lineare Differenzialgleichung mit konstanten Koeffizienten. Ihre Fundamentalslösungen  $\{\sin \pi x, \cos \pi x\}$  erfüllen (9.6) nicht. Jede Lösung hat die Form

$$y(x) = c_1 \sin \pi x + c_2 \cos \pi x \tag{9.12}$$

Hier setzen wir die Randpunkte ein, um  $c_1$  und  $c_2$  zu bestimmen:

$$y(0) = c_2 \implies c_2 = 0, \quad y(1) = -c_2 \implies c_2 = -1.$$

Also existiert keine Lösung.

(9.11) hat natürlich die gleichen Grundlösungen. Die Randbedingungen  $y(0) = y(1) = 0$  sind mit  $c_2 = 0$  erfüllt. Damit sind alle Funktionen

$$y(x) = c_1 \sin \pi x, \quad c_1 \in \mathbb{R} \text{ beliebig,}$$

Lösungen der Randwertaufgabe.  $\square$

Dieses Beispiel zeigt, dass die Existenz und Eindeutigkeit von Lösungen allein von den Randbedingungen abhängen kann. Das liegt daran, dass eine Randwertaufgabe eine Problemstellung im Großen ist, wir also nicht wie bei Anfangswertproblem lokal arbeiten und argumentieren können.  $\triangle$

### Beispiel 9.2. Mehrere Lösungen

$$y'' = y^5 - 10y + \frac{1}{2}, \quad y(0) = 0, \quad y'(1) = -3, \quad (9.13)$$

hat zwei Lösungen, siehe Abb. 9.1.

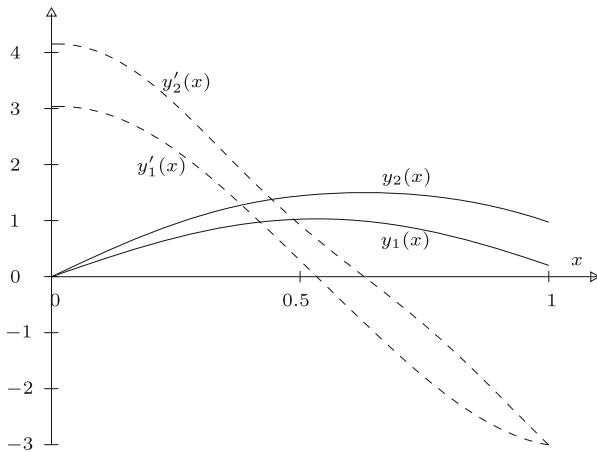


Abb. 9.1 Die Lösungen  $y_1(x)$ ,  $y_2(x)$  und ihre Ableitungen (- -).

Eine geringfügige Änderung der Koeffizienten dieser nichtlinearen Randwertaufgabe kann zu einer eindeutigen Lösung führen oder die Anzahl existierender Lösungen noch erhöhen. Die Eindeutigkeit von Lösungen kann zusätzlich von den Randbedingungen abhängen.

Verallgemeinern wir das Problem zu

$$\begin{aligned} y'' &= \eta_1 y^5 - \eta_2 y + \eta_3, \\ y(0) &= \alpha \quad y'(1) = \beta, \end{aligned} \quad (9.14)$$

so bekommen wir unterschiedlich viele Lösungen abhängig von den Werten der Parameter  $\eta_i$ . Die Lösung ist eindeutig für  $5\eta_1 y^4 > \eta_2$ . Die Anzahl der Lösungen wächst mit  $\eta_2$ .  $\triangle$

**Beispiel 9.3. Durchbiegung eines Balkens** Ein homogener, ideal elastischer Balken sei an seinen Enden frei beweglich gestützt. Er unterliege einer axialen Druckkraft  $P$  und der transversalen Belastung  $h(x)$ . Seine Länge sei 2, seine Biegesteifigkeit  $E J(x)$ ,  $-1 \leq x \leq 1$ , mit dem Elastizitätsmodul  $E$  und dem Flächenträgheitsmoment  $J(x)$ .  $J(x)$  ist proportional zu  $D^4(x)$ , wenn  $D$  der Durchmesser des Balkens ist.

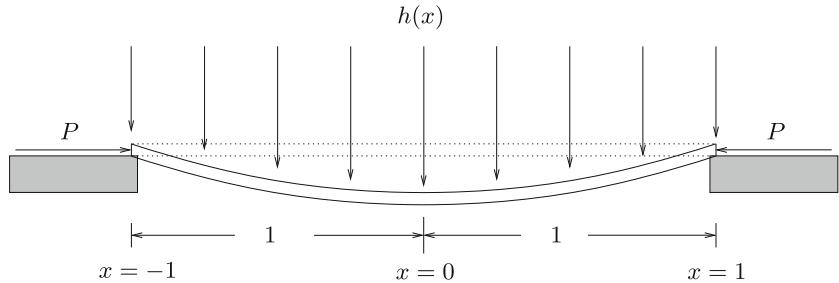


Abb. 9.2 Durchbiegung eines Balkens.

Gesucht ist das Biegemoment  $M$ , für das die Differenzialgleichung

$$M''(x) + \frac{P}{E J(x)} M(x) = -h(x)$$

gilt. Wegen der freien Beweglichkeit muss es an den Randpunkten verschwinden:

$$M(-1) = M(1) = 0.$$

Die Aufgabenstellung wird durch folgende Festlegungen vereinfacht:

Es sei  $h(x) \equiv h$ ,  $J(x) := \frac{J_0}{1+x^2}$ ,  $P := E J_0$ , d.h. der Balken, der an den Enden geringeren Querschnitt als in der Mitte hat, unterliegt konstanten Kräften. Jetzt liefert die Transformation  $y = -\frac{M}{h}$  die eindeutig lösbare Sturmsche Randwertaufgabe:

$$\begin{aligned} -y''(x) - (1+x^2)y(x) &= 1, \\ y(-1) = y(1) &= 0. \end{aligned} \tag{9.15}$$

△

## 9.2 Lineare Randwertaufgaben

### 9.2.1 Allgemeine Lösung

Eine Randwertaufgabe heißt *linear*, wenn sowohl die Differenzialgleichung als auch die Randbedingungen linear sind. Wir wollen jetzt den allgemeinen linearen Fall systematischer betrachten und Bedingungen für die Existenz und Eindeutigkeit von Lösungen formulieren. Auch spezielle numerische Methoden für lineare Randwertaufgaben werden behandelt. Dazu legen wir ein System von  $n$  linearen Differenzialgleichungen erster Ordnung zu Grunde, siehe auch Lemma 8.1.

Ein lineares Differenzialgleichungssystem lautet

$$\mathbf{y}'(x) = \mathbf{F}(x) \mathbf{y}(x) + \mathbf{g}(x), \quad (9.16)$$

wo  $\mathbf{F}(x) \in \mathbb{R}^{n,n}$  eine von  $x$  abhängige  $(n \times n)$ -Matrix und  $\mathbf{y}(x), \mathbf{g}(x) \in \mathbb{R}^n$   $n$ -dimensionale Vektorfunktionen sind.

Die zugehörigen  $n$  Randbedingungen (9.2) sind jetzt auch linear und lassen sich mit Hilfe von zwei Matrizen  $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n,n}$  und einem Vektor  $\mathbf{c} \in \mathbb{R}^n$  wie folgt formulieren:

$$\mathbf{A} \mathbf{y}(a) + \mathbf{B} \mathbf{y}(b) = \mathbf{c}. \quad (9.17)$$

Das homogene Differenzialgleichungssystem  $\mathbf{y}'(x) = \mathbf{F}(x) \mathbf{y}(x)$  besitzt im Intervall  $I := [a, b]$  unter der Voraussetzung, dass die Matrixelemente von  $\mathbf{F}(x)$  in  $I$  stetige Funktionen sind, ein System von  $n$  linear unabhängigen Lösungen  $\mathbf{y}_1(x), \mathbf{y}_2(x), \dots, \mathbf{y}_n(x)$ , die man zur *Fundamentalmatrix*  $\mathbf{Y}(x) := (\mathbf{y}_1(x), \mathbf{y}_2(x), \dots, \mathbf{y}_n(x)) \in \mathbb{R}^{n,n}$  zusammenfasst, die für alle  $x \in I$  regulär ist. Ein solches Fundamentalsystem lässt sich beispielsweise durch  $n$ -malige numerische Integration des homogenen Differenzialgleichungssystems unter den  $n$  speziellen Anfangsbedingungen

$$\mathbf{y}_k(a) = \mathbf{e}_k, \quad k = 1, 2, \dots, n, \quad (9.18)$$

wo  $\mathbf{e}_k \in \mathbb{R}^n$  den  $k$ -ten Einheitsvektor darstellt, näherungsweise bestimmen. Unter den oben genannten Voraussetzungen an  $\mathbf{F}(x)$  folgt dann aus  $\det \mathbf{Y}(a) = \det \mathbf{I} = 1$ , dass  $\mathbf{Y}(x)$  für alle  $x \in I$  regulär ist [Heu 09].

Für die allgemeine Lösung eines inhomogenen Differenzialgleichungssystems (9.16) mit  $\mathbf{g}(x) \not\equiv 0$  benötigt man eine partikuläre Lösung  $\mathbf{y}_0(x)$  von (9.16). Dafür gibt es auf Grund der Fundamentalmatrix  $\mathbf{Y}(x)$  eine geschlossene, formelmäßige Darstellung. Man ermittelt aber eine spezielle partikuläre Lösung  $\mathbf{y}_0(x)$  konkret durch numerische Integration von (9.16) etwa unter der Anfangsbedingung

$$\mathbf{y}_0(a) = \mathbf{0}. \quad (9.19)$$

Die allgemeine Lösung des linearen, inhomogenen Differenzialgleichungssystems (9.16) ist dann gegeben durch

$$\mathbf{y}_{\text{allg}}(x) = \mathbf{y}_0(x) + \sum_{k=1}^n \alpha_k \mathbf{y}_k(x) = \mathbf{y}_0(x) + \mathbf{Y}(x) \boldsymbol{\alpha} \quad (9.20)$$

mit  $\boldsymbol{\alpha} := (\alpha_1, \alpha_2, \dots, \alpha_n)^T \in \mathbb{R}^n$  beliebig. Mit Hilfe dieser allgemeinen Lösung kann jetzt die Lösbarkeit der Randwertaufgabe untersucht werden, indem sie in die geforderten Randbedingungen (9.17) eingesetzt wird. Das liefert die Bedingungsgleichung für den unbekannten Vektor  $\boldsymbol{\alpha}$

$$\mathbf{A} [\mathbf{y}_0(a) + \mathbf{Y}(a) \boldsymbol{\alpha}] + \mathbf{B} [\mathbf{y}_0(b) + \mathbf{Y}(b) \boldsymbol{\alpha}] = \mathbf{c},$$

oder geordnet

$$[\mathbf{A} \mathbf{Y}(a) + \mathbf{B} \mathbf{Y}(b)] \boldsymbol{\alpha} = \mathbf{c} - \mathbf{A} \mathbf{y}_0(a) - \mathbf{B} \mathbf{y}_0(b). \quad (9.21)$$

Die Lösbarkeit des linearen Gleichungssystems (9.21) für den Vektor  $\boldsymbol{\alpha} \in \mathbb{R}^n$  mit der Koeffizientenmatrix  $\mathbf{D} := \mathbf{A} \mathbf{Y}(a) + \mathbf{B} \mathbf{Y}(b) \in \mathbb{R}^{n,n}$  und der rechten Seite  $\mathbf{d} := \mathbf{c} - \mathbf{A} \mathbf{y}_0(a) - \mathbf{B} \mathbf{y}_0(b) \in \mathbb{R}^n$  entscheidet über die Lösbarkeit der linearen Randwertaufgabe. Auf Grund von Sätzen der linearen Algebra [Bun 95, Sta 94] folgt unmittelbar

- Satz 9.1.** a) Die lineare Randwertaufgabe (9.16), (9.17) besitzt eine eindeutige Lösung genau dann, wenn die Matrix  $\mathbf{D} := \mathbf{A} \mathbf{Y}(a) + \mathbf{B} \mathbf{Y}(b)$  regulär ist, d.h. wenn  $\text{Rang } (\mathbf{D}) = n$  ist.  
 b) Die lineare Randwertaufgabe (9.16), (9.17) hat eine mehrdeutige Lösung genau dann, wenn  $\text{Rang } (\mathbf{D}) = \text{Rang } (\mathbf{D}| \mathbf{d}) < n$  gilt.  
 c) Die lineare Randwertaufgabe (9.16), (9.17) besitzt keine Lösung genau dann, wenn  $\text{Rang } (\mathbf{D}) < \text{Rang } (\mathbf{D}| \mathbf{d})$  ist.

Man kann übrigens leicht zeigen, dass die Aussage von Satz 9.1 unabhängig ist von der verwendeten Fundamentalmatrix  $\mathbf{Y}(x)$  und der partikulären Lösung  $\mathbf{y}_0(x)$ . Im Fall von nichtlinearen Randwertaufgaben ist es wesentlich schwieriger, Bedingungen für die Existenz und Eindeutigkeit einer Lösung anzugeben, wie schon an Beispiel 9.2 zu sehen war.

### 9.2.2 Analytische Methoden

In manchen Aufgabenstellungen der Ingenieurwissenschaften treten lineare Randwertaufgaben mit einer einzelnen Differenzialgleichung  $n$ -ter Ordnung auf. Unter der Annahme, die gestellte Randwertaufgabe besitze eine eindeutige Lösung, betrachten wir die Aufgabe, in einem gegebenen Intervall  $I := [a, b]$  eine Funktion  $y(x)$  als Lösung einer linearen, inhomogenen Differenzialgleichung

$$L[y] := \sum_{i=0}^n f_i(x)y^{(i)}(x) = g(x) \quad (9.22)$$

unter den  $n$  allgemeinen Randbedingungen

$$r_i(y) := \sum_{j=1}^n a_{ij}y^{(j-1)}(a) + \sum_{j=1}^n b_{ij}y^{(j-1)}(b) = c_i, \quad i = 1, 2, \dots, n, \quad (9.23)$$

zu bestimmen mit stetigen Funktionen  $f_i, g \in C([a, b])$  und mit  $f_n(x) \neq 0$  für alle  $x \in [a, b]$ .

Eine erste Methode setzt die theoretischen Betrachtungen, die zur Aussage des Satzes 9.1 geführt haben, in die Praxis um. Die allgemeine Lösung der Differenzialgleichung (9.22) stellt sich dar als Linearkombination

$$y(x) = y_0(x) + \sum_{k=1}^n \alpha_k y_k(x), \quad (9.24)$$

einer partikulären Lösung  $y_0(x)$  der inhomogenen Differenzialgleichung  $L[y] = g$  und den  $n$  Funktionen  $y_k(x)$ , welche ein Fundamentalsystem der homogenen Differenzialgleichung  $L[y] = 0$  bilden. Für einfache Differenzialgleichungen (9.22) kann das System der  $(n+1)$  Funktionen oft formelmäßig angegeben werden. Im allgemeinen Fall gewinnt man diese Funktionen näherungsweise durch numerische Integration des zu (9.22) äquivalenten Differenzialgleichungssystems erster Ordnung beispielsweise unter den Anfangsbedingungen

$$\begin{aligned} y_0(a) &= y'_0(a) = \dots = y_0^{(n-1)}(a) = 0, \\ y_k^{(j)}(a) &= \begin{cases} 1 & \text{für } k = j+1 \\ 0 & \text{für } k \neq j+1 \end{cases} \quad k = 1, 2, \dots, n, \\ &\quad j = 0, 1, \dots, n-1. \end{aligned} \quad (9.25)$$

Bei diesem Vorgehen ist festzuhalten, dass die numerische Integration des Differentialgleichungssystems automatisch die Ableitungen der Funktionen bis zur  $(n - 1)$ -ten Ordnung mitliefert, so dass die Werte  $y_k(b)$ ,  $y'_k(b)$ ,  $\dots$ ,  $y_k^{(n-1)}(b)$ ,  $k = 0, 1, 2, \dots, n$ , zur Verfügung stehen. Bildet man mit den  $n$  Funktionen des Fundamentalsystems die *Wronsky-Matrix*

$$\mathbf{Y}(x) := \begin{pmatrix} y_1(x) & y_2(x) & \cdots & y_n(x) \\ y'_1(x) & y'_2(x) & \cdots & y'_n(x) \\ \vdots & \vdots & & \vdots \\ y_1^{(n-1)}(x) & y_2^{(n-1)}(x) & \cdots & y_n^{(n-1)}(x) \end{pmatrix},$$

dann resultiert für die unbekannten Koeffizienten  $\alpha_k$  in (9.24) das lineare Gleichungssystem (9.21). Erfüllen die  $(n + 1)$  Funktionen  $y_0(x)$ ,  $y_1(x)$ ,  $\dots$ ,  $y_n(x)$  die Anfangsbedingungen (9.25), so lautet wegen  $\mathbf{Y}(a) = \mathbf{I}$  das lineare Gleichungssystem für  $\boldsymbol{\alpha} := (\alpha_1, \alpha_2, \dots, \alpha_n)^T$

$$[\mathbf{A} + \mathbf{B} \mathbf{Y}(b)]\boldsymbol{\alpha} = \mathbf{c} - \mathbf{B} \mathbf{z}_0(b) \quad (9.26)$$

mit  $\mathbf{z}_0(b) := (y_0(b), y'_0(b), \dots, y_0^{(n-1)}(b))^T$ . Daraus ist ersichtlich, dass zur Bestimmung der Koeffizienten  $\alpha_k$  nur die Werte der partikulären Lösung  $y_0(x)$  und die Fundamentallösungen  $y_1(x), y_2(x), \dots, y_n(x)$  zusammen mit ihren ersten  $(n - 1)$  Ableitungen an der Stelle  $x = b$  benötigt werden. Nach erfolgter Aufstellung des Gleichungssystems (9.26) und seiner Auflösung steht im Vektor  $\boldsymbol{\alpha}$  die Information über die Anfangsbedingungen der gesuchten Lösungsfunktion  $y(x)$  der Randwertaufgabe zur Verfügung, denn wegen (9.24) und (9.25) gelten für sie

$$y(a) = \alpha_1, \quad y'(a) = \alpha_2, \quad \dots, \quad y^{(n-1)}(a) = \alpha_n.$$

Auf Grund dieser so festgelegten Anfangsbedingungen kann die Lösungsfunktion  $y(x)$  durch eine erneute Integration des Differentialgleichungssystems ermittelt werden. Nach durchgeführter Integration ist die Richtigkeit der Ergebnisse anhand der Randbedingungen überprüfbar. Beim skizzierten Lösungsweg müssen nur die rechten Randwerte der  $(n + 1)$  numerisch berechneten Funktionen  $y_0(x), y_1(x), \dots, y_n(x)$  gespeichert werden.

**Beispiel 9.4.** Wir betrachten die lineare Randwertaufgabe

$$y'' - xy' + 4y = x \quad (9.27)$$

für das Intervall  $[0, 1]$  unter den Randbedingungen

$$y(0) = 1, \quad y(1) = 0. \quad (9.28)$$

Die äquivalente, für die numerische Behandlung geeignete Formulierung als Differentialgleichungssystem lautet mit den Funktionen  $z_1(x) := y(x)$ ,  $z_2(x) := y'(x)$

$$\begin{aligned} z'_1(x) &= z_2(x), \\ z'_2(x) &= -4z_1(x) + xz_2(x) + x, \end{aligned} \quad \text{oder: } \mathbf{z}' = \mathbf{F}\mathbf{z} + \mathbf{g} \text{ mit } \mathbf{F} = \begin{pmatrix} 0 & 1 \\ -4 & x \end{pmatrix}, \quad \mathbf{g} = \begin{pmatrix} 0 \\ x \end{pmatrix} \quad (9.29)$$

mit den Randbedingungen

$$z_1(0) = 1, \quad z_1(1) = 0. \quad (9.30)$$

Für die Vektorfunktion  $\mathbf{z}(x) := (z_1(x), z_2(x))^T$  erhalten die Randbedingungen die allgemeine Form (9.17), wenn wir die folgenden Matrizen  $\mathbf{A}$  und  $\mathbf{B}$  und den Vektor  $\mathbf{c}$  definieren:

$$\mathbf{A} := \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad \mathbf{B} := \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \quad \mathbf{c} := \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Durch numerische Integration des homogenen Differenzialgleichungssystems

$$z'_1(x) = z_2(x), \quad z'_2(x) = -4z_1(x) + xz_2(x)$$

unter den beiden Anfangsbedingungen  $z^1(0) = e_1$ ,  $z^2(0) = e_2$  und des gegebenen inhomogenen Systems (9.29) unter der Anfangsbedingung  $z_0(0) = 0$  ergeben sich mit dem klassischen Runge-Kutta-Verfahren (8.44) bei einer Schrittweite  $h = 0.02$  die folgenden Ergebnisse

$$\mathbf{Z}(1) = \begin{pmatrix} -0.6666666 & 0.5256212 \\ -2.6666666 & -0.3705970 \end{pmatrix}, \quad z_0(1) = \begin{pmatrix} 0.1581263 \\ 0.4568657 \end{pmatrix}.$$

Das daraus resultierende lineare Gleichungssystem (9.26) für die Koeffizienten  $\alpha_1$  und  $\alpha_2$

$$\begin{array}{rcl} \alpha_1 & = 1 \\ -0.6666666 \alpha_1 + 0.5256212 \alpha_2 & = -0.1581262 \end{array}$$

hat die Lösungen  $\alpha_1 = 1$ ,  $\alpha_2 = 0.9675037$ . Die nachfolgende Integration von (9.29) unter der Anfangsbedingung  $z(0) = \alpha$  liefert die gesuchte Lösung  $y(x)$  der Randwertaufgabe. In Abb. 9.3 ist ihr Graph zusammen mit denen der partikulären Lösung  $\tilde{y}_0(x)$  und der Fundamentalslösungen  $y_1(x)$  und  $y_2(x)$  dargestellt, und die Lösung ist für spätere Vergleichszwecke auszugsweise an diskreten Stellen tabelliert.

Der Aufwand zur Lösung der betrachteten Randwertaufgabe hätte verkleinert werden können, wenn die gesuchte Lösung als Superposition einer partikulären Lösung  $\tilde{y}_0(x)$  unter den Anfangsbedingungen  $\tilde{y}_0(0) = 1$ ,  $\tilde{y}'_0(0) = 0$  und einer Lösung  $\tilde{y}_1(x)$  der homogenen Differenzialgleichung unter den Anfangsbedingungen  $\tilde{y}_1(0) = 0$ ,  $\tilde{y}'_1(0) = 1$  angesetzt worden wäre

$$y(x) = \tilde{y}_0(x) + \alpha \tilde{y}_1(x).$$

Für beliebiges  $\alpha$  erfüllt die Ansatzfunktion die Randbedingung am linken Rand des Intervalls und natürlich die Differenzialgleichung. Der Koeffizient  $\alpha$  wird auf Grund der zweiten Randbedingung bestimmt als Lösung einer linearen Gleichung.

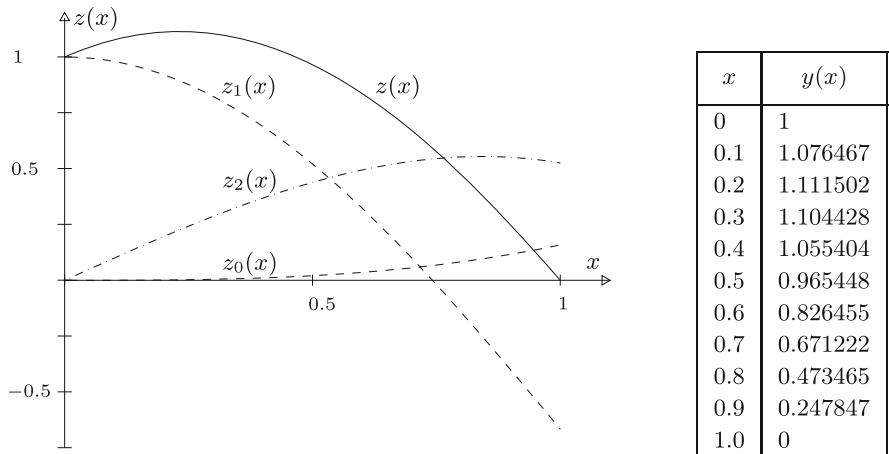


Abb. 9.3 Lösung einer linearen Randwertaufgabe.

△

### 9.2.3 Analytische Methoden mit Funktionenansätzen

In einer anderen Klasse von analytischen, weniger aufwändigen Methoden wird das Ziel gesetzt, wenigstens eine Näherungslösung der Randwertaufgabe (9.22), (9.23) zu bestimmen. Die Grundidee besteht darin, eine Ansatzfunktion als Linearkombination von vorgegebenen, dem Problem angepassten Funktionen zu verwenden, welche in der Regel keine Lösungen der Differenzialgleichung sind. An die Ansatzfunktionen  $w_k(x)$ ,  $k = 0, 1, 2, \dots, m$ , im Ansatz

$$z(x) = w_0(x) + \sum_{k=1}^m \alpha_k w_k(x) \quad (9.31)$$

werden aber Bedingungen hinsichtlich der Randbedingungen gestellt. So soll  $w_0(x)$  die gegebenen Randbedingungen

$$r_i(w_0) = c_i, \quad i = 1, 2, \dots, n, \quad (9.32)$$

erfüllen, während die anderen  $m$  Funktionen  $w_k(x)$  den homogenen Randbedingungen

$$r_i(w_k) = 0, \quad i = 1, 2, \dots, n; \quad k = 1, 2, \dots, m, \quad (9.33)$$

genügen. Durch diese Postulate wird erreicht, dass die Funktion  $z(x)$  (9.31) kraft der Linearität der Randbedingungen für beliebige Werte der  $\alpha_k$  den Randbedingungen

$$r_i(z) = r_i \left( w_0 + \sum_{k=1}^m \alpha_k w_k \right) = r_i(w_0) + \sum_{k=1}^m \alpha_k r_i(w_k) = r_i(w_0) = c_i$$

für  $i = 1, 2, \dots, n$  genügt. Die Koeffizienten  $\alpha_k$  sollen im Folgenden so bestimmt werden, dass die Differenzialgleichung von der Funktion  $z(x)$  in einem noch zu präzisierenden Sinn möglichst gut erfüllt wird. Setzt man den Ansatz für  $z(x)$  in die Differenzialgleichung (9.22) ein, so resultiert eine *Fehlerfunktion*

$$\varepsilon(x; \alpha_1, \alpha_2, \dots, \alpha_m) := L[z] - g(x) = \sum_{k=1}^m \alpha_k L[w_k] + L[w_0] - g(x), \quad (9.34)$$

welche infolge der Linearität der Differenzialgleichung eine lineare Funktion bezüglich der Koeffizienten  $\alpha_k$  ist. Diese werden auf Grund einer der folgenden Methoden bestimmt.

#### Kollokationsmethode

Es wird gefordert, dass die Ansatzfunktion  $z(x)$  die Differenzialgleichung an  $m$  verschiedenen Stellen  $a \leq x_1 < x_2 < \dots < x_{m-1} < x_m \leq b$  des Intervalls erfüllt. Diese  $m$  diskreten Intervallstellen bezeichnet man als *Kollokationspunkte*. Das Postulat führt somit auf die  $m$  Bedingungen  $\varepsilon(x_i; \alpha_1, \alpha_2, \dots, \alpha_m) = 0$  an die Fehlerfunktion, und folglich müssen die  $\alpha_k$  Lösung des linearen Gleichungssystems

$$\sum_{k=1}^m L[w_k]_{x_i} \alpha_k + L[w_0]_{x_i} - g(x_i) = 0, \quad i = 1, 2, \dots, m, \quad (9.35)$$

sein. Die Matrixelemente des Gleichungssystems sind gleich den Differenzialausdrücken  $L[w_k]$  der Ansatzfunktionen  $w_k$ , ausgewertet an den Kollokationspunkten  $x_i$ , und in die rechte Seite des Systems gehen die Ansatzfunktion  $w_0$  und der Wert der Funktion  $g$  ein.

Die Auswertung aller Ausdrücke  $L[w_k]_{x_i}$  scheint wenig attraktiv zu sein. Die Methode eignet sich aber sehr gut zur Lösung von linearen Randwertaufgaben, bei denen die Funktionen  $f_i(x)$  der Differenzialgleichung (9.22) einfache Polynome sind. Dann wird man auch für die Ansatzfunktionen geeignete Polynome verwenden, so dass die Bildung von  $L[w_k]$  zwar viele, aber einfache Operationen erfordert.

### Teilintervallmethode

Entsprechend der Zahl  $m$  der Ansatzfunktionen  $w_k(x)$  wird das Intervall  $[a, b]$  in  $m$  Teilintervalle durch die Punkte  $a = x_0 < x_1 < x_2 < \dots < x_{m-1} < x_m = b$  unterteilt. Dann wird gefordert, dass der Mittelwert der Fehlerfunktion bezüglich eines jeden Teilintervalls gleich null ist. Diese Bedingung führt auf das lineare Gleichungssystem

$$\begin{aligned} & \int_{x_{i-1}}^{x_i} \varepsilon(x; \alpha_1, \alpha_2, \dots, \alpha_m) dx \\ &= \sum_{k=1}^m \alpha_k \int_{x_{i-1}}^{x_i} L[w_k] dx + \int_{x_{i-1}}^{x_i} \{L[w_0] - g(x)\} dx = 0, \quad i = 1, 2, \dots, m. \end{aligned} \tag{9.36}$$

Oft können die darin auftretenden  $m(m+1)$  Integrale über die Teilintervalle analytisch – z.B. mit einem Computer-Algebra-System – berechnet werden. Ist dies nicht der Fall, so sollten sie mit Hilfe der Gauß-Quadratur (vgl. Abschnitt 7.4) numerisch ermittelt werden.

### Fehlerquadratmethode

Es wird zwischen der kontinuierlichen und der diskreten Methode unterschieden ganz entsprechend der Gauß-Approximation, vgl. Abschnitt 3.6. Die Berechnung der komplizierter aufgebauten Integrale bei der kontinuierlichen Methode macht diese wenig attraktiv. Aus diesem Grund wird die *diskrete Fehlerquadratmethode* vorgezogen, die verlangt, dass die Summe der Quadrate der Fehlerfunktionen an  $M > m$  verschiedenen Stellen  $x_i \in [a, b]$ ,  $i = 1, 2, \dots, M$ , minimal wird.

$$\sum_{i=1}^M \varepsilon^2(x_i; \alpha_1, \alpha_2, \dots, \alpha_m) = \min! \quad x_i \in [a, b] \tag{9.37}$$

Diese Forderung ist aber mit der Aufgabe äquivalent, das überbestimmte Gleichungssystem

$$\sum_{k=1}^m \alpha_k L[w_k]_{x_i} = g(x_i) - L[w_0]_{x_i}, \quad i = 1, 2, \dots, M,$$

nach der Methode der kleinsten Quadrate zu behandeln, vgl. Abschnitte 6.1 und 6.2. Die Koeffizienten  $c_{ik} = L[w_k]_{x_i}$  der Fehlergleichungsmatrix  $\mathbf{C} = (c_{ik}) \in \mathbb{R}^{M,m}$  sind gleich den Differenzialausdrücken der Ansatzfunktionen  $w_k$ , ausgewertet an den Stellen  $x_i$  und die Komponenten  $d_i = g(x_i) - L[w_0]_{x_i}$  des Vektors  $\mathbf{d} = (d_1, d_2, \dots, d_M)^T \in \mathbb{R}^m$  berechnen sich auf analoge Weise. Das Fehlergleichungssystem wird vorteilhafter Weise mit einer der Methoden der Orthogonaltransformation bearbeitet.

### Galerkin-Methode

Für  $m$  Ansatzfunktionen  $w_k(x)$  soll die Fehlerfunktion  $\varepsilon(x; \alpha_1, \dots, \alpha_m)$  orthogonal zu einem linearen,  $m$ -dimensionalen Unterraum  $U := \text{span}\{v_1, \dots, v_m\}$  von linear unabhängigen Funktionen  $v_i : [a, b] \rightarrow \mathbb{R}$  sein:

$$\int_a^b \varepsilon(x; \alpha_1, \alpha_2, \dots, \alpha_m) v_i(x) dx = 0, \quad i = 1, 2, \dots, m. \quad (9.38)$$

Das liefert ein System von  $m$  linearen Gleichungen für die  $\alpha_k$

$$\sum_{k=1}^m \alpha_k \int_a^b L[w_k] v_i(x) dx = \int_a^b \{g(x) - L[w_0]\} v_i(x) dx, \quad i = 1, 2, \dots, m. \quad (9.39)$$

Auch diese Methode erfordert die Berechnung von  $m(m + 1)$  Integralen.

In der Regel werden in der Galerkin-Methode die Funktionen  $v_i = w_i$ ,  $i = 1, 2, \dots, m$ , gewählt, so dass die Fehlerfunktion zum linearen Unterraum  $U = \text{span}\{w_1, w_2, \dots, w_m\}$  orthogonal wird, der durch die gewählten Ansatzfunktionen aufgespannt wird.

Eine moderne Weiterentwicklung der Galerkin-Methode führte zur vielseitig anwendbaren *Methode der finiten Elemente*. Das gegebene Intervall  $[a, b]$  wird zuerst in Teilintervalle unterteilt, und dann werden die Ansatzfunktionen  $w_k$  so festgesetzt, dass sie bezüglich der Intervallzerlegung nur einen relativ kleinen Träger besitzen, d.h. nur in wenigen, aneinanderliegenden Teilintervallen von null verschiedene Werte aufweisen. Üblicherweise wird  $v_i = w_i$ ,  $i = 1, 2, \dots, m$ , gewählt, und auf die Integrale wird teilweise partielle Integration angewandt, um die höchsten Ableitungen zu eliminieren. Auf Grund all dieser Maßnahmen bekommt das Gleichungssystem (9.39) Bandstruktur mit sehr kleiner Bandbreite. Für die Behandlung eines selbstadjungierten Sturm-Problems (9.7) ist die Bandmatrix symmetrisch positiv definit; sonst ist sie i.A. nicht symmetrisch. Ist die Differenzialgleichung von zweiter Ordnung, dann sind Ansatzfunktionen zulässig, die stetig und mindestens stückweise einmal stetig differenzierbar sind. Die einfachsten derartigen Funktionen sind die stückweise linearen Hutfunktionen, wie wir sie von den B-Splines kennen, siehe etwa Abb. 3.8.

**Beispiel 9.5.** Wir betrachten die lineare Randwertaufgabe von Beispiel 9.4

$$\begin{aligned} y'' - xy' + 4y &= x \\ y(0) &= 1, \quad y(1) = 0. \end{aligned}$$

Zur Illustration einiger der Näherungsmethoden wählen wir die Funktion

$$w_0(x) = 1 - x,$$

welche die gegebenen Randbedingungen erfüllt, und ferner

$$w_k(x) = x^k(1 - x), \quad k = 1, 2, \dots, m,$$

welche den homogenen Randbedingungen  $y(0) = y(1) = 0$  genügen. Für die benötigten Differenzialausdrücke  $L[w_k] = w_k'' - xw_k' + 4w_k$  ergeben sich nach einfacher Rechnung

$$\begin{aligned} L[w_0] &= 4 - 3x, \\ L[w_1] &= -2x^2 + 3x - 2, \quad L[w_2] = -x^3 + 2x^2 - 6x + 2, \end{aligned}$$

$$L[w_k] = (k-3)x^{k+1} + (4-k)x^k - k(k+1)x^{k-1} + k(k-1)x^{k-2}, \quad k \geq 2.$$

Mit diesen Ausdrücken hat die Fehlerfunktion im Fall  $m = 4$  die explizite Darstellung

$$\varepsilon(x; \alpha_1, \alpha_2, \alpha_3, \alpha_4) = \alpha_1 L[w_1] + \alpha_2 L[w_2] + \alpha_3 L[w_3] + \alpha_4 L[w_4] + L[w_0] - x.$$

Für die Kollokationsmethode soll der Einfluss der gewählten Kollokationspunkte aufgezeigt werden. Für die vier äquidistanten Kollokationspunkte  $x_1 = 0.2, x_2 = 0.4, x_3 = 0.6, x_4 = 0.8$  berechnen sich die Unbekannten  $\alpha_k$  aus dem Gleichungssystem

$$\begin{pmatrix} -1.48 & 0.872 & 0.728 & 0.32032 \\ -1.12 & -0.144 & 0.544 & 0.65024 \\ -0.92 & -1.096 & -0.504 & 0.07776 \\ -0.88 & -2.032 & -2.368 & -2.23232 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix} = \begin{pmatrix} -3.2 \\ -2.4 \\ -1.6 \\ -0.8 \end{pmatrix}$$

zu  $\alpha_1 = 1.967613, \alpha_2 = -0.033081, \alpha_3 = -0.347925, \alpha_4 = -0.018094$ . Die daraus resultierende Näherungslösung der Randwertaufgabe lautet somit

$$z(x) = (1-x)(1+1.967613x - 0.033081x^2 - 0.347925x^3 - 0.018094x^4).$$

Die zugehörige Fehlerfunktion berechnet sich daraus zu

$$\varepsilon(x; \alpha_k) = -0.001388 + 0.013775x - 0.043416x^2 + 0.047036x^3 - 0.018094x^5.$$

In Tab. 9.1 sind die Werte der Näherungslösung zusammen mit denjenigen der Fehlerfunktion tabelliert. Die tabellierten Näherungswerte stellen sehr gute Approximationen dar und weichen um höchstens  $1 \cdot 10^{-5}$  von den exakten Werten ab. Die Fehlerfunktion  $\varepsilon(x; \alpha_k)$  weist aber einen wenig ausgeglichenen Verlauf auf.

Tab. 9.1 Näherungslösungen der Randwertaufgabe.

	Kollokation, $m = 4$ äquidistant		Kollokation, $m = 4$ T-Abszissen		Fehlerquadrat- methode	
$x =$	$z(x) =$	$10^6 \varepsilon =$	$\tilde{z}(x) =$	$10^6 \tilde{\varepsilon} =$	$\bar{z}(x) =$	$10^6 \bar{\varepsilon} =$
0	1.000000	-1388	1.000000	-286	1.000000	-242
0.1	1.076473	-398	1.076467	253	1.076464	313
0.2	1.111510	1	1.111505	265	1.111500	324
0.3	1.104436	63	1.104436	24	1.104430	72
0.4	1.055413	0	1.055416	-238	1.055410	-208
0.5	0.965457	-40	0.965462	-358	0.965456	-344
0.6	0.836465	0	0.836468	-258	0.836462	-254
0.7	0.671231	73	0.671230	29	0.671224	34
0.8	0.473474	-1	0.473467	337	0.473463	361
0.9	0.247855	-553	0.247847	350	0.247844	413
1.0	0.000000	-2087	0.000000	-428	0.000000	-300

Wählt man hingegen als Kollokationspunkte die auf das Intervall  $[0, 1]$  transformierten, aufsteigend angeordneten Nullstellen des Tschebyscheff-Polynoms  $T_4(x)$

$$x_1 = 0.038060, \quad x_2 = 0.308658, \quad x_3 = 0.691342, \quad x_4 = 0.961940,$$

so lautet das zugehörige Gleichungssystem

$$\begin{pmatrix} -1.888717 & 1.774482 & 0.211032 & 0.016280 \\ -1.264566 & 0.309186 & 0.738117 & 0.557923 \\ -0.881882 & -1.522574 & -1.256964 & -0.715215 \\ -0.964837 & -2.811093 & -4.442192 & -5.874624 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix} = \begin{pmatrix} -3.847760 \\ -2.765368 \\ -1.234632 \\ -0.152240 \end{pmatrix}.$$

Die resultierende Näherungslösung ist

$$\tilde{z}(x) = (1-x)(1+1.967508x - 0.032635x^2 - 0.348222x^3 - 0.018295x^4)$$

mit der Fehlerfunktion

$$\tilde{\varepsilon}(x; \alpha_k) = -0.000286 + 0.009002x - 0.041162x^2 + 0.050313x^3 + 0.018295x^5.$$

Die Fehlerfunktion ist jetzt praktisch nivelliert, die Nährungswerte weisen auch etwas kleinere Fehler auf (vgl. Tab. 9.1).

Die diskrete Fehlerquadratmethode wird mit den neun äquidistanten Stellen  $x_i = (i-1)/8$ ,  $i = 1, \dots, 9$ , durchgeführt. Das Fehlergleichungssystem mit neun Fehlergleichungen für die vier Unbekannten  $\alpha_1, \alpha_2, \alpha_3, \alpha_4$  liefert die Näherungslösung

$$\bar{z}(x) = (1-x)(1+1.967479x - 0.032642x^2 - 0.348169x^3 - 0.018344x^4)$$

mit der Fehlerfunktion

$$\bar{\varepsilon}(x; \alpha_k) = -0.000242 + 0.009272x - 0.042342x^2 + 0.051353x^3 - 0.018344x^5.$$

Die sich aus dieser Näherungslösung ergebenden Approximationen sind ebenfalls sehr gut, und die Fehlerfunktion ist recht gleichmäßig nivelliert (vgl. Tab. 9.1).  $\triangle$

## 9.3 Schießverfahren

Die analytischen Verfahren sind für Randwertaufgaben, denen eine nichtlineare Differenzialgleichung zu Grunde liegt, nicht anwendbar, denn ihre allgemeine Lösung ist in diesem Fall nicht linear von den freien Parametern abhängig. Die Grundidee der Schießverfahren besteht darin, die Behandlung von nichtlinearen Randwertaufgaben auf Anfangswertprobleme zurückzuführen, für welche die Anfangswerte so zu bestimmen sind, dass die Lösung der Randwertaufgabe resultiert.

### 9.3.1 Das Einfach-Schießverfahren

Das Prinzip lässt sich am einfachsten anhand eines Differentialgleichungssystems erster Ordnung für zwei Funktionen  $y_1(x), y_2(x)$

$$\mathbf{y}'(x) = \mathbf{f}(x, \mathbf{y}(x)), \quad \mathbf{y}(x) = \begin{pmatrix} y_1(x) \\ y_2(x) \end{pmatrix} \quad (9.40)$$

unter den Randbedingungen

$$r_1(y_1(a), y_2(a)) = 0, \quad r_2(y_1(b), y_2(b)) = 0 \quad (9.41)$$

beschreiben. Zur Vereinfachung wird angenommen, dass die Randbedingungen je nur die Werte der beiden Funktionen in den Randpunkten betreffen. Um die Aufgabenstellung als

Anfangswertaufgabe bearbeiten zu können, sind an der Stelle  $a$  die Werte von  $y_1(a)$  und  $y_2(a)$  vorzugeben. Die Randbedingung  $r_1(y_1(a), y_2(a)) = 0$  bestimmt entweder den einen der beiden Werte oder stellt eine Relation zwischen ihnen her. Jedenfalls kann man den einen Wert problemgerecht gleich einem Wert  $s$  setzen und den anderen aus der Randbedingung bestimmen. Durch die Vorgabe von  $s$  sind damit die notwendigen Anfangsbedingungen vorhanden, und das Differenzialgleichungssystem kann numerisch im Intervall  $[a, b]$  integriert werden. Die resultierende Lösung ist selbstverständlich vom gewählten Wert  $s$  abhängig, so dass wir sie mit  $\mathbf{y}(x; s)$  bezeichnen wollen. Der Parameter  $s$  ist nun so zu bestimmen, dass die zweite Randbedingung

$$F(s) := r_2(y_1(b; s), y_2(b; s)) = 0 \quad (9.42)$$

erfüllt ist. Dies stellt eine nichtlineare Gleichung für  $s$  dar, welche durch die Lösungsfunktion der Anfangswertaufgabe und die zweite Randbedingung definiert ist. Der Parameter  $s$  ist so festzulegen, dass die zweite Randbedingung erfüllt wird. Da das Vorgehen analog zum Einschießen der Artillerie ist, spricht man vom *Schießverfahren* (engl. shooting). Die nichtlineare Gleichung  $F(s) = 0$  kann mit einer der Methoden des Abschnitts 4.2 gelöst werden, z.B. mit der Sekantenmethode, um Ableitungen von  $F(s)$  nach  $s$  zu vermeiden.

Liegt eine Randwertaufgabe mit  $n$  Differenzialgleichungen erster Ordnung

$$\mathbf{y}'(x) = \mathbf{f}(x, \mathbf{y}(x)) \quad (9.43)$$

mit den  $n$  Randbedingungen

$$r_i(\mathbf{y}(a), \mathbf{y}(b)) = 0, \quad i = 1, 2, \dots, n, \quad (9.44)$$

vor, so wird allgemein ein Parametervektor  $\mathbf{s} \in \mathbb{R}^n$  verwendet, man löst die Anfangswertaufgabe für das Differenzialgleichungssystem (9.43) numerisch unter der Anfangsbedingung  $\mathbf{y}(a) = \mathbf{s}$  und erhält eine von  $\mathbf{s}$  abhängige Lösung  $\mathbf{y}(x; \mathbf{s})$ . Eingesetzt ergeben die Randbedingungen  $n$  nichtlineare Gleichungen

$$F_i(\mathbf{s}) := r_i(\mathbf{y}(a; \mathbf{s}), \mathbf{y}(b; \mathbf{s})) = 0, \quad i = 1, 2, \dots, n, \quad (9.45)$$

für den unbekannten Parametervektor  $\mathbf{s} := (s_1, s_2, \dots, s_n)^T$ . Will man das System  $\mathbf{F}(\mathbf{s}) = \mathbf{0}$  mit der Methode von Newton oder einer ihrer Varianten lösen, werden die partiellen Ableitungen nach den  $s_k$  benötigt. Diese sind in der Regel nur näherungsweise durch numerische Differenziation als Differenzenquotienten zu erhalten. Dabei ist aber zu beachten, dass die Änderung eines Parameters  $s_k$  um  $\Delta s_k$  die Integration des Differenzialgleichungssystems (9.43) erfordert, um die entsprechende Differenzenquotienten berechnen zu können.

Die Dimension des Parametervektors  $\mathbf{s}$  kann in vielen Anwendungen um die Anzahl der Randbedingungen reduziert werden, die nur die Werte der  $n$  Funktionen im linken Randpunkt betreffen. Wir wollen also annehmen, die Randbedingungen lauten wie folgt:

$$r_i(\mathbf{y}(a)) = 0, \quad i = 1, 2, \dots, r, \quad (9.46)$$

$$r_i(\mathbf{y}(b)) = 0, \quad i = r + 1, r + 2, \dots, n. \quad (9.47)$$

Es genügt dann, einen Parametervektor  $\mathbf{s} \in \mathbb{R}^{n-r}$  so zu wählen, dass für  $\mathbf{y}(a; \mathbf{s})$  die Randbedingungen (9.46) erfüllt ist. Die Lösungsfunktion  $\mathbf{y}(x; \mathbf{s})$  muss dann noch die  $(n-r)$  Randbedingungen (9.47) erfüllen. Dies liefert die notwendigen  $(n-r)$  nichtlinearen Gleichungen für  $\mathbf{s}$ .

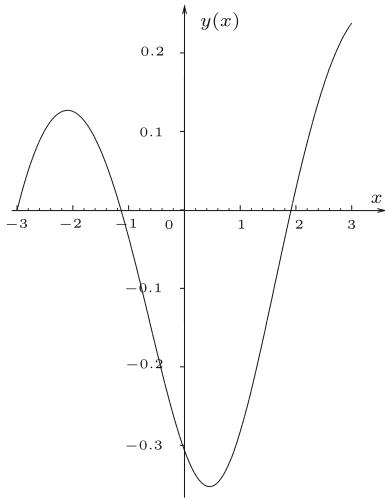


Abb. 9.4  
Lösung der nichtlinearen Randwertaufgabe.

**Beispiel 9.6.** Wir betrachten die nichtlineare Randwertaufgabe

$$y'' + 2y'^2 + 1.5y + 0.5y^2 = 0.05x$$

$$y(-3) = 0, \quad y'(3) - 0.4y(3) = 0.$$

Als System formuliert lautet sie

$$y'_1(x) = y_2(x)$$

$$y'_2(x) = -1.5y_1(x) - 0.5y_1(x)^2 - 2y_2(x)^2 + 0.05x$$

$$r_1(y(-3)) = y_1(-3) = 0, \quad r_2(y(3)) = y_2(3) - 0.4y_1(3) = 0.$$

Für das Schießverfahren wird der Parameter  $s$  als Anfangsbedingung für  $y_2$  eingeführt, es wird also mit den Anfangsbedingungen  $y_1(-3) = 0$ ,  $y_2(-3) = s$  gerechnet. Für die gesuchte Funktion  $y(x) = y_1(x)$  bedeutet dies, dass ihre Anfangssteigung variiert wird. Das Anfangswertproblem ist mit dem klassischen Runge-Kutta-Verfahren vierter Ordnung (8.44) unter Verwendung der Schrittweite  $h = 0.1$  behandelt worden, welche eine hinreichende Genauigkeit der berechneten Näherungslösung garantiert. Für die Folge von Parameterwerten  $s = 0, 0.1, 0.2, 0.3$  ist die Integration durchgeführt und der Funktionswert  $F(s) := y_2(3; s) - 0.4y_1(3; s)$  bestimmt worden. Auf Grund des Vorzeichenwechsels von  $F(s)$  für die Werte  $s = 0.2$  und  $s = 0.3$  ist der gesuchte Wert von  $s$  lokalisierter. Sein Wert ist iterativ mit der Sekantenmethode ermittelt worden. Der Rechengang ist in Tab. 9.2 mit auf sechs Nachkommastellen gerundeten Zahlenwerten zusammengestellt. Die für  $s = 0.281475$  resultierende Lösung ist in Abb. 9.4 dargestellt.  $\triangle$

**Beispiel 9.7.** Die Behandlung der nichtlinearen Randwertaufgabe vierter Ordnung

$$y^{(4)}(x) - (1 + x^2)y''(x)^2 + 5y(x)^2 = 0$$

unter den linearen Randbedingungen

$$y(0) = 1, \quad y'(0) = 0, \quad y''(1) = -2, \quad y^{(3)}(1) = -3$$

erfolgt nach dem skizzierten Vorgehen.

Tab. 9.2 Beispiel 9.6: Sekantenmethode beim Schießverfahren.

$k$	$s^{(k)}$	$y_1(3; s^{(k)})$	$y_2(3; s^{(k)})$	$F(s^{(k)})$
0	0.2	0.214493	0.026286	-0.059511
1	0.3	0.238697	0.119049	0.023570
2	0.271636	0.236654	0.084201	-0.010461
3	0.280357	0.237923	0.093916	-0.001253
4	0.281544	0.238057	0.095300	0.000078
5	0.281475	0.238050	0.095219	$-3.1 \cdot 10^{-7}$
6	<u>0.281475 = s</u>			

Mit den Substitutionen  $y_1(x) = y(x)$ ,  $y_2(x) = y'(x)$ ,  $y_3(x) = y''(x)$ ,  $y_4(x) = y^{(3)}(x)$  lautet die Aufgabe als System von Differenzialgleichungen erster Ordnung

$$\begin{aligned} y'_1(x) &= y_2(x), & y'_2(x) &= y_3(x), & y'_3(x) &= y_4(x) \\ y'_4(x) &= (1+x^2)y_3(x)^2 - 5y_1(x)^2 \end{aligned} \quad (9.48)$$

mit den Randbedingungen

$$y_1(0) = 1, \quad y_2(0) = 0; \quad y_3(1) = -2, \quad y_4(1) = -3.$$

Da hier die Randbedingungen nur die Werte an den Intervallenden betreffen, kann mit zwei Parametern in den Anfangswerten gearbeitet werden:

$$y_1(0) = 1, \quad y_2(0) = 0; \quad y_3(0) = s, \quad y_4(0) = t.$$

Für die zugehörige Lösung  $\mathbf{y}(x; s, t)$  lauten die zu erfüllenden Bedingungsgleichungen

$$g(s, t) := y_3(1; s, t) + 2 = 0$$

$$h(s, t) := y_4(1; s, t) + 3 = 0.$$

Wegen der Nichtlinearität des Differenzialgleichungssystems wird das nichtlineare System von Gleichungen für  $s$  und  $t$  mit der Methode von Newton (vgl. Abschnitt 4.3.2) gelöst. Die Elemente der Funktionalmatrix

$$\Phi(s, t) = \begin{pmatrix} g_s & g_t \\ h_s & h_t \end{pmatrix}_{(s, t)}$$

werden genähert als Differenzenquotienten ermittelt gemäß

$$\frac{\partial g(s, t)}{\partial s} \approx \frac{g(s + \Delta s, t) - g(s, t)}{\Delta s}, \quad \frac{\partial h(s, t)}{\partial s} \approx \frac{h(s + \Delta s, t) - h(s, t)}{\Delta s}.$$

Analoge Formeln gelten für die partiellen Ableitungen nach  $t$ . Um die Differenzenquotienten zu gegebenem Parameterpaar  $(s_k, t_k)$  berechnen zu können, ist das Differenzialgleichungssystem (9.48) pro Iterationsschritt dreimal zu integrieren für die drei Parameterkombinationen  $(s_k, t_k)$ ,  $(s_k + \Delta s_k, t_k)$  und  $(s_k, t_k + \Delta t_k)$ . Die Parameterinkremente  $\Delta s$  und  $\Delta t$  sind problemabhängig geeignet zu wählen. Einerseits dürfen sie nicht zu klein sein, um Auslöschung im Zähler des Differenzenquotienten zu vermeiden, und andererseits nicht zu groß, um den Diskretisierungsfehler in tolerablen Grenzen zu halten. Ihre Wahl beeinflusst die Iterationsfolge dadurch, dass die Näherungen der Funktionalmatrix von den Größen  $\Delta s$  und  $\Delta t$  abhängig sind und somit verschiedene Korrekturen der einzelnen Iterationsschritte bewirken.

Die Anfangswertaufgaben sind mit dem klassischen Runge-Kutta-Verfahren (8.44) unter Verwendung der Schrittweite  $h = 0.025$  numerisch integriert worden. Mit den dem Problem angepassten

Werten  $\Delta s = \Delta t = 0.05$  ergeben sich für die Startwerte  $s_0 = t_0 = 0$  für den ersten Iterationsschritt die folgenden Zahlwerte:

$$\begin{aligned} g(s_0, t_0) &= -0.144231, & h(s_0, t_0) &= 0.196094, \\ g(s_0 + \Delta s, t_0) &= -0.135303, & h(s_0 + \Delta s, t_0) &= 0.046900, \\ g(s_0, t_0 + \Delta t_0) &= -0.113508, & h(s_0, t_0 + \Delta t_0) &= 0.151890. \end{aligned}$$

Das resultierende lineare Gleichungssystem für die Korrekturen  $\sigma$  und  $\tau$  lautet

$$\begin{aligned} 0.178556 \sigma + 0.614459 \tau &= 0.144231 \\ -2.983871 \sigma - 0.884072 \tau &= -0.196094 \end{aligned}$$

mit den Lösungen  $\sigma = -0.004189$ ,  $\tau = 0.235946$ . Die Näherungen für den nächsten Iterationsschritt sind somit  $s_1 = -0.004189$ ,  $t_1 = 0.235946$ . Das zugehörige lineare Gleichungssystem für die Korrekturen im zweiten Iterationsschritt ist

$$\begin{aligned} 0.241491 \sigma + 0.652710 \tau &= -0.003440 \\ -2.893751 \sigma - 0.766208 \tau &= -0.010520 \end{aligned}$$

mit den Lösungen  $\sigma = 0.005577$ ,  $\tau = -0.007333$ , welche die zweiten Näherungen  $s_2 = 0.001388$ ,  $t_2 = 0.228613$  liefern. Die Fortsetzung der Iteration erzeugt die rasch konvergente Parameterfolge  $s_3 = 0.001394$ ,  $t_3 = 0.228712$ ,  $s_4 = 0.001393$ ,  $t_4 = 0.228714$  und damit Parameterwerte für  $s$  und  $t$  in ausreichender Genauigkeit. Mit diesen Werten ergibt sich die gesuchte Lösung der Randwertaufgabe durch eine weitere Integration des Differentialgleichungssystems mit den jetzt vollständig bekannten Anfangsbedingungen.  $\triangle$

Das beschriebene Einfach-Schießverfahren besitzt in bestimmten Fällen von Randwertaufgaben eine katastrophale Schwäche, welche die numerische Bestimmung der Lösung praktisch unmöglich macht. Diese Schwierigkeiten treten dann auf, wenn die Lösung  $\mathbf{y}(x, \mathbf{s})$  empfindlich auf kleine Änderungen des Parametervektors  $\mathbf{s}$  reagiert oder das Lösungsintervall recht groß ist. In diesem Fall sind die Bedingungsgleichungen für  $\mathbf{s}$  schwierig oder nur mit großer Unsicherheit zu lösen.

**Beispiel 9.8.** Die Situation soll anhand des linearen Randwertproblems

$$\begin{aligned} y'' - 2y' - 8y &= 0 \\ y(0) = 1, \quad y(6) &= 1 \end{aligned} \tag{9.49}$$

dargelegt werden, weil an dieser Aufgabenstellung die grundsätzlichen Überlegungen vollständig analytisch durchführbar sind. Die Differentialgleichung besitzt die allgemeine Lösung

$$y(x) = c_1 e^{4x} + c_2 e^{-2x}, \quad c_1, c_2 \in \mathbb{R} \text{ beliebig}.$$

Die nach dem Schießverfahren bestimmte Lösungsschar mit dem Parameter  $s$  zu den Anfangsbedingungen  $y(0) = 1$ ,  $y'(0) = s$  lautet

$$y(x; s) = \frac{1}{6}(2+s)e^{4x} + \frac{1}{6}(4-s)e^{-2x}. \tag{9.50}$$

Die zweite Randbedingung liefert für  $s$

$$\begin{aligned} y(6; s) &= \frac{1}{6}(2+s)e^{24} + \frac{1}{6}(4-s)e^{-12} = 1 \quad \Rightarrow \\ s &= \frac{6 - 2e^{24} - 4e^{-12}}{e^{24} - e^{-12}} \doteq -1.999\,999\,999\,77. \end{aligned}$$

Die Lösung der Randwertaufgabe ist somit gegeben durch

$$y(x) \doteq 3.77511134906 \cdot 10^{-11} \cdot e^{4x} + 0.999999999962 \cdot e^{-2x}.$$

Da die Lösungsschar (9.50) linear von  $s$  abhängt, folgt für die Änderung bei einer Variation von  $s$  um  $\Delta s$  allgemein

$$\Delta y(x) := y(x; s + \Delta s) - y(x; s) = \frac{1}{6} \Delta s \{e^{4x} - e^{-2x}\},$$

und insbesondere am Intervallende  $x = 6$

$$\Delta y(6) = \frac{1}{6} \Delta s \{e^{24} - e^{-12}\} \doteq \Delta s \cdot 4.415 \cdot 10^9.$$

Jede Änderung um  $\Delta s$  verstärkt sich um den Faktor  $4.415 \cdot 10^9$  auf die Änderung des Funktionswertes am Ende des Intervalls. Rechnet man etwa mit zwölf wesentlichen Dezimalstellen und variiert  $s$  in der Gegend von  $-2$  um die kleinstmöglichen Werte  $\Delta s = 10^{-11}$ , so beträgt die Änderung  $\Delta y(6) \approx 0.0442$ . Die Randbedingung  $y(6) = 1$  kann im extremsten Fall nur mit einer Abweichung von  $0.0221$  erfüllt werden.  $\triangle$

### 9.3.2 Das Mehrfach-Schießverfahren

Weil das Empfindlichkeitsmaß von Lösungen auf Änderungen der Parameterwerte auf kurzen Intervallen bedeutend kleiner ist, wird dieser Tatsache beim *Mehrfach-Schießverfahren* (engl. multiple shooting) Rechnung getragen. Dazu wird das gegebene Intervall  $[a, b]$  in Teilintervalle unterteilt, um in jedem Teilintervall Lösungen der Differentialgleichung unter Anfangsbedingungen zu bestimmen, welche von Parametersätzen abhängig sind. Diese Teillösungen werden sodann durch Anpassung der Parametersätze dadurch zur Gesamtlösung der Randwertaufgabe zusammengesetzt, dass einerseits an den Intervalteipunkten Übergangsbedingungen und andererseits die Randbedingungen erfüllt sind. Unter der Annahme, die Randwertaufgabe sei in der Form eines Differentialgleichungssystems erster Ordnung formuliert, betreffen die Übergangsbedingungen allein die Stetigkeit der beteiligten Lösungsfunktionen. Insgesamt führt das zu einem entsprechend großen, nichtlinearen Gleichungssystem für die Parameter.

Das prinzipielle Vorgehen soll an einer konkreten Randwertaufgabe so dargelegt werden, dass die sinngemäße Verallgemeinerung offensichtlich ist. Zu lösen sei

$$\begin{aligned} y'_1(x) &= f_1(x, y_1(x), y_2(x)) \\ y'_2(x) &= f_2(x, y_1(x), y_2(x)) \end{aligned} \tag{9.51}$$

unter den speziellen Randbedingungen

$$r_1(y_1(a), y_2(a)) = 0, \quad r_2(y_1(b), y_2(b)) = 0. \tag{9.52}$$

Das Intervall  $[a, b]$  werde durch die Teipunkte  $a = x_1 < x_2 < x_3 < x_4 = b$  in drei Teilintervalle  $[x_i, x_{i+1}]$ ,  $i = 1, 2, 3$ , zerlegt. In Analogie zum Einfach-Schießverfahren wird im ersten Teilintervall  $[x_1, x_2]$  eine Lösung  $\mathbf{Y}_1(x; s_1) = (Y_{11}(x; s_1), Y_{21}(x; s_1))^T$  von (9.51) bestimmt, die vom Parameter  $s_1$  abhängt. In den anderen Teilintervallen  $[x_i, x_{i+1}]$ ,  $i = 2, 3$ , werde unter den Anfangsbedingungen

$$Y_{1i}(x_i) = y_i, \quad Y_{2i}(x_i) = s_i$$

je die zugehörige Lösung  $\mathbf{Y}_i(x; y_i, s_i) = (Y_{1i}(x; y_i, s_i), Y_{2i}(x; y_i, s_i))^T$  durch numerische Integration berechnet. Unbekannt und somit zu bestimmen sind die Werte der Komponenten des Parametervektors  $\mathbf{s} := (s_1, y_2, s_2, y_3, s_3)^T \in \mathbb{R}^5$ . Die Stetigkeitsbedingungen an den

inneren Teilpunkten  $x_2, x_3$ , sowie die zweite Randbedingung ergeben das Gleichungssystem

$$\begin{aligned} g_1(s_1, y_2) &:= Y_{11}(x_2; s_1) - y_2 = 0, \\ g_2(s_1, s_2) &:= Y_{21}(x_2; s_1) - s_2 = 0, \\ g_3(y_2, s_2, y_3) &:= Y_{12}(x_3; y_2, s_2) - y_3 = 0, \\ g_4(y_2, s_2, s_3) &:= Y_{22}(x_3; y_2, s_2) - s_3 = 0, \\ g_5(y_3, s_3) &:= r_2(Y_{13}(x_4; y_3, s_3), Y_{23}(x_4; y_3, s_3)) = 0. \end{aligned} \quad (9.53)$$

Um das nichtlineare Gleichungssystem (9.53) mit der Methode von Newton zu behandeln, muss die Funktionalmatrix zu einem iterierten Parametervektor

$$\mathbf{s}^{(k)} = (s_1^{(k)}, y_2^{(k)}, s_2^{(k)}, y_3^{(k)}, s_3^{(k)})^T$$

bestimmt werden. Da die einzelnen Gleichungen im betrachteten Fall höchstens von je drei Parametern abhängen, weist die Funktionalmatrix eine sehr spezielle Bandstruktur auf, und die von null verschiedenen Matrixelemente haben überdies sehr spezielle Werte und stehen in engem Zusammenhang mit den Lösungen der Differenzialgleichungen der Teilintervalle. So gelten

$$\begin{aligned} \frac{\partial g_1}{\partial y_2} &= \frac{\partial g_2}{\partial s_2} = \frac{\partial g_3}{\partial y_3} = \frac{\partial g_4}{\partial s_3} = -1 \quad \text{und} \\ \frac{\partial g_1}{\partial s_1} &= \frac{\partial Y_{11}}{\partial s_1}, \quad \frac{\partial g_2}{\partial s_1} = \frac{\partial Y_{21}}{\partial s_1}, \quad \frac{\partial g_3}{\partial y_2} = \frac{\partial Y_{12}}{\partial y_2}, \\ \frac{\partial g_3}{\partial s_2} &= \frac{\partial Y_{12}}{\partial s_2}, \quad \frac{\partial g_4}{\partial y_2} = \frac{\partial Y_{22}}{\partial y_2}, \quad \frac{\partial g_4}{\partial s_2} = \frac{\partial Y_{22}}{\partial s_2}. \end{aligned}$$

Die nichttrivialen partiellen Ableitungen sind im allgemeinen wieder durch Differenzenquotienten zu approximieren, die durch zusätzliche Integrationen des Differenzialgleichungssystems unter varierten Anfangsbedingungen gewonnen werden können. Die Funktionalmatrix besitzt im konkreten Fall den folgenden Aufbau:

$$\Phi = \left( \begin{array}{ccccc} \frac{\partial g_1}{\partial s_1} & -1 & 0 & 0 & 0 \\ \frac{\partial g_2}{\partial s_1} & 0 & -1 & 0 & 0 \\ 0 & \frac{\partial g_3}{\partial y_2} & \frac{\partial g_3}{\partial s_2} & -1 & 0 \\ 0 & \frac{\partial g_4}{\partial y_2} & \frac{\partial g_4}{\partial s_2} & 0 & -1 \\ 0 & 0 & 0 & \frac{\partial r_2}{\partial y_3} & \frac{\partial r_2}{\partial s_3} \end{array} \right). \quad (9.54)$$

Die beiden Elemente der letzten Zeile von  $\Phi$  sind von der konkreten Randbedingung abhängig. Darin treten partielle Ableitungen der Lösungsfunktionen  $Y_{i3}(x_4; y_3, s_3)$  auf.

Zusammenfassend halten wir fest, dass zur genäherten Berechnung von  $\Phi$  das Differenzialgleichungssystem (9.51) im ersten Teilintervall zweimal für die Parameter  $s_1^{(k)}$  und  $s_1^{(k)} + \Delta s_1$  zu integrieren ist, in den weiteren Teilintervallen  $[x_i, x_{i+1}]$  jeweils dreimal für die Parameterkombinationen  $(y_i^{(k)}, s_i^{(k)}), (y_i^{(k)} + \Delta y_i, s_i^{(k)}), (y_i^{(k)}, s_i^{(k)} + \Delta s_i)$ . Auf diese Weise steigt der

Rechenaufwand sicher an, doch ist zu beachten, dass die numerische Integration nur je über die Teilintervalle zu erfolgen hat, insgesamt also nur über das ganze Intervall. Weil zudem die Empfindlichkeit der Lösungsfunktionen auf Änderungen der Anfangswerte auf den Teilintervallen geringer ist als auf dem ganzen Intervall, ist diese Problematik eliminiert, und die nichtlinearen Gleichungssysteme lassen sich weitgehend problemlos lösen.

**Beispiel 9.9.** Wir betrachten die wenig problematische nichtlineare Randwertaufgabe von Beispiel 9.6

$$\begin{aligned} y'_1(x) &= y_2(x), \\ y'_2(x) &= -1.5 y_1(x) - 0.5 y_1(x)^2 - 2 y_2(x)^2 + 0.05 x, \\ y_1(-3) &= 0, \quad y_2(3) - 0.4 y_1(3) = 0. \end{aligned} \quad (9.55)$$

Wir unterteilen das Intervall  $[-3, 3]$  in die drei gleich langen Teilintervalle  $[-3, -1], [-1, 1], [1, 3]$ . Der Parametervektor ist deshalb  $\mathbf{s} = (s_1, y_2, s_2, y_3, s_3)^T \in \mathbb{R}^5$ , und für ihn ist das nichtlineare Gleichungssystem (9.53)

$$\begin{aligned} g_1(s_1, y_2) &= Y_{11}(-1; s_1) - y_2 &= 0 \\ g_2(s_1, s_2) &= Y_{21}(-1; s_1) - s_2 &= 0 \\ g_3(y_2, s_2, y_3) &= Y_{12}(1; y_2, s_2) - y_3 &= 0 \\ g_4(y_2, s_2, s_3) &= Y_{22}(1; y_2, s_2) - s_3 &= 0 \\ g_5(y_3, s_3) &= Y_{23}(3; y_3, s_3) - 0.4 Y_{13}(3; y_3, s_3) &= 0 \end{aligned}$$

zu lösen. Ein Problem zu seiner Lösung besteht wohl darin, der Aufgabenstellung angepasste Startwerte des Parametervektors  $\mathbf{s}^{(0)}$  vorzugeben, damit die Methode von Newton eine konvergente, in möglichst wenigen Iterationen zur Lösung führende Folge liefert. Auf dieses Problem werden wir unten noch eingehen. Zur Berechnung der Koeffizientenmatrix (9.54) des linearen Gleichungssystems zum Newton-Verfahren ist das Differenzialgleichungssystem (9.55) insgesamt achtmal in den Teilintervallen zu lösen. Mit dem guten Startvektor  $\mathbf{s}^{(0)} = (0.28, 0, -0.25, -0.28, 0.24)^T$  und den Parameterinkrementen  $\Delta = 0.01$  lautet dann das lineare Gleichungssystem für den Korrekturvektor  $\boldsymbol{\sigma} = (\sigma_1, \eta_2, \sigma_2, \eta_3, \sigma_3)^T$

$$\begin{pmatrix} 0.159218 & -1 & 0 & 0 & 0 \\ -0.891498 & 0 & -1 & 0 & 0 \\ 0 & -2.156751 & 2.366674 & -1 & 0 \\ 0 & -1.251487 & -0.327109 & 0 & -1 \\ 0 & 0 & 0 & -0.536515 & -0.509051 \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \eta_2 \\ \sigma_2 \\ \eta_3 \\ \sigma_3 \end{pmatrix} = \begin{pmatrix} 0.035364 \\ 0.015244 \\ 0.029353 \\ 0.042562 \\ 0.002430 \end{pmatrix}.$$

Seine Lösung  $\boldsymbol{\sigma} = (0.009177, -0.033903, -0.023425, -0.011673, 0.007530)^T$  ergibt den ersten iterierten Parametervektor  $\mathbf{s}^{(1)} = (0.289177, -0.033903, -0.273425, -0.291673, 0.247530)^T$ . Das lineare Gleichungssystem für die zugehörigen Korrekturen

$$\begin{pmatrix} 0.149176 & -1 & 0 & 0 & 0 \\ -0.896325 & 0 & -1 & 0 & 0 \\ 0 & -1.912048 & 2.372381 & -1 & 0 \\ 0 & -0.966532 & -0.410931 & 0 & -1 \\ 0 & 0 & 0 & -0.540519 & -0.500333 \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \eta_2 \\ \sigma_2 \\ \eta_3 \\ \sigma_3 \end{pmatrix} = \begin{pmatrix} 0.000004 \\ 0.000002 \\ 0.009725 \\ 0.007629 \\ 0.009349 \end{pmatrix}$$

liefert  $\boldsymbol{\sigma} = (-0.007654, -0.001138, 0.006863, 0.008732, -0.009349)^T$  und damit die zweite Näherung für den Parametervektor  $\mathbf{s}^{(2)} = (0.281523, -0.035040, -0.266563, -0.282941, 0.238180)^T$ . Ein weiterer Iterationsschritt führt zum Parametervektor

$$\mathbf{s}^{(3)} = (0.281480, -0.035122, -0.266560, -0.282786, 0.238165)^T \doteq \mathbf{s},$$

welcher für die gesuchte Lösung der Randwertaufgabe eine sehr gute Näherung definiert. Das Residuum der Randbedingung am rechten Rand ist kleiner als  $1 \cdot 10^{-6}$ . Das entspricht der Genauigkeit von Beispiel 9.6.  $\triangle$

## Wahl der Zwischenstellen und Startwerte

Die erfolgreiche Durchführung des Mehrfach-Schießverfahrens und des zugehörigen Newton-Verfahrens hängt stark von der Wahl der Zwischenstellen  $x_i$  und Startwerte  $s_i$  und  $y_i$  ab. Hierzu kommen verschiedene Strategien in Frage.

1. Ist der Verlauf der Lösung ungefähr bekannt und das Verfahren nicht empfindlich gegenüber der Wahl der Zwischenstellen und Anfangswerte, dann können diese ‘per Hand’ gewählt werden.

2. *Randwertlinearisierung:*

Spezielle Randbedingungen wie bei dem Problem

$$y'' = f(x, y, y'), \quad (9.56)$$

$$y(a) = \alpha, \quad y(b) = \beta, \quad (9.57)$$

erlauben die Konstruktion einer Hilfsfunktion, und zwar der linearen Verbindung der Randwerte und deren Ableitung:

$$\eta_1(x) = \frac{(\beta - \alpha)}{(b - a)}(x - a) + \alpha,$$

$$\eta_2(x) = \eta'_1(x) = \frac{(\beta - \alpha)}{(b - a)},$$

$$\boldsymbol{\eta} = \begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix}.$$

Diese Hilfsfunktion  $\boldsymbol{\eta}$  kann im Algorithmus Tab. 9.3 eingesetzt werden.

3. Noch effizienter können die Zwischenstellen bestimmt werden, wenn es gelingt, das Problem zu linearisieren und eine *Näherungslösung durch Linearisierung* oder durch Einbettungsmethoden [Sto 05] zu finden. Dann kann diese Näherungslösung mit ihrer Ableitung als Hilfsfunktion  $\boldsymbol{\eta}$  für den Algorithmus Tab. 9.3 gewählt werden.

Für die letzten beiden Möglichkeiten geben wir den Algorithmus Tab. 9.3 und ein Beispiel an. Abhängig von den Eigenschaften der Hilfsfunktion kann die Konstante  $K$  aus Schritt (1) des Algorithmus auch als additive Konstante gewählt werden. Dann werden die zweite und dritte Abfrage in Schritt (4) ersetzt durch

$$\|\mathbf{u}_j - \boldsymbol{\eta}(\tilde{x}_j)\| \geq K.$$

**Beispiel 9.10.** Die Lösung der Randwertaufgabe (siehe auch [Sto 05])

$$y''(x) = 5 \sinh(5y(x)), \quad y(0) = 0, \quad y(1) = 1.$$

ist näherungsweise gleich der Lösung der Anfangswertaufgabe

$$y''(x) = 5 \sinh(5y(x)), \quad y(0) = 0, \quad y'(0) = \bar{s} \quad \text{mit } \bar{s} = 0.0457504614.$$

Dieses  $\bar{s}$  müsste also mit dem (Mehrfach-)Schießverfahren gefunden werden. Das Problem dabei ist aber, dass die Lösung eine logarithmische Singularität bei  $x = 1.03$  hat, die für  $s > 0.05$  in den Integrationsbereich hineinwandert. Deshalb ist das Einfach-Schießverfahren z.B. mit Schießversuchen für  $s = 0.1, 0.2, \dots$  zum Scheitern verurteilt.

Tab. 9.3 Algorithmus zur Bestimmung der Zwischenstellen.

- (1) Bestimme einen Lösungsschlauch.  
Wähle dazu eine Konstante  $K > 1$ , etwa  $K = 2$ , als Faktor.
- (2) Setze  $x_1 := a$ ,  $y_1 := y(a)$ ,  $s_1 := \eta_2(a)$  und  $i := 1$ .
- (3) Schiesse von  $x_i$  mit den Startwerten  $(y_i, s_i)$  so lange, bis (4) eintritt, d.h.  
löse  $\mathbf{y}' = \mathbf{f}(x, \mathbf{y})$  mit  $y_1(x_i) = \eta_1(x_i)$ ,  $y_2(x_i) = \eta_2(x_i)$ .  
Das zur Lösung verwendete Anfangswertverfahren benutzt ein Gitter  $\{\tilde{x}_j\}$  und liefert zugehörige Näherungslösungen  $\mathbf{u}_j$ .
- (4) Breche dieses Verfahren ab, falls  
 $\tilde{x}_j \geq b$  oder  $\|\mathbf{u}_j\| \leq \|\boldsymbol{\eta}(\tilde{x}_j)\|/K$  oder  $\|\mathbf{u}_j\| \geq K\|\boldsymbol{\eta}(\tilde{x}_j)\|$ .
- (5) Ist  $\tilde{x}_j \geq b$ , dann sind alle Zwischenstellen bestimmt. Ende!
- (6) Setze  $i := i + 1$ ,  $x_i := \tilde{x}_j$ ,  $y_i := \eta_1(x_i)$ ,  $s_i := \eta_2(x_i)$  und gehe zu (3).

Für das Mehrfachschießen nehmen wir die Verbindungsgerade der Randwerte  $\eta_1(x) := x$  als Hilfsfunktion. Sie ist in Abb. 9.5 links als Strichpunkt-Linie eingezeichnet, der Lösungsschlauch für den Faktor  $K = 2$  für  $y_1(x)$  gepunktet und die Lösungskurven gestrichelt. Damit ergeben sich die Zwischenstellen

$$\{x_i\} = \{0, 0.37, 0.54, 0.665, 0.765, 0.85, 0.907, 0.965\}, \quad (9.58)$$

mit denen das Mehrfach-Schießverfahren das Randwertproblem gut löst. Diese Lösung ist als durchgehend gezeichnete Kurve abgebildet.

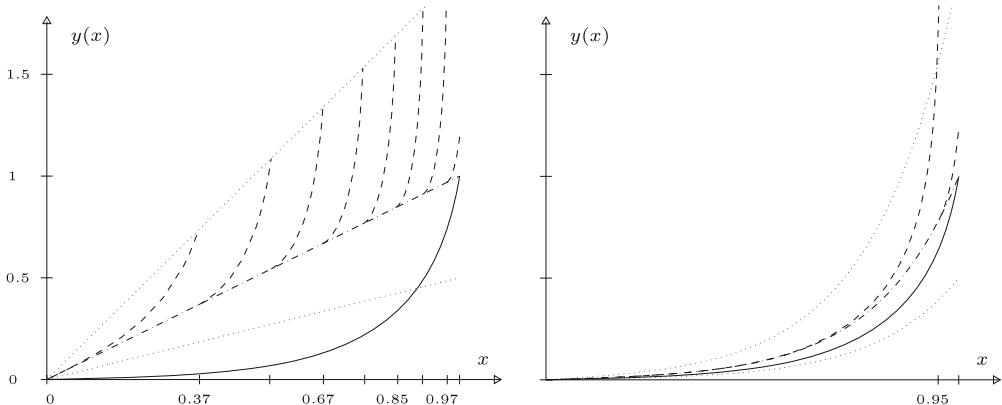


Abb. 9.5 Intervalleinteilung für das mehrfache Schießen.

In einem zweiten Versuch linearisieren wir das Randwertproblem. Es ist ja

$$\sinh(5y) = 5y + \frac{(5y)^3}{3!} + \dots$$

Damit lautet das linearisierte Randwertproblem

$$\tilde{y}'' = 25\tilde{y}, \quad \tilde{y}(0) = 0, \quad \tilde{y}(1) = 1.$$

Es ist leicht analytisch zu lösen. Zur Bestimmung der Zwischenstellen und besserer Anfangswerte für die Startrampen nehmen wir seine Lösung  $\tilde{y}(x)$  als Hilfsfunktion:

$$\eta_1(x) := \tilde{y}(x) = \frac{\sinh(5x)}{\sinh(5)}, \quad \eta_2(x) := \tilde{y}'(x) = \frac{5 \cosh(5x)}{\sinh(5)}, \quad \boldsymbol{\eta}(x) = \begin{pmatrix} \eta_1(x) \\ \eta_2(x) \end{pmatrix}.$$

Hier ergibt sich nur eine Zwischenstelle  $x_2 = 0.94$ . In Abb. 9.5 rechts sind wieder alle Anteile des Lösungsvorganges in gleicher Weise wie links eingezeichnet. Bei besserer Wahl der Näherungslösung genügt also eine Zwischenstelle. Sie vermeidet das Hineinlaufen in die Singularität.  $\triangle$

## 9.4 Differenzenverfahren

Das Intervall  $[a, b]$  wird in  $n + 1$  gleich lange Intervalle  $[x_i, x_{i+1}]$  aufgeteilt mit

$$\begin{aligned} x_i &:= a + i h, \quad i = 0, 1, \dots, n + 1, \quad \text{wo} \\ h &:= \frac{b - a}{n + 1}. \end{aligned} \tag{9.59}$$

Jetzt ersetzt man für jeden inneren Punkt  $x_i$  dieses Gitters die Differenzialgleichung durch eine algebraische Gleichung mit Näherungen der Funktionswerte  $y(x_i)$  als Unbekannte, indem man alle Funktionen in  $x_i$  auswertet und die Ableitungswerte durch dividierte Differenzen approximiert. So bekommt man statt einer Differenzialgleichung ein System von  $n$  Gleichungen mit den  $n$  unbekannten Werten der Lösungsfunktion  $y(x_i)$ . Die gegebenen Randwerte können dabei eingesetzt werden. Wir werden dies für ein Beispiel gleich durchführen. Vorher wollen wir verschiedene Differenzenapproximationen kennenlernen.

### 9.4.1 Dividierte Differenzen

Sei  $y(x)$  eine Funktion genügend hoher Differenzierbarkeit, sei weiter  $y_i := y(x_i)$ ,  $x_i$  wie in (9.59). Im Abschnitt 3.1.6 haben wir mit Hilfe der Lagrange-Interpolation dividierte Differenzen als Approximationen für Ableitungswerte kennen gelernt. Diese wollen wir hier wieder aufgreifen und zusätzliche Differenzenapproximationen kennenlernen, zunächst für die erste Ableitung von  $y$ :

Vorwärtsdifferenz

$$\Delta y_i := \frac{y(x_{i+1}) - y(x_i)}{h} = y'(x_i) + \frac{h}{2} y''(\zeta) \tag{9.60}$$

Rückwärtsdifferenz

$$\nabla y_i := \frac{y(x_i) - y(x_{i-1})}{h} = y'(x_i) - \frac{h}{2} y''(\zeta) \tag{9.61}$$

Zentrale Differenz

$$\delta y_i := \frac{y(x_{i+1}) - y(x_{i-1})}{2h} = y'(x_i) + \frac{h^2}{3} y'''(\zeta) \tag{9.62}$$

Dabei ist  $\zeta$  jeweils eine Zwischenstelle im entsprechenden Intervall.

Man sieht, dass die zentrale Differenz für die 1. Ableitung die einzige Differenzenapproximation 2. Ordnung ist. Wir wollen jetzt für die ersten vier Ableitungen von  $y$  Differenzenapproximationen angeben, die alle diese Genauigkeit  $O(h^2)$  besitzen:

$$\begin{aligned} y'(x_i) &= \frac{y(x_{i+1}) - y(x_{i-1})}{2h} + O(h^2), \\ y''(x_i) &= \frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2} + O(h^2), \\ y'''(x_i) &= \frac{y(x_{i+2}) - 2y(x_{i+1}) + 2y(x_{i-1}) - y(x_{i-2})}{2h^3} + O(h^2), \\ y^{(4)}(x_i) &= \frac{y(x_{i+2}) - 4y(x_{i+1}) + 6y(x_i) - 4y(x_{i-1}) + y(x_{i-2})}{h^4} + O(h^2). \end{aligned} \quad (9.63)$$

Näherungen höherer Ordnung lassen sich bei Einbeziehung von mehr Nachbarwerten konstruieren; hier seien noch zwei Differenzenapproximationen 4. Ordnung genannt:

$$\begin{aligned} y'(x_i) &= \frac{y(x_{i-2}) - 8y(x_{i-1}) + 8y(x_{i+1}) - y(x_{i+2})}{12h} + O(h^4) \\ y''(x_i) &= \frac{-y(x_{i-2}) + 16y(x_{i-1}) - 30y(x_i) + 16y(x_{i+1}) - y(x_{i+2})}{12h^2} + O(h^4) \end{aligned}$$

Weitere Differenzenapproximationen lassen sich entsprechend konstruieren [Col 66].

### 9.4.2 Diskretisierung der Randwertaufgabe

Zur Veranschaulichung der Differenzenverfahren wollen wir die folgende Randwertaufgabe diskretisieren:

$$\begin{aligned} -y''(x) + q(x)y(x) &= g(x), \\ y(a) = \alpha, \quad y(b) = \beta. \end{aligned} \quad (9.64)$$

Mit den Funktionswerten  $q_i := q(x_i)$  und  $g_i := g(x_i)$  ergeben sich für die Näherungswerte  $u_i \approx y(x_i)$  die linearen Gleichungen

$$\begin{aligned} u_0 &= \alpha, \\ \frac{-u_{i+1} + 2u_i - u_{i-1}}{h^2} + q_i u_i &= g_i, \quad i = 1, \dots, n, \\ u_{n+1} &= \beta. \end{aligned} \quad (9.65)$$

Multipliziert man die Gleichungen mit  $h^2$  und bringt die Randwerte auf die rechte Seite, so bekommt man das lineare Gleichungssystem

$$A\mathbf{u} = \mathbf{k} \quad (9.66)$$

mit

$$\mathbf{A} = \begin{pmatrix} 2 + q_1 h^2 & -1 & 0 & \cdots & 0 \\ -1 & 2 + q_2 h^2 & -1 & 0 & \\ 0 & \ddots & \ddots & \ddots & \ddots \\ \vdots & \ddots & \ddots & \ddots & -1 \\ 0 & \cdots & 0 & -1 & 2 + q_n h^2 \end{pmatrix},$$

$$\mathbf{u} = (u_1, u_2, \dots, u_n)^T,$$

$$\mathbf{k} = (h^2 g_1 + \alpha, h^2 g_2, \dots, h^2 g_{n-1}, h^2 g_n + \beta)^T.$$

Leicht beweisen lässt sich

**Lemma 9.2.** Das tridiagonale symmetrische Gleichungssystem (9.66) ist positiv definit, falls  $q_i \geq 0$ .

Das Gleichungssystem (9.66) kann also mit einem speziellen Cholesky-Verfahren für Bandgleichungen mit dem Aufwand  $O(n)$  gelöst werden, siehe Abschnitt 2.3.2.

### Satz 9.3. Fehlerabschätzung

Besitzt die Randwertaufgabe (9.64) eine eindeutige, viermal stetig differenzierbare Lösung  $y$  mit

$$|y^{(4)}(x)| \leq M \quad \forall x \in [a, b],$$

und ist  $q(x) \geq 0$ , dann gilt

$$|y(x_i) - u_i| \leq \frac{Mh^2}{24}(x_i - a)(b - x_i). \quad (9.67)$$

Der Satz sagt aus, dass die Fehlerordnung der Differenzenapproximation für die Lösung erhalten bleibt. Den Beweis findet man z.B. in [Sto 05].

Wie bei der Trapezregel zur numerischen Quadratur (7.24) lässt sich der Diskretisierungsfehler  $u_i - y(x_i)$  des Differenzenverfahrens an einer Stelle  $x_i$  mit Hilfe von Taylor-Reihen in eine Potenzreihe in  $h^2$  entwickeln [Sch 97]. Es ist deshalb empfehlenswert, mit zwei Schrittweiten  $h$  und  $q h$  zu rechnen und mit *Extrapolation auf  $h^2 = 0$*  die Genauigkeit der Lösung zu verbessern. Für  $q = 1/2$  gilt:

$$\frac{1}{3}(4u_{2i}^{[qh]} - u_i^{[h]}) - y(x_i) = O(h^4). \quad (9.68)$$

**Beispiel 9.11.** Wir wollen das einführende Balkenbeispiel 9.3

$$-y''(x) - (1 + x^2) y(x) = 1, \quad y(-1) = y(1) = 0 \quad (9.69)$$

mit der Differenzenmethode behandeln. Es hat die Form (9.64), allerdings ist  $q(x) < 0$ . Die Matrix  $\mathbf{A}$  in (9.66) ist trotzdem für alle in Frage kommenden Werte von  $h$  positiv definit. Zur Verkleinerung der Ordnung des linearen Gleichungssystems kann noch die Symmetrie der Lösung ausgenutzt werden. Es ist ja offensichtlich  $y(-x) = y(x)$ . Die Symmetrie der Koeffizientenmatrix  $\mathbf{A}$  kann durch

Multiplikation einer Gleichung mit dem Faktor  $1/2$  wieder hergestellt werden. Ohne Ausnutzung der Symmetrie ergibt sich für  $h = 0.4$  nach (9.66)

$$\mathbf{A} = \begin{pmatrix} 1.7824 & -1.0000 & 0 & 0 \\ -1.0000 & 1.8336 & -1.0000 & 0 \\ 0 & -1.0000 & 1.8336 & -1.0000 \\ 0 & 0 & -1.0000 & 1.7824 \end{pmatrix}.$$

Wir haben die Lösung für verschiedene Werte von  $h$  mit sechzehnstelliger Genauigkeit berechnet und die Lösungswerte in Tab. 9.4 auszugsweise wiedergegeben.

Tab. 9.4 Ergebnisse des Differenzenverfahrens für (9.69).

$x$	$h = 0.2$	$h = 0.1$	$h = 0.05$	$h = 0.01$	$h = 0.001$
0.0	0.93815233	0.93359133	0.93243889	0.93206913	0.93205387
0.2	0.89938929	0.89492379	0.89379608	0.89343431	0.89341938
0.4	0.78321165	0.77908208	0.77804080	0.77770688	0.77769310
0.6	0.59069298	0.58728118	0.58642288	0.58614781	0.58613646
0.8	0.32604062	0.32393527	0.32340719	0.32323808	0.32323111
$ u_j(0) - y(0) $	0.0061	0.0015	0.00039	0.000015	0.00000015
$\text{cond}(\mathbf{A}) \approx$	100	300	1500	37282	3728215

Der Fehlerverlauf im Intervall  $(-1, 1)$  ist recht gleichmäßig, und es ist an den Fehlerwerten für  $x = 0$  sehr schön zu sehen, dass der Fehler sich wie  $h^2$  verhält. Es sind noch die Konditionszahlen des linearen Gleichungssystems mit der Matrix  $\mathbf{A}$  geschätzt und mit angegeben worden. Es gilt [Sch 97]

$$\text{cond}(\mathbf{A}) = O(h^{-2}) = O(n^2). \quad (9.70)$$

Auf die Ergebnisse aus Tab. 9.4 wenden wir noch Richardsons Extrapolation auf  $h^2 = 0$  an. So erhalten wir beispielsweise für die Werte in  $x = 0$  folgende Verbesserungen:

$x$	$h = 0.2$	$h = 0.1$	$h = 0.05$
0.0	0.93815233	0.93359133	0.93243889
extrapoliert:	—	0.93207101	0.93205366

Der einmal extrapolierte Wert  $u_{\text{extra1}}(0) = (4u_{h=0.1}(0) - u_{h=0.2}(0))/3$  erreicht schon etwa die Genauigkeit des Wertes  $u_{h=0.01}(0)$ . Der entsprechend dem Romberg-Verfahren (7.27) zweifach extrapolierte Wert unten rechts in der Tabelle ist genauer als  $u_{h=0.001}(0)$ .  $\triangle$

### Ableitungen in den Randbedingungen

Wenn an den Rändern des Integrationsintervalls Ableitungswerte vorgeschrieben sind, dann lassen sich die Randbedingungen nicht ohne Weiteres auf die rechte Seite des Gleichungssystems (9.66) bringen. Ist etwa im rechten Randpunkt  $b = x_{n+1}$  die Randbedingung

$$y'(b) + \gamma y(b) = \beta \quad (9.71)$$

zu erfüllen, dann ist der Wert  $u_{n+1}$  nicht vorgegeben und ist als Unbekannte zu behandeln. Die Randbedingung (9.71) wird wie die Differenzialgleichung an einer inneren Stützstelle durch eine Differenzengleichung approximiert. Dazu wird eine zusätzliche, außerhalb des Intervalls  $[a, b]$  liegende Stützstelle  $x_{n+2} = b + h$  betrachtet mit dem Stützwert  $u_{n+2}$ . Als Differenzenapproximation von (9.71) wird dann

$$\frac{u_{n+2} - u_n}{2h} + \gamma u_{n+1} = \beta \quad (9.72)$$

betrachtet. (9.65) wird um die Differenzengleichungen für die Stützstelle  $x_{n+1}$  ergänzt. Dort wird dann für  $u_{n+2}$  der aus (9.72) gewonnene Wert  $u_{n+2} = u_n - 2h\gamma u_{n+1} + 2h\beta$  eingesetzt und somit  $u_{n+2}$  eliminiert. Die dritte Gleichung in (9.65) wird dann durch

$$\frac{(2 + 2h\gamma)u_{n+1} - 2u_n}{h^2} + q_{n+1}u_{n+1} = g_{n+1} + \frac{2\beta}{h} \quad (9.73)$$

ersetzt. Sie vervollständigt die  $n$  Differenzengleichungen (9.65) zu einem System von  $n + 1$  Gleichungen für die  $n + 1$  Unbekannten  $u_1, u_2, \dots, u_n, u_{n+1}$ . Liegt eine Randbedingung mit Ableitungswert an der Stelle  $a = x_0$  vor, wird entsprechend verfahren.

Das zu lösende lineare Gleichungssystem bleibt wie in Beispiel 9.11 durch Multiplikation einer oder zweier Gleichungen mit einem Faktor symmetrisch.

### Nichtlineare Randwertaufgaben

Bei der Diskretisierung der Differenzialgleichung

$$\begin{aligned} y''(x) &= f(x, y, y'), \\ r_1(y(a), y'(a), y(b), y'(b)) &= 0, \\ r_2(y(a), y'(a), y(b), y'(b)) &= 0, \end{aligned} \quad (9.74)$$

entstehen nichtlineare Gleichungssysteme, die für (9.74) von der Form

$$\mathbf{B}\mathbf{u} = \mathbf{F}(\mathbf{u}) \quad (9.75)$$

sind. Dabei ist für (9.74)  $\mathbf{B}$  eine tridiagonale Matrix und  $\mathbf{F}$  eine nichtlineare Vektorfunktion. (9.75) kann mit dem Newton-Verfahren oder mit speziellen Relaxationsmethoden gelöst werden. Konvergiert das gewählte Iterationsverfahren, so erhält man dieselbe Fehlerordnung wie bei linearen Problemen.

### Beispiel 9.12. Diskretisierung der Randwertaufgabe aus Beispiel 9.2

$$\begin{aligned} y'' &= y^5 - 10y + \frac{1}{2}, \\ y(0) &= 0, \\ y'(1) &= -3. \end{aligned}$$

In die Diskretisierung dieser Randwertaufgabe nehmen wir die Werte am rechten Rand und am außerhalb des Intervalls  $[0, 1]$  liegenden Punkt  $x_{n+2} = 1 + h$  wie in (9.72) als unbekannte Werte mit hinein und können dann die Randbedingung rechts mit einer zentralen Differenz und damit mit der Ordnung  $O(h^2)$  diskretisieren und anschliessend  $u_{n+2}$  wie in (9.73) eliminieren. Das führt zu

$$\begin{aligned} u_0 &= 0, \\ \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + 10u_i &= u_i^5 + \frac{1}{2}, \quad i = 1, \dots, n, \end{aligned}$$

$$\frac{2(u_n - u_{n+1})}{h^2} + 10u_{n+1} = u_{n+1}^5 + \frac{1}{2} + \frac{6}{h}.$$

Nach Multiplikation mit  $h^2$  wird das für die unbekannten Werte  $\mathbf{u} := (u_1, \dots, u_{n+1})$  zu

$$\begin{pmatrix} -2 + 10h^2 & 1 & & & \\ 1 & -2 + 10h^2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 + 10h^2 & 1 \\ & & & 2 & -2 + 10h^2 \end{pmatrix} \mathbf{u} = h^2 \mathbf{F}(\mathbf{u}).$$

Dabei ist  $\mathbf{F}(\mathbf{u}) = (F_1(\mathbf{u}), \dots, F_{n+1}(\mathbf{u}))^T$  mit

$$F_i(\mathbf{u}) = u_i^5 + \frac{1}{2}, \quad i = 1, \dots, n, \quad \text{und} \quad F_{n+1}(\mathbf{u}) = u_{n+1}^5 + \frac{1}{2} + \frac{6}{h}.$$

Für  $h = 0.01$  ergibt sich:

$$\mathbf{B} = \begin{pmatrix} -1.999 & 1 & & & \\ 1 & -1.999 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -1.999 & 1 \\ & & & 2 & -1.999 \end{pmatrix}.$$

Wir setzen  $\mathbf{u}^{[0]} = (0, \dots, 0)^T$  und iterieren  $\mathbf{B}\mathbf{u}^{[i+1]} = h^2 \mathbf{F}(\mathbf{u}^{[i]})$ ,  $i = 0, 1, \dots$ . Diese Iterationsvorschrift benötigt 8 (17) Schritte für 4 (9) Stellen Genauigkeit. Diskretisiert man die rechte Randbedingung nur mit  $O(h)$ -Genauigkeit  $(u_{n+1} - u_n)/h = -3$ , dann wird die Konvergenz um 2 (7) Schritte langsamer.

Abhängig von der Startnäherung  $\mathbf{u}^{[0]}$  liefert das Verfahren die kleinere der beiden Lösungen aus Beispiel 9.2 und Abb. 9.1. Die ersten vier Iterierten sind in Abb. 9.6 zu sehen.  $\triangle$

## Differenzenverfahren für Eigenwertprobleme

Bei der Diskretisierung der speziellen Sturm-Liouville'schen Eigenwertaufgabe

$$\begin{aligned} -y'' + q(x)y &= \lambda v(x)y, \\ y(a) = 0, \quad y(b) = 0, \end{aligned} \tag{9.76}$$

entsteht das Matrix-Eigenwertproblem

$$\begin{aligned} u_0 &= 0, \\ \frac{-u_{i+1} + 2u_i - u_{i-1}}{h^2} + q_i u_i - \lambda v_i u_i &= 0, \quad i = 1, \dots, n, \\ u_{n+1} &= 0. \end{aligned} \tag{9.77}$$

Sind alle Werte  $v_i = v(x_i) \neq 0$ , so bekommt man das spezielle Matrix-Eigenwertproblem

$$(\mathbf{A} - \lambda \mathbf{I}) \mathbf{u} = 0 \quad \text{mit} \tag{9.78}$$

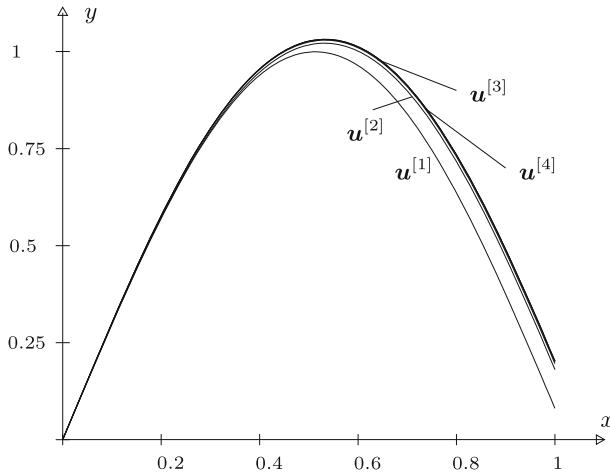


Abb. 9.6 Iterative Lösung einer nichtlinearen Randwertaufgabe.

$$A = \begin{pmatrix} \frac{2}{v_1 h^2} + \frac{q_1}{v_1} & \frac{-1}{v_1 h^2} & 0 & \cdots & 0 \\ \frac{-1}{v_2 h^2} & \frac{2}{v_2 h^2} + \frac{q_2}{v_2} & \frac{-1}{v_2 h^2} & 0 & \\ 0 & \cdots & \cdots & \cdots & \\ \vdots & & \frac{-1}{v_{n-1} h^2} & \frac{2}{v_{n-1} h^2} + \frac{q_{n-1}}{v_{n-1}} & \frac{-1}{v_{n-1} h^2} \\ 0 & \cdots & 0 & \frac{-1}{v_n h^2} & \frac{2}{v_n h^2} + \frac{q_n}{v_n} \end{pmatrix}.$$

Dies ist ein Eigenwertproblem mit einer tridiagonalen Matrix, dessen Lösung in den allermeisten Fällen unproblematisch ist.

## 9.5 Software

In Abschnitt 8.6 haben wir schon darauf hingewiesen, dass in den großen Bibliotheken die Kapitel zu gewöhnlichen Differenzialgleichungen sowohl Routinen für Anfangs- als auch für Randwertprobleme enthalten. Zu letzteren findet man deutlich weniger einzelne Programme oder Pakete. Wir verweisen deshalb auf Abschnitt 8.6.

Hinweise wollen wir aber noch auf den Befehl `bvp4c` in MATLAB, der Randwertprobleme mit einer Kollokationsmethode mit kubischen Splines löst [Sha 00]. `bvp4c` erlaubt zudem die Bestimmung unbekannter Problem-Parameter. Das erlaubt die Lösung von Eigenwertproblemen ebenso wie von Parameter-Identifikations-Problemen im Zusammenhang mit Randwertproblemen bei gewöhnlichen Differenzialgleichungen. Die schönen Beispiele aus [Sha 00]

sind in MATLAB eingebunden (`help bvp4c`) und im Internet erhältlich.

Unsere Problemlöseumgebung PAN (<http://www.upb.de/SchwarzKoeckler/>) verfügt über zwei Programme zur Lösung von Randwertproblemen mit einer Differenzenmethode und mit einem Mehrfach-Schießverfahren.

## 9.6 Aufgaben

**Aufgabe 9.1.** Man untersuche die Lösbarkeit der linearen Randwertaufgabe

$$y'' + y = 1$$

unter den verschiedenen Randbedingungen

- a)  $y(0) = 0, \quad y(\pi/2) = 1,$
- b)  $y(0) = 0, \quad y(\pi) = 1,$
- c)  $y(0) = 0, \quad y(\pi) = 2,$

auf Grund der allgemeinen Theorie mit Hilfe der Fundamentalmatrix des homogenen Differenzialgleichungssystems.

**Aufgabe 9.2.** Gegeben sei die lineare Randwertaufgabe

$$y'' + xy' + y = 2x,$$

$$y(0) = 1, \quad y(1) = 0.$$

- a) Durch numerische Integration bestimme man die allgemeine Lösung der Differenzialgleichung und daraus die Lösung der Randwertaufgabe.
- b) Auf Grund von  $y_0(x) = 1 - x$ , welche die inhomogenen Randbedingungen erfüllt, und von  $y_k(x) = x^k(1-x)$ ,  $k = 1, 2, \dots, n$ , welche den homogenen Randbedingungen genügen, ermittle man eine Näherungslösung nach der Kollokationsmethode im Fall  $n = 4$  einmal für äquidistante Kollokationspunkte im Innern des Intervalls und dann für die nichtäquidistanten Kollokationspunkte, welche den auf das Intervall  $[0, 1]$  transformierten Nullstellen des Tschebyscheff-Polynoms  $T_4(x)$  entsprechen. Welchen Verlauf hat die Fehlerfunktion in den beiden Fällen?
- c) Mit der Differenzenmethode bestimme man diskrete Näherungslösungen im Fall der Schrittweiten  $h = 1/m$  für die Werte  $m = 5, 10, 20, 40$ . Durch Extrapolation der Näherungswerte an den vier gemeinsamen Stützstellen ermittle man genauere Werte, und man vergleiche sie mit den Ergebnissen aus Teilaufgabe a).

**Aufgabe 9.3.** Die Lösung der linearen Randwertaufgabe

$$y'' - 5y' - 24y = 0$$

$$y(0) = 1, \quad y(2) = 2$$

kann analytisch gefunden werden. Dazu bestimme man zunächst die Lösung  $y(x, s)$  der Differenzialgleichung zur Anfangsbedingung  $y(0) = 1, y'(0) = s$ ,  $s \in \mathbb{R}$ , und dann aus der zweiten Randbedingung die gesuchte Lösungsfunktion der Randwertaufgabe. Auf Grund der analytischen Darstellung von  $y(x, s)$  analysiere man weiter die Empfindlichkeit des Funktionswertes an der Stelle  $x = 2$  auf kleine Änderungen von  $s$  um  $\Delta s$ . Was bedeutet das Ergebnis für das Einfach-Schießverfahren?

Man ermittle die Lösung auch mit dem Schießverfahren und verifiziere dabei experimentell die Empfindlichkeit der Lösung.

**Aufgabe 9.4.** Welche Empfindlichkeit auf kleine Änderungen der Anfangssteigung  $y'(0) = s$  im Rahmen des Schießverfahrens ist bei der linearen Randwertaufgabe

$$y'' - (5 + x)y' - (20 + 4x^2)y = 0$$

$$y(0) = 1, \quad y(2) = 2$$

experimentell festzustellen? Hinweis: Es gilt  $s \in [-2.7, -2.5]$ .

**Aufgabe 9.5.** Die nichtlineare Randwertaufgabe

$$y'' = 2y^2, \quad y(0) = 3/4, \quad y(1) = 1/3$$

besitzt neben  $y_1(x) = 3/(2+x)^2$  eine zweite Lösung  $y_2(x)$ .

a) Man bestimme beide Lösungen mit der Differenzenmethode für die Schrittweiten  $h = 1/2, 1/4, 1/8, 1/16$ . Aus den resultierenden Näherungswerten an der gemeinsamen Stelle  $x = 0.5$  schließe man durch Extrapolation auf den Wert der Lösungsfunktion.

b) Auf Grund der so erhaltenen Information über beide Lösungen bestimme man sie auch mit dem Schießverfahren und vergleiche insbesondere die Näherungen an der Stelle  $x = 0.5$ .

**Aufgabe 9.6.** Man behandle die Randwertaufgabe

$$y'' = \frac{6(y-1)}{x^2}, \quad y(0) = 1, \quad y(1) = 0,$$

mit dem Mehrfach-Schießverfahren.

Man bestimme die Zwischenstellen und Startwerte mit Hilfe des Algorithmus Tab. 9.3. Zur Bestimmung der Funktion  $\eta(x)$  linearisiere man die Randwerte und wähle den Faktor  $K = 2$ . Man berücksichtige das Problem der Singularität der rechten Seite der Differenzialgleichung.

Schließlich fertige man eine Zeichnung an mit  $\eta(x)$ , den Startrampen und der analytischen Lösung  $y(x) = -x^3 + 1$ .

## 10 Partielle Differenzialgleichungen

Zahlreiche Vorgänge oder Zustände, die man in der Physik, Chemie oder Biologie beobachten kann, lassen sich durch Funktionen in mehreren unabhängigen Variablen beschreiben, welche auf Grund von einschlägigen Naturgesetzen bestimmten partiellen Differenzialgleichungen genügen müssen. Die Vielfalt der in den Anwendungen auftretenden partiellen Differenzialgleichungen und Differenzialgleichungssysteme ist sehr groß, und ihre sachgemäße numerische Behandlung erfordert in der Regel sehr spezielle Methoden, so dass wir uns im Folgenden einschränken müssen. Wir betrachten nur die Lösung von partiellen Differenzialgleichungen zweiter Ordnung für eine unbekannte Funktion in zwei und drei unabhängigen Variablen. Die hier untersuchten partiellen Differenzialgleichungen sind zudem entweder *elliptisch* oder *parabolisch*. Im ersten Fall haben alle unabhängigen Variablen die Bedeutung von räumlichen Koordinaten, und die gesuchte Funktion beschreibt in der Regel einen *stationären* d.h. zeitunabhängigen Zustand. Man spricht hier auch von Gleichgewichtsproblemen. Im anderen Fall ist eine Variable gleich der Zeit, während die übrigen wieder Ortskoordinaten sind, und die Funktion beschreibt einen *instationären Vorgang* in Abhängigkeit der Zeit. Hier handelt es sich um Evolutions- bzw. Fortpflanzungsprobleme und speziell oft um *Diffusionsprozesse*. Mit diesen beiden Problemklassen erfassen wir eine große Zahl von praktisch relevanten Aufgabenstellungen, an denen wir einige einfache Lösungsmethoden entwickeln und ihre grundlegenden Eigenschaften diskutieren wollen. Ausführlichere Darstellungen findet man beispielsweise in [Col 66, Hac 96, Lap 99, Mar 86, Mit 80, Mor 94, Kan 95, Gro 05, Smi 85], wo auch andere Typen von partiellen Differenzialgleichungen behandelt werden. Dazu gehört besonders der dritte wichtige Typ der hyperbolischen oder Wellengleichung, auf deren Behandlung wir verzichtet haben.

### 10.1 Elliptische Randwertaufgaben, Differenzenverfahren

#### 10.1.1 Problemstellung

Gesucht sei eine Funktion  $u(x, y)$ , welche in einem Gebiet  $G \subset \mathbb{R}^2$  eine *lineare partielle Differenzialgleichung zweiter Ordnung*

$$Au_{xx} + 2Bu_{xy} + Cu_{yy} + Du_x + Eu_y + Fu = H \quad (10.1)$$

erfüllen soll. Dabei können die gegebenen Koeffizienten  $A, B, C, D, E, F$  und  $H$  in (10.1) stückweise stetige Funktionen von  $x$  und  $y$  sein und die Indizes an  $u$  bedeuten partielle Ableitungen ( $u_x := \frac{\partial u}{\partial x}$ ,  $u_{xx} := \frac{\partial^2 u}{\partial x^2}$ ,  $\dots$ ).

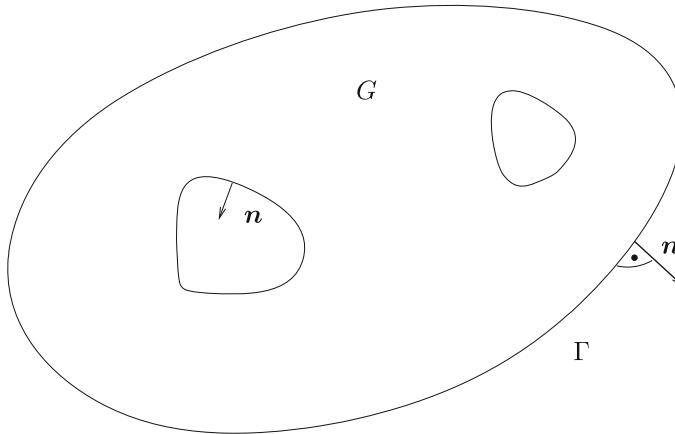


Abb. 10.1 Grundgebiet  $G$  mit Rand  $\Gamma$ .

In Analogie zur Klassifikation von Kegelschnittgleichungen

$$Ax^2 + 2Bxy + Cy^2 + Dx + Ey + F = 0$$

teilt man die partiellen Differenzialgleichungen (10.1) in drei Klassen ein:

**Definition 10.1.** Eine partielle Differenzialgleichung zweiter Ordnung (10.1) mit  $A^2 + B^2 + C^2 \neq 0$  heißt in einem Gebiet  $G$

- a) *elliptisch*, falls  $AC - B^2 > 0$  für alle  $(x, y) \in G$
- b) *hyperbolisch*, falls  $AC - B^2 < 0$  für alle  $(x, y) \in G$
- c) *parabolisch*, falls  $AC - B^2 = 0$  für alle  $(x, y) \in G$  gilt.

Die klassischen Repräsentanten von *elliptischen Differenzialgleichungen* sind

$$-\Delta u := -u_{xx} - u_{yy} = 0 \quad \text{Laplace-Gleichung,} \quad (10.2)$$

$$-\Delta u = f(x, y) \quad \text{Poisson-Gleichung,} \quad (10.3)$$

$$-\Delta u + \varrho(x, y) u = f(x, y) \quad \text{Helmholtz-Gleichung.} \quad (10.4)$$

Die Laplace-Gleichung tritt beispielsweise auf bei Problemen aus der Elektrostatik sowie der Strömungslehre. Die Lösung der Poisson-Gleichung beschreibt die stationäre Temperaturverteilung in einem homogenen Medium oder den Spannungszustand bei bestimmten Torsionsproblemen.

Um die gesuchte Lösungsfunktion einer elliptischen Differenzialgleichung eindeutig festzulegen, müssen auf dem Rand des Grundgebietes  $G$  *Randbedingungen* vorgegeben sein. Wir wollen der Einfachheit halber annehmen, das Gebiet  $G$  sei beschränkt, und es werde durch mehrere Randkurven berandet (vgl. Abb. 10.1). Die Vereinigung sämtlicher Randkurven bezeichnen wir mit  $\Gamma$ . Der Rand bestehe aus stückweise stetig differenzierbaren Kurven, auf

denen die vom Gebiet  $G$  ins Äußere zeigende Normalenrichtung  $\mathbf{n}$  erklärt werden kann. Der Rand  $\Gamma$  werde in drei disjunkte Randteile  $\Gamma_1, \Gamma_2$  und  $\Gamma_3$  aufgeteilt, derart dass

$$\Gamma_1 \cup \Gamma_2 \cup \Gamma_3 = \Gamma \quad (10.5)$$

gilt. Dabei ist es durchaus zulässig, dass leere Teilränder vorkommen. Die problemgerechte Formulierung der Randbedingungen zu (10.1) oder speziell zu (10.2) bis (10.4) lautet dann

$$u = \varphi \text{ auf } \Gamma_1 \quad (\text{Dirichlet-Randbedingung}), \quad (10.6)$$

$$\frac{\partial u}{\partial \mathbf{n}} = \gamma \text{ auf } \Gamma_2 \quad (\text{Neumann-Randbedingung}), \quad (10.7)$$

$$\frac{\partial u}{\partial \mathbf{n}} + \alpha u = \beta \text{ auf } \Gamma_3 \quad (\text{Cauchy-Randbedingung}), \quad (10.8)$$

wobei  $\varphi, \gamma, \alpha$  und  $\beta$  gegebene Funktionen auf den betreffenden Randteilen bedeuten. In der Regel sind sie als Funktionen der Bogenlänge  $s$  auf dem Rand erklärt. Die Bedingungen (10.6), (10.7) und (10.8) werden oft auch als erste, zweite und dritte Randbedingung bezeichnet. Sind zur elliptischen Differenzialgleichung nur Dirichletsche Randbedingungen gegeben ( $\Gamma_1 = \Gamma$ ), dann bezeichnet man das Problem auch als *Dirichletsche Randwertaufgabe*. Ist dagegen  $\Gamma_2 = \Gamma$ , so liegt eine *Neumannsche Randwertaufgabe* vor.

### 10.1.2 Diskretisierung der Aufgabe

Wir wollen die Laplace-, Poisson- oder Helmholtz-Gleichung in einem Gebiet  $G$  unter Randbedingungen (10.6) bis (10.8) näherungsweise lösen. Wir beginnen mit einfachen Aufgabenstellungen, um dann sukzessive kompliziertere Situationen in die Behandlung einzubeziehen. Das Vorgehen des *Differenzenverfahrens* lässt sich durch die folgenden, recht allgemein formulierten Lösungsschritte beschreiben.

1. *Lösungsschritt*. Die gesuchte Funktion  $u(x, y)$  wird ersetzt durch ihre Werte an diskreten Punkten des Gebietes  $G$  und des Randes  $\Gamma$ . Für diese *Diskretisierung* von  $u(x, y)$  ist es naheliegend, ein regelmäßiges quadratisches Netz mit der *Gitterweite*  $h$  über das Grundgebiet  $G$  zu legen (vgl. Abb. 10.2). Die Funktionswerte  $u$  in den *Gitterpunkten* sollen berechnet werden, soweit sie nicht schon durch Dirichletsche Randbedingungen bekannt sind. Im Fall von krummlinigen Randstücken wird es auch nötig sein, Gitterpunkte als Schnittpunkte von Netzgeraden mit dem Rand zu betrachten. In Abb. 10.2 sind die Gitterpunkte durch ausgefüllte Kreise markiert.

Den Wert der exakten Lösungsfunktion  $u(x, y)$  in einem Gitterpunkt  $P$  mit den Koordinaten  $x_i$  und  $y_j$  bezeichnen wir mit  $u(x_i, y_j)$ . Den zugehörigen Näherungswert, den wir auf Grund der Methode erhalten werden, bezeichnen wir mit  $u_{i,j}$ .

Ein regelmäßiges quadratisches Netz zur Generierung der Gitterpunkte besitzt besonders angenehme und einfache Eigenschaften, die wir im Folgenden auch als wesentlich erkennen werden. In bestimmten Problemstellungen ist es angezeigt oder sogar erforderlich, ein Netz mit variablen Gitterweiten in  $x$ - und  $y$ -Richtung zu verwenden, um so entweder dem Gebiet oder dem Verhalten der gesuchten Lösungsfunktion besser gerecht zu werden, siehe Abschnitt 10.1.5. Aber auch regelmäßige Dreieck- und Sechsecknetze können sich als sehr zweckmäßig erweisen [Col 66, Mar 86].

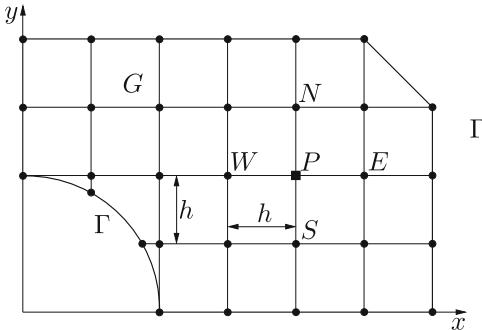


Abb. 10.2  
Grundgebiet mit Netz und Gitterpunkten.

**2. Lösungsschritt.** Nach vorgenommener Diskretisierung der Funktion ist die partielle Differenzialgleichung mit Hilfe der diskreten Funktionswerte  $u_{i,j}$  in den Gitterpunkten geeignet zu approximieren. Im Fall eines regelmäßigen quadratischen Netzes können die ersten und zweiten partiellen Ableitungen durch entsprechende Differenzenquotienten (siehe Abschnitt 9.4.1) angenähert werden, wobei für die ersten partiellen Ableitungen mit Vorteil zentrale Differenzenquotienten (9.63) verwendet werden. Für einen *regelmäßigen inneren Gitterpunkt*  $P(x_i, y_j)$ , welcher vier benachbarte Gitterpunkte im Abstand  $h$  besitzt, ist

$$u_x(x_i, y_j) \approx \frac{u_{i+1,j} - u_{i-1,j}}{2h}, \quad u_y(x_i, y_j) \approx \frac{u_{i,j+1} - u_{i,j-1}}{2h} \quad (10.9)$$

$$\begin{aligned} u_{xx}(x_i, y_j) &\approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2}, \\ u_{yy}(x_i, y_j) &\approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h^2}, \end{aligned} \quad (10.10)$$

wobei wir die Differenzenquotienten bereits mit den Näherungswerten in den Gitterpunkten gebildet haben. Um für das Folgende eine leicht einprägsame Schreibweise ohne Doppelindizes zu erhalten, bezeichnen wir die vier Nachbarpunkte von  $P$  nach den Himmelsrichtungen mit  $N, W, S$  und  $E$  (vgl. Abb. 10.2) und definieren

$$u_P := u_{i,j}, \quad u_N := u_{i,j+1}, \quad u_W := u_{i-1,j}, \quad u_S := u_{i,j-1}, \quad u_E := u_{i+1,j}. \quad (10.11)$$

Die Poisson-Gleichung (10.3) wird damit im Gitterpunkt  $P$  approximiert durch die *Differenzengleichung*

$$\frac{-u_E + 2u_P - u_W}{h^2} + \frac{-u_N + 2u_P - u_S}{h^2} = f_P, \quad f_P := f(x_i, y_j),$$

welche nach Multiplikation mit  $h^2$  übergeht in

$4u_P - u_N - u_W - u_S - u_E = h^2 f_P.$

(10.12)

Der von  $h^2$  befreite Differenzenausdruck in (10.12) wird häufig durch einen so genannten *Differenzenstern* geometrisch symbolisiert. An ihm können die Punkte eines Gitters mit ihren Faktoren abgelesen werden, die an der Gleichung für den durch  $\blacksquare$  gekennzeichneten Punkt beteiligt sind. Viele solche Sterne mit Fehlerglied findet man in [Col 66], ebenso Neun-Punkte-Sterne für  $-\Delta u$  und Sterne für triangulierte Gebiete.

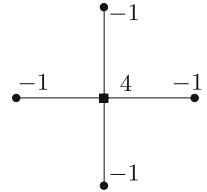


Abb. 10.3  
Fünf-Punkte-Differenzenstern zu (10.12).

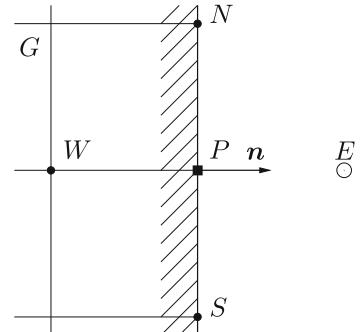


Abb. 10.4  
Spezielle Neumannsche Randbedingung.

**3. Lösungsschritt.** Die gegebenen Randbedingungen der Randwertaufgabe sind jetzt zu berücksichtigen, und allenfalls ist die Differenzenapproximation der Differenzialgleichung den Randbedingungen anzupassen.

Die einfachste Situation liegt vor, falls nur Dirichletsche Randbedingungen zu erfüllen sind und das Netz so gewählt werden kann, dass nur regelmäßige innere Gitterpunkte entstehen. In diesem Fall ist die Differenzengleichung (10.12) für alle inneren Gitterpunkte, in denen der Funktionswert unbekannt ist, uneingeschränkt anwendbar, wobei die bekannten Randwerte eingesetzt werden können. Existieren jedoch unregelmäßige Gitterpunkte wie in Abb. 10.2, so sind für diese geeignete Differenzengleichungen herzuleiten. Auf die Behandlung von solchen randnahen, unregelmäßigen Gitterpunkten werden wir in Abschnitt 10.1.3 eingehen.

Neumannsche und Cauchysche Randbedingungen (10.7) und (10.8) erfordern im Allgemeinen umfangreichere Maßnahmen, die wir systematisch im Abschnitt 10.1.3 behandeln werden. An dieser Stelle wollen wir wenigstens eine einfache Situation betrachten. Wir wollen annehmen, der Rand falle mit einer Netzgeraden parallel zur  $y$ -Achse zusammen, und die Neumannsche Randbedingung verlange, dass die Normalableitung verschwinde (vgl. Abb. 10.4). Die äußere Normale  $\mathbf{n}$  zeige in Richtung der positiven  $x$ -Achse. Mit dem vorübergehend eingeführten Hilfsgitterpunkt  $E$  und dem Wert  $u_E$  kann die Normalableitung durch den zentralen Differenzenquotienten approximiert werden. Das ergibt

$$\left. \frac{\partial u}{\partial \mathbf{n}} \right|_P \approx \frac{u_E - u_W}{2h} = 0 \implies u_E = u_W.$$

Das Verschwinden der Normalableitung bedeutet oft, dass die Funktion  $u(x, y)$  bezüglich des Randes symmetrisch ist. Wegen dieser Symmetrieeigenschaft darf die Funktion  $u(x, y)$  über den Rand hinaus fortgesetzt werden, und die allgemeine Differenzengleichung (10.12) darf angewendet werden. Aus ihr erhalten wir nach Division durch 2, die später begründet

wird,

$$2u_P - \frac{1}{2}u_N - u_W - \frac{1}{2}u_S = \frac{1}{2}h^2f_P. \quad (10.13)$$

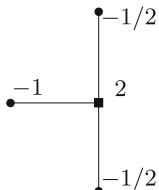


Abb. 10.5

Vier-Punkte-Differenzenstern zu (10.13).

**4. Lösungsschritt.** Um die unbekannten Funktionswerte in den Gitterpunkten berechnen zu können, sind dafür Gleichungen zu formulieren. Da nach den beiden vorangehenden Lösungsschritten für jeden solchen Gitterpunkt eine lineare Differenzengleichung vorliegt, ist es möglich, ein lineares Gleichungssystem für die unbekannten Funktionswerte zu formulieren. Zu diesem Zweck werden zur Vermeidung von Doppelindizes die Gitterpunkte des Netzes, deren Funktionswerte unbekannt sind, durchnummertiert. Die Nummerierung der Gitterpunkte muss nach bestimmten Gesichtspunkten erfolgen, damit das entstehende Gleichungssystem geeignete Strukturen erhält, welche den Lösungsverfahren angepasst sind. Das lineare Gleichungssystem stellt die *diskrete Form* der gegebenen Randwertaufgabe dar.

**Beispiel 10.1.** Im Grundgebiet  $G$  der Abb. 10.6 soll die Poisson-Gleichung

$$-\Delta u = 2 \text{ in } G \quad (10.14)$$

unter den Randbedingungen

$$u = 0 \quad \text{auf } DE \text{ und } EF \quad (10.15)$$

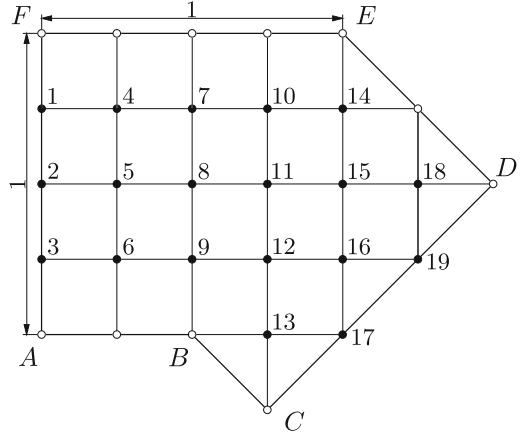
$$u = 1 \quad \text{auf } AB \text{ und } BC \quad (10.16)$$

$$\frac{\partial u}{\partial n} = 0 \quad \text{auf } CD \text{ und } FA \quad (10.17)$$

gelöst werden. Die Lösung der Randwertaufgabe beschreibt beispielsweise den Spannungszustand eines unter Torsion belasteten Balkens. Sein Querschnitt ist ringförmig und geht aus  $G$  durch fortgesetzte Spiegelung an den Seiten  $CD$  und  $FA$  hervor. Aus Symmetriegründen kann die Aufgabe im Gebiet der Abb. 10.6 gelöst werden, wobei die Neumannschen Randbedingungen (10.17) auf den beiden Randstücken  $CD$  und  $FA$  die Symmetrie beinhalten. Die betrachtete Randwertaufgabe (10.14) bis (10.17) kann auch so interpretiert werden, dass die stationäre Temperaturverteilung  $u(x, y)$  in dem ringförmigen Querschnitt eines (langen) Behälters gesucht ist, falls durch eine chemische Reaktion eine konstante Wärmequelle vorhanden ist. Die Wandtemperatur des Behälters werde innen auf den (normierten) Wert  $u = 1$  und außen auf den Wert  $u = 0$  gesetzt.

Zur Diskretisierung der Randwertaufgabe soll das in Abb. 10.6 eingezeichnete regelmäßige Netz mit der Gitterweite  $h = 0.25$  verwendet werden. Die Gitterpunkte sind entweder Randpunkte oder reguläre innere Punkte. Die Gitterpunkte mit unbekanntem Funktionswert sind durch ausgefüllte Kreise markiert, diejenigen mit nach (10.15) und (10.16) bekannten Werten durch leere Kreise.

Für alle im Innern des Grundgebietes liegenden Gitterpunkte ist die Differenzengleichung (10.12) anwendbar mit  $f_P = 2$ . Für die auf dem Randstück  $FA$  liegenden Gitterpunkte ist ein zu Abb. 10.5 gespiegelter Differenzenstern zu verwenden. Für die Gitterpunkte auf  $CD$  erhalten wir aus Symmetriegründen mit  $u_S = u_W$  und  $u_E = u_N$  aus (10.12) die Differenzengleichung (Drei-Punkte-Stern)

Abb. 10.6 Grundgebiet  $G$  mit Netz und Gitterpunkten,  $h = 0.25$ .

$4u_P - 2u_N - 2u_W = h^2 f_P$ , die aus einem bald ersichtlichen Grund durch 2 dividiert wird. Wir fassen die Differenzengleichungen für diese Randwertaufgabe zusammen:

$4u_P - u_N - u_W - u_S - u_E = h^2 f_P \quad \text{im Innern}$	$2u_P - \frac{1}{2}u_N - u_E - \frac{1}{2}u_S = \frac{1}{2}h^2 f_P \quad \text{auf } FA$	$2u_P - u_N - u_W = \frac{1}{2}h^2 f_P \quad \text{auf } CD$	(10.18)
---	--	--	---------

Die Gitterpunkte mit unbekanntem Funktionswert nummerieren wir spaltenweise durch, wie dies in Abb. 10.6 erfolgt ist. Für die zugehörigen 19 Unbekannten  $u_1, u_2, \dots, u_{19}$  können wir das lineare Gleichungssystem aufstellen. Dabei werden wir die Differenzengleichungen vernünftigerweise in der Reihenfolge der nummerierten Gitterpunkte aufschreiben und dabei in den Gleichungen allfällige Dirichletsche Randbedingungen einsetzen. Auf diese Weise entsteht das lineare Gleichungssystem (10.19), das wir in der aus Kapitel 2 bekannten Tabellen-Schreibweise dargestellt haben. Dabei sind nur die von null verschiedenen Koeffizienten und die rechte Seite (r.S.) angegeben.

Die Systemmatrix  $\mathbf{A}$  ist *symmetrisch*. Hätten wir die Differenzengleichungen für die Randpunkte mit Neumannscher Randbedingung nicht durch 2 dividiert, so wäre die Matrix  $\mathbf{A}$  unsymmetrisch geworden. Die Matrix  $\mathbf{A}$  ist schwach diagonal dominant und ist, wie man relativ leicht feststellen kann, irreduzibel oder nicht zerfallend. Da die Diagonalelemente positiv sind, ist  $\mathbf{A}$  *positiv definit* [Mae 85, Sch 72], und somit besitzt das lineare Gleichungssystem (10.19) eine eindeutige Lösung. Sie kann mit dem Verfahren von Cholesky nach Abschnitt 2.3.1 oder auch iterativ (vgl. Kapitel 11) berechnet werden. Die verwendete Nummerierung der Gitterpunkte und damit der Unbekannten hat zur Folge, dass die schwach besetzte Koeffizientenmatrix  $\mathbf{A}$  *Bandstruktur* hat mit einer Bandbreite  $m = 4$ . Mit der Rechen- und Speichertechnik von Abschnitt 2.3.2 kann das Gleichungssystem effizient gelöst werden. Die auf fünf Stellen nach dem Komma gerundete Lösung von (10.19) ist entsprechend der Lage der Gitterpunkte zusammen mit den gegebenen Randwerten

in (10.20) zusammengestellt.

$u_1 \ u_2 \ u_3 \ u_4 \ u_5 \ u_6 \ u_7 \ u_8 \ u_9 \ u_{10} \ u_{11} \ u_{12} \ u_{13} \ u_{14} \ u_{15} \ u_{16} \ u_{17} \ u_{18} \ u_{19} \quad \text{r. S.}$

$\begin{matrix} 2 & -\frac{1}{2} \\ -\frac{1}{2} & 2 & -\frac{1}{2} \\ -\frac{1}{2} & & 2 \end{matrix}$	$-1$						0.0625
$-1$	$4 \ -1$	$-1$					0.125
$-1$	$-1 \ 4 \ -1$	$-1$					0.125
$-1$	$-1 \ 4$	$-1$					1.125
	$-1$	$4 \ -1$	$-1$				0.125
	$-1$	$-1 \ 4 \ -1$	$-1$				0.125
	$-1$	$-1 \ 4$	$-1$				1.125
	$-1$	$4 \ -1$	$-1$				0.125
	$-1$	$-1 \ 4 \ -1$	$-1$				0.125
	$-1$	$-1 \ 4 \ -1$	$-1$				0.125
	$-1$	$-1 \ 4$	$-1$				2.125
		$-1$	$4 \ -1$	$-1$			0.125
		$-1$	$-1 \ 4 \ -1$	$-1$			0.125
		$-1$	$-1 \ 4 \ -1$	$-1$			0.125
		$-1$	$-1 \ 4$	$-1$			0.0625
			$-1$	$4 \ -1$	$-1$		0.125
			$-1$	$-1 \ 4 \ -1$	$-1$		0.125
			$-1$	$-1 \ 4 \ -1$	$-1$		0.125
			$-1$	$-1 \ 2$	$-1$		0.0625
				$-1$	$4 \ -1$	$0.125$	
				$-1$	$-1 \ 2$	$0.0625$	

(10.19)

$$\begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \\ 0.41686 & 0.41101 & 0.39024 & 0.34300 & 0.24049 \quad 0 \\ 0.72044 & 0.71193 & 0.68195 & 0.61628 & 0.49398 \quad 0.28682 \quad 0 \\ 0.91603 & 0.90933 & 0.88436 & 0.82117 & 0.70731 \quad 0.52832 \\ 1 & 1 & 1 & 0.95174 & 0.86077 \\ & & & 1 & \end{array}$$

(10.20)

△

### 10.1.3 Randnahe Gitterpunkte, allgemeine Randbedingungen

Wir wollen unregelmäßige innere Punkte sowie auf dem Rand liegende Gitterpunkte mit Neumannschen oder Cauchyschen Randbedingungen allgemeiner Art betrachten. Die systematische Behandlung solcher Situationen erläutern wir an typischen Beispielen, so dass die Übertragung auf andere Fälle möglich ist. Dabei geht es stets um das Problem, zu einem gegebenen Differenzialausdruck, in unserem momentan betrachteten Fall  $-\Delta u$ , eine geeignete Differenzenapproximation zu konstruieren.

**Beispiel 10.2.** Wir betrachten einen unregelmäßigen inneren Gitterpunkt  $P$ , der in der Nähe des Randes  $\Gamma$  so liegen möge, wie dies in Abb. 10.7 dargestellt ist. Die Randkurve  $\Gamma$  schneide die Netzgeraden in den Punkten  $W'$  und  $S'$ , welche von  $P$  die Abstände  $ah$  und  $bh$  mit  $0 < a, b \leq 1$  besitzen.

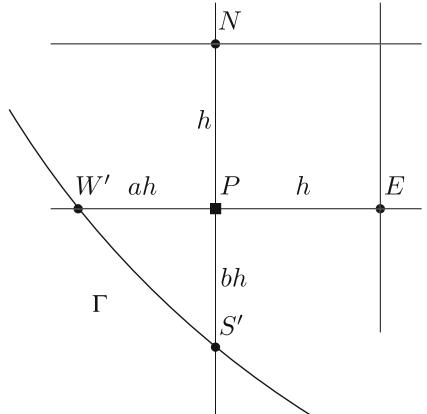


Abb. 10.7

Randnaher, unregelmäßiger Gitterpunkt.

Unser Ziel besteht darin, für die zweiten partiellen Ableitungen  $u_{xx}$  und  $u_{yy}$  im Punkt  $P(x, y)$  eine Approximation herzuleiten, die sich als Linearkombination der Werte  $u_P, u_E$  und  $u_{W'}$  beziehungsweise von  $u_P, u_N$  und  $u_{S'}$  darstellen lassen. Wir setzen  $u(x, y)$  als genügend oft stetig differenzierbar voraus. Mit Hilfe der Taylor-Entwicklungen mit Restglied erhalten wir die folgenden Darstellungen für die Funktionswerte  $u(x, y)$  in den betreffenden Punkten. Auf die Angabe des Restgliedes wird verzichtet.

$$\begin{aligned} u(x+h, y) &= u(x, y) + hu_x(x, y) + \frac{1}{2}h^2u_{xx}(x, y) + \frac{1}{6}h^3u_{xxx}(x, y) + \dots \\ u(x-ah, y) &= u(x, y) - ahu_x(x, y) + \frac{1}{2}a^2h^2u_{xx}(x, y) - \frac{1}{6}a^3h^3u_{xxx}(x, y) + \dots \\ u(x, y) &= u(x, y) \end{aligned}$$

Mit Koeffizienten  $c_1, c_2, c_3$  bilden wir die Linearkombination der drei Darstellungen

$$\begin{aligned} &c_1u(x+h, y) + c_2u(x-ah, y) + c_3u(x, y) \\ &= (c_1 + c_2 + c_3)u(x, y) + (c_1 - ac_2)hu_x(x, y) + (c_1 + a^2c_2)\frac{h^2}{2}u_{xx}(x, y) + \dots \end{aligned}$$

Aus unserer Forderung, dass die Linearkombination die zweite partielle Ableitung  $u_{xx}$  im Punkt  $P(x, y)$  approximieren soll, ergeben sich notwendigerweise die drei Bedingungsgleichungen

$$c_1 + c_2 + c_3 = 0, \quad (c_1 - ac_2)h = 0, \quad \frac{h^2}{2}(c_1 + a^2c_2) = 1.$$

Daraus folgen die Werte

$$c_1 = \frac{2}{h^2(1+a)}, \quad c_2 = \frac{2}{h^2a(1+a)}, \quad c_3 = -\frac{2}{h^2a}.$$

Zur Approximation der zweiten Ableitung  $u_{xx}(P)$  verwenden wir dividierte Differenzen mit den Näherungen  $u_E, u_P$  und  $u_{W'}$ :

$$\begin{aligned} u_{xx}(P) &\approx \frac{1}{(h+ah)/2} \left\{ \frac{u_E - u_P}{h} + \frac{u_P - u_{W'}}{ah} \right\} \\ &= \frac{2}{h^2} \left\{ \frac{u_E}{1+a} + \frac{u_{W'}}{a(1+a)} - \frac{u_P}{a} \right\}. \end{aligned} \tag{10.21}$$

Analog ergibt sich mit  $u_N, u_P$  und  $u_{S'}$  für  $u_{yy}(P)$ :

$$u_{yy}(P) \approx \frac{2}{h^2} \left\{ \frac{u_N}{1+b} + \frac{u_{S'}}{b(1+b)} - \frac{u_P}{b} \right\}. \quad (10.22)$$

Aus (10.21) und (10.22) erhalten wir so für die Poisson-Gleichung (10.3) im unregelmäßigen Gitterpunkt  $P$  der Abb. 10.7 nach Multiplikation mit  $h^2$  die Differenzengleichung

$$\left( \frac{2}{a} + \frac{2}{b} \right) u_P - \frac{2}{1+b} u_N - \frac{2}{a(1+a)} u_{W'} - \frac{2}{b(1+b)} u_{S'} - \frac{2}{1+a} u_E = h^2 f_P. \quad (10.23)$$

Falls  $a \neq b$  ist, sind die Koeffizienten von  $u_N$  und  $u_E$  in (10.23) verschieden. Dies wird im Allgemeinen zur Folge haben, dass die Matrix  $\mathbf{A}$  des Systems von Differenzengleichungen *unsymmetrisch* sein wird. Im Spezialfall  $a = b$  soll (10.23) mit dem Faktor  $(1+a)/2$  multipliziert werden, so dass  $u_N$  und  $u_E$  die Koeffizienten  $-1$  erhalten, und die Symmetrie von  $\mathbf{A}$  wird hinsichtlich  $P$  bewahrt werden können. In diesem Fall geht die Differenzengleichung (10.23) nach Multiplikation mit  $(1+a)/2$  über in

$$a = b \implies \frac{2(1+a)}{a} u_P - u_N - \frac{1}{a} u_{W'} - \frac{1}{a} u_{S'} - u_E = \frac{1}{2}(1+a)h^2 f_P. \quad (10.24)$$

△

**Beispiel 10.3.** Auf einem Randstück  $\Gamma_2$  sei eine Neumann-Randbedingung zu erfüllen. Wir betrachten die einfache Situation, wo der Randpunkt  $P$  ein Gitterpunkt ist. Die Randkurve  $\Gamma_2$  verlaufe gemäß Abb. 10.8. Die äußere Normalenrichtung  $\mathbf{n}$  im Punkt  $P$  bilde mit der positiven  $x$ -Richtung den Winkel  $\psi$ , definiert in der üblichen Weise im Gegenuhrzeigersinn. Wir verwenden auch den Wert der Normalableitung, um den Differenzialausdruck  $-\Delta u$  im Punkt  $P(x, y)$  durch eine geeignete Linearkombination zu approximieren.

Sicher werden die Funktionswerte von  $u$  in den Gitterpunkten  $P, W$  und  $S$  der Abb. 10.8 verwendet werden. Zusammen mit der Normalableitung in  $P$  sind dies vier Größen. Es ist leicht einzusehen, dass es im Allgemeinen nicht möglich ist, eine Linearkombination dieser vier Werte zu finden, welche den gegebenen Differenzialausdruck in  $P$  approximiert. Dazu sind mindestens fünf Größen notwendig, falls in keiner Taylor-Entwicklung die gemischte zweite partielle Ableitung auftritt. Andernfalls benötigen wir sechs Größen, denn der Koeffizientenvergleich liefert dann sechs Bedingungsgleichungen. Wir wählen die beiden zusätzlichen Randpunkte  $R(x - h, y + bh)$  und  $T(x + ah, y - h)$  mit  $0 < a, b \leq 1$ . Es gelten die folgenden Näherungsdarstellungen bezüglich des Punktes  $P(x, y)$ , wobei wir die Argumente und die Restglieder auf der rechten Seite weglassen.

$$\begin{aligned} P : \quad u(x, y) &= u \\ W : \quad u(x - h, y) &= u - hu_x & + \frac{h^2}{2} u_{xx} \\ S : \quad u(x, y - h) &= u & - hu_y & + \frac{h^2}{2} u_{yy} \\ R : \quad u(x - h, y + bh) &= u - hu_x & + bhu_y & + \frac{h^2}{2} u_{xx} & - bh^2 u_{xy} & + \frac{h^2}{2} b^2 u_{yy} \\ T : \quad u(x + ah, y - h) &= u + ahu_x & - hu_y & + \frac{h^2}{2} a^2 u_{xx} & - ah^2 u_{xy} & + \frac{h^2}{2} u_{yy} \\ P : \quad \frac{\partial u(x, y)}{\partial n} &= u_x \cos \psi + u_y \sin \psi \end{aligned}$$

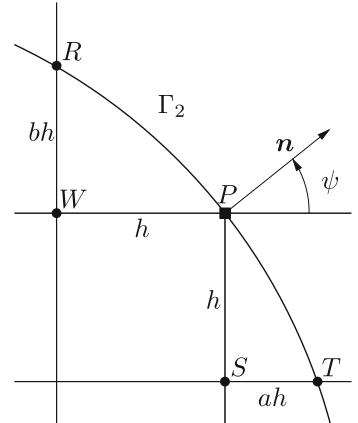


Abb. 10.8  
Neumann-Randbedingung im Randpunkt  $P$ .

Eine Linearkombination dieser sechs Darstellungen mit den Koeffizienten  $c_P, c_W, c_S, c_R, c_T$  und  $c_n$  zur angrenzenden Darstellung von  $-\Delta u$  ergibt die sechs Bedingungsgleichungen

$$\begin{aligned}
 u : \quad & c_P + c_W + c_S + c_R + c_T = 0 \\
 u_x : \quad & -hc_W - hc_R + ahc_T + c_n \cos \psi = 0 \\
 u_y : \quad & -hcs + bhc_R - hct + c_n \sin \psi = 0 \\
 u_{xx} : \quad & \frac{h^2}{2}c_W + \frac{h^2}{2}c_R + \frac{h^2}{2}a^2c_T = -1 \\
 u_{xy} : \quad & -bh^2c_R - ah^2c_T = 0 \\
 u_{yy} : \quad & \frac{h^2}{2}c_S + \frac{h^2}{2}b^2c_R + \frac{h^2}{2}c_T = -1
 \end{aligned} \tag{10.25}$$

Das Gleichungssystem (10.25) besitzt eine eindeutige Lösung, die die Differenzenapproximation ergibt. In sie geht die Geometrie des Gebietes  $G$  in der Umgebung des Randpunktes  $P$  ein. Im konkreten Fall mit gegebenen Werten von  $a, b$  und  $\psi$  wird das lineare Gleichungssystem (10.25) numerisch gelöst. Mit den Lösungswerten für die Koeffizienten lautet dann die Differenzengleichung im Punkt  $P$

$$\boxed{c_P u_P + c_W u_W + c_S u_S + c_R u_R + c_T u_T + c_n \frac{\partial u}{\partial n}}_P = f_P, \tag{10.26}$$

die zweckmäßigerweise noch mit  $h^2$  multipliziert wird. Die Neumann-Randbedingung im Punkt  $P$  wird so berücksichtigt, dass der vorgegebene Wert der Normalableitung in (10.26) eingesetzt wird.

Im Spezialfall  $\psi = 45^\circ$  mit  $\cos \psi = \sin \psi = \frac{1}{2}\sqrt{2}$  sind die Koeffizienten  $c_T$  und  $c_R$  gleich null. Somit treten die Werte  $u_T$  und  $u_R$  von diesen beiden Randpunkten in der Differenzengleichung (10.26) nicht auf. In diesem Fall können die Koeffizienten in einfacher geschlossener Form angegeben werden, nämlich als

$$c_n = \frac{-2\sqrt{2}}{h}, \quad c_S = \frac{-2}{h^2}, \quad c_W = \frac{-2}{h^2}, \quad c_P = \frac{4}{h^2}.$$

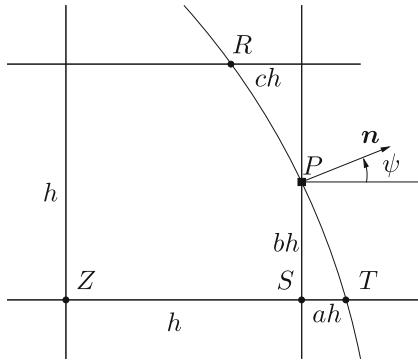


Abb. 10.9  
Cauchy-Randbedingung im Randpunkt  $P$ .

Die Differenzengleichung lautet nach Multiplikation mit  $h^2/2$

$$\psi = 45^\circ \implies 2u_P - u_W - u_S - h\sqrt{2} \frac{\partial u}{\partial n} \Big|_P = \frac{1}{2}h^2 f_P. \quad (10.27)$$

Für den Sonderfall  $\frac{\partial u}{\partial n} \Big|_P = 0$  erhalten wir im Wesentlichen die letzte der Differenzengleichungen (10.18), die dort auf andere Weise hergeleitet wurde.  $\triangle$

**Beispiel 10.4.** Die Behandlung einer Cauchyschen Randbedingung (10.8) in einem allgemeinen Randpunkt  $P$  erfolgt analog zu derjenigen einer Neumannschen Randbedingung. Um das Vorgehen aufzuzeigen, betrachten wir die Situation von Abb. 10.9. Der Randpunkt  $P(x, y)$  sei nicht Schnittpunkt von Netzgeraden. Die Richtung der äußeren Normalen bilde den Winkel  $\psi$  mit der positiven  $x$ -Achse.

Wiederum werden wir zur Approximation von  $-\Delta u$  sechs Größen benötigen. Neben dem Ausdruck  $\frac{\partial u}{\partial n} + \alpha u$  der linken Seite der Cauchy-Randbedingung im Punkt  $P$  werden wir in naheliegender Weise die Werte von  $u$  in den Punkten  $P, S, R$  und  $T$  verwenden. Als sechste Größe wählen wir den Wert von  $u$  im zu  $P$  nächstgelegenen Gitterpunkt im Innern des Gebietes. Für  $b \leq 1/2$  ist dies  $Z$ . Wir erhalten mit  $B := 1 - b$  und wieder ohne Argumente und Restglieder rechts die Näherungsgleichungen

$$\begin{aligned}
 P : u(x, y) &= u \\
 S : u(x, y - bh) &= u - bh u_y + \frac{h^2}{2} b^2 u_{yy} \\
 Z : u(x - h, y - bh) &= u - hu_x - bh u_y + \frac{h^2}{2} u_{xx} + bh^2 u_{xy} + \frac{h^2}{2} b^2 u_{yy} \\
 R : u(x - ch, y + Bh) &= u - ch u_x + Bh u_y + \frac{h^2}{2} c^2 u_{xx} - cBh^2 u_{xy} + \frac{h^2}{2} B^2 u_{yy} \\
 T : u(x + ah, y - bh) &= u + ah u_x - bh u_y + \frac{h^2}{2} a^2 u_{xx} - abh^2 u_{xy} + \frac{h^2}{2} b^2 u_{yy} \\
 P : \frac{\partial u}{\partial n} + \alpha u &= \alpha u + u_x \cos \psi + u_y \sin \psi
 \end{aligned}$$

Für die Koeffizienten  $c_P, c_S, c_Z, c_R, c_T$  und  $c_n$  der Linearkombination zur Darstellung von  $-\Delta u$  zu

bildenden Linearkombination ergeben sich durch Koeffizientenvergleich die sechs Bedingungsgleichungen

$$\begin{aligned}
 u : \quad c_P + c_S + c_Z + c_R + c_T + \alpha c_n &= 0 \\
 u_x : \quad -hc_Z - chc_R + ahc_T + c_n \cos \psi &= 0 \\
 u_y : \quad -bhcs - bhc_Z + Bhc_R - bhct + c_n \sin \psi &= 0 \\
 u_{xx} : \quad \frac{h^2}{2}cz + \frac{h^2}{2}c^2c_R + \frac{h^2}{2}a^2ct &= -1 \\
 u_{xy} : \quad bh^2cz - cBh^2c_R - abh^2ct &= 0 \\
 u_{yy} : \quad \frac{h^2}{2}b^2cs + \frac{h^2}{2}b^2cz + \frac{h^2}{2}B^2c_R + \frac{h^2}{2}b^2ct &= -1
 \end{aligned}$$

Bei zahlenmäßig gegebenen Werten für  $a, b, c, h$  und  $\psi$  ist das Gleichungssystem numerisch lösbar. Mit den erhaltenen Koeffizienten lautet die Differenzenapproximation der Poisson-Gleichung

$$c_P u_P + c_S u_S + c_Z u_Z + c_R u_R + c_T u_T + c_n \beta = f_P, \quad (10.28)$$

wo wir bereits für die linke Seite der Cauchy-Randbedingung den bekannten Wert  $\beta$  gemäß (10.8) eingesetzt haben. Die Randbedingung im Punkt  $P$  ist einerseits implizit in den Koeffizienten der  $u$ -Werte der Differenzengleichung (10.28) und andererseits im konstanten Beitrag  $c_n \beta$  berücksichtigt.

In der Differenzengleichung (10.28) wird im Allgemeinen  $c_Z \neq 0$  sein. Ist der Gitterpunkt  $Z$  ein regelmäßiger innerer Punkt, wie dies nach Abb. 10.9 anzunehmen ist, dann ist für ihn die Fünf-Punkte-Differenzengleichung (10.12) anwendbar. In ihr tritt der Funktionswert  $u_P$  nicht auf, und somit wird die Matrix  $\mathbf{A}$  des Gleichungssystems auf jeden Fall *unsymmetrisch*. Denn nehmen wir an, der Punkt  $P$  erhalte die Nummer  $i$  und  $Z$  die Nummer  $j \neq i$ . Dann ist in der Tat das Matrixelement  $a_{ij} \neq 0$ , aber das dazu symmetrische  $a_{ji} = 0$ .  $\triangle$

**Beispiel 10.5.** Wir betrachten im Gebiet  $G$  der Abb. 10.10 die Randwertaufgabe

$$\begin{aligned}
 -\Delta u &= 2 && \text{in } G \\
 u &= 0 && \text{auf } CD \\
 \frac{\partial u}{\partial n} &= 0 && \text{auf } BC \text{ und } DA \\
 \frac{\partial u}{\partial n} + 2u &= -1 && \text{auf } AB.
 \end{aligned} \quad (10.29)$$

Die Randkurve  $AB$  ist ein Kreisbogen mit dem Radius  $r = 1$ , dessen Mittelpunkt der Schnittpunkt der Verbindungsgeraden  $DA$  und  $CB$  ist. Die Randwertaufgabe kann als Wärmeleitungsproblem interpretiert werden, bei dem die stationäre Temperaturverteilung  $u(x, y)$  im Querschnitt eines (langen) Behälters gesucht ist, in welchem eine konstante Wärmequellendichte vorhanden ist. Der Behälter enthält eine Röhre, in welcher Wärme entzogen wird. Aus Symmetriegründen genügt es, die Lösung im Gebiet  $G$  zu bestimmen.

Zur Diskretisierung der Aufgabe soll das in Abb. 10.10 eingezeichnete Netz mit der Gitterweite  $h = 1/3$  verwendet werden. Die resultierenden Gitterpunkte sind eingezeichnet, wobei diejenigen mit unbekanntem Funktionswert durch ausgefüllte Kreise markiert und bereits spaltenweise nummeriert sind.

Für die meisten Gitterpunkte sind die Differenzengleichungen (10.18) anwendbar. Eine Sonderbehandlung erfordern die Punkte 3, 6, 7, 10 und 11. Die Differenzengleichungen für die Punkte 6

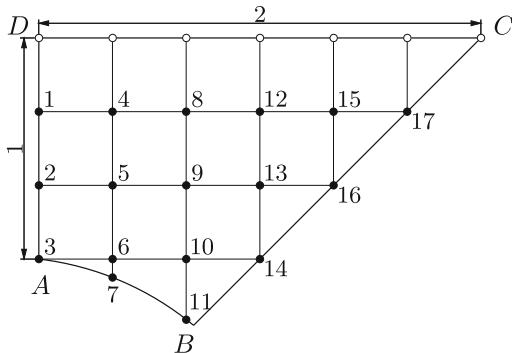


Abb. 10.10

Grundgebiet  $G$  der Randwertaufgabe mit Netz und Gitterpunkten,  $h = 1/3$ .

und 10 ergeben sich unmittelbar aus (10.23). Mit den einfach zu berechnenden, zur Schrittweite  $h$  relativen Abständen zwischen den Punkten 6 und 7 bzw. 10 und 11

$$b_6 = \overline{P_6 P_7}/h = 3 - 2\sqrt{2} \doteq 0.171573, \quad b_{10} = \overline{P_{10} P_{11}}/h = 3 - \sqrt{5} \doteq 0.763932$$

ergeben sich mit  $a = 1$  aus (10.23) die beiden Differenzengleichungen

$$\begin{aligned} -u_3 - 1.70711u_5 + 13.65685u_6 - 9.94975u_7 - u_{10} &= \frac{2}{9} \\ -u_6 - 1.13383u_9 + 4.61803u_{10} - 1.48420u_{11} - u_{14} &= \frac{2}{9} \end{aligned} \quad (10.30)$$

Im Punkt 3 stoßen zwei Randstücke aneinander, an denen eine Neumann-, bzw. eine Cauchy-Randbedingung zu erfüllen ist. Wir behandeln diese Situation entsprechend den Beispielen 10.3 und 10.4, wobei die wesentliche Vereinfachung vorliegt, dass im Punkt 3 bezüglich des Randstückes  $DA$ :  $\frac{\partial u}{\partial n} = -u_x$  und bezüglich  $AB$ :  $\frac{\partial u}{\partial n} = -u_y$  gilt. Es genügt hier, fünf Größen zur Approximation des Differenzialausdrucks  $-\Delta u$  heranzuziehen, nämlich

3 : $u(x, y) = u$	$c_3$
2 : $u(x, y + h) = u + hu_y +$	$\frac{h^2}{2}u_{yy} + \dots$ $c_2$
6 : $u(x + h, y) = u + hu_x + \frac{h^2}{2}u_{xx} + \dots$	$c_6$
$3_1 : \frac{\partial u(x, y)}{\partial n_1} = -u_x$	$c_n$
$3_2 : \frac{\partial u}{\partial n_2} + 2u = 2u - u_y$	$c_m$

Durch Koeffizientenvergleich erhält man

$$c_2 = \frac{-2}{h^2}, \quad c_6 = \frac{-2}{h^2}, \quad c_n = \frac{-2}{h}, \quad c_m = \frac{-2}{h}, \quad c_3 = \frac{4}{h^2} + \frac{4}{h},$$

und damit nach Multiplikation mit  $\frac{1}{2}h^2$  die Differenzengleichung

$$2(1 + h)u_3 - u_2 - u_6 - h \left( \frac{\partial u}{\partial n_1} \right) - h \left( \frac{\partial u}{\partial n_2} + 2u \right) = h^2.$$

Unter Berücksichtigung der beiden verschiedenen Randbedingungen im Punkt 3 ergibt sich die Drei-Punkte-Differenzengleichung

$$-u_2 + 2(1 + h)u_3 - u_6 = h^2 - h. \quad (10.31)$$

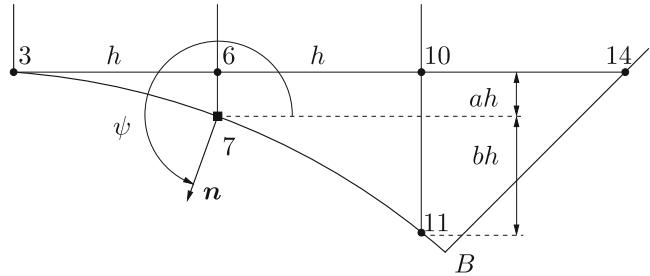


Abb. 10.11 Zur Herleitung der Differenzengleichung im Randpunkt 7.

Die Differenzengleichung (10.31) ist im Punkt  $A$  des Gebietes  $G$  unter den gegebenen Randbedingungen für beliebige Gitterweiten  $h$  gültig.

Die Herleitung der Differenzengleichungen für die Randpunkte 7 und 11 erfolgt nach dem im Beispiel 10.4 beschriebenen Vorgehen. In Abb. 10.11 ist die Situation für den Punkt 7 mit den für die Differenzengleichung verwendeten umliegenden Gitterpunkten dargestellt. Der Winkel  $\psi$  zwischen der positiven  $x$ -Richtung und der Normalenrichtung  $n$  beträgt  $\psi \doteq 250.53^\circ$ , und die benötigten trigonometrischen Funktionswerte sind  $\cos \psi = -1/3$  und  $\sin \psi = -2\sqrt{2}/3 \doteq -0.942809$ . Ferner sind  $a = b_6 = 3 - 2\sqrt{2} \doteq 0.171573$  und  $b = b_{10} - b_6 = 2\sqrt{2} - \sqrt{5} \doteq 0.592359$ . Mit diesen Zahlenwerten können die Taylor-Entwicklungen der Funktion in den fünf Punkten sowie der Ausdruck der linken Seite der Cauchy-Randbedingung bezüglich des Punktes 7 aufgeschrieben werden. Wenn noch die zweite und dritte Gleichung durch  $h$ , die vierte und sechste Gleichung durch  $h^2/2$  und die fünfte Gleichung durch  $h^2$  dividiert werden, dann erhalten wir das folgende Gleichungssystem für die gesuchten sechs Parameter.

$c_7$	$c_3$	$c_6$	$c_{10}$	$c_{11}$	$c_n$	1
1	1	1	1	1	2	0
0	-1	0	1	1	-1	0
0	0.171573	0.171573	0.171573	-0.592359	-2.828427	0
0	1	0	1	1	0	-18
0	-0.171573	0	0.171573	-0.592359	0	0
0	0.0294373	0.0294373	0.0294373	0.350889	0	-18

Daraus resultieren die Werte  $c_7 \doteq 597.59095$ ,  $c_3 \doteq 6.33443$ ,  $c_6 \doteq -518.25323$ ,  $c_{10} \doteq -17.44645$ ,  $c_{11} \doteq -6.88798$ ,  $c_n \doteq -30.66886$ . Wenn wir den vorgeschriebenen Wert -1 der Cauchy-Randbedingung berücksichtigen und die Differenzengleichung mit  $h^2 = 1/9$  multiplizieren, erhalten wir für den Punkt 7 die Differenzengleichung

$$0.70383u_3 - 57.58369u_6 + 66.39899u_7 - 1.93849u_{10} - 0.76533u_{11} = -3.18543$$

Auffällig an dieser Differenzengleichung sind die betragsmäßig großen Koeffizienten der Funktionswerte  $u_7$  und  $u_6$ . Dies wird einerseits durch den kleinen Abstand von Punkt 6 zum Punkt 7 und andererseits durch die Cauchy-Randbedingung verursacht.

Für den verbleibenden Gitterpunkt 11 sind in Abb. 10.12 diejenigen Punkte markiert, deren Funktionswerte verwendet werden zur Approximation des Differenzialausdrucks. Für den Winkel  $\psi$  gelten jetzt  $\cos \psi = -2/3$  und  $\sin \psi = -\sqrt{5}/3 \doteq -0.745356$ . Aus dem für die gesuchten Koeffizienten

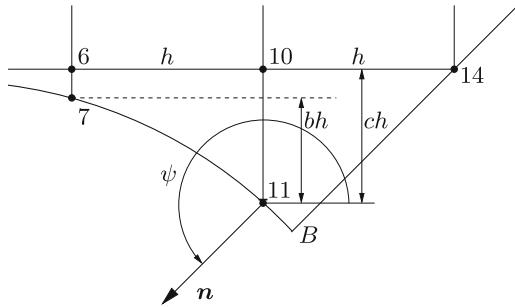


Abb. 10.12

Zur Herleitung der Differenzengleichung im Randpunkt 11.

$c_{11}, c_7, c_6, c_{10}, c_{14}$  und  $c_n$  analog hergeleiteten Gleichungssystem

$c_{11}$	$c_7$	$c_6$	$c_{10}$	$c_{14}$	$c_n$	1
1	1	1	1	1	2	0
0	-1	-1	0	1	-2	0
0	0.592359	0.763932	0.763932	0.763932	-2.236068	0
0	1	1	0	1	0	-18
0	-0.592359	-0.763932	0	0.763932	0	0
0	0.350889	0.583592	0.583592	0.583592	0	-18

ergibt sich nach Berücksichtigung der Cauchy-Randbedingung die Differenzengleichung für den Punkt 11

$$-7.04997u_6 + 6.81530u_7 + 1.29050u_{10} + 2.24016u_{11} - 1.76533u_{14} = -0.54311$$

Nach diesen Vorbereitungen für die unregelmäßigen randnahen und für die auf dem Kreisbogen liegenden Punkte kann das System der Differenzengleichungen für die Randwertaufgabe (10.29) formuliert werden. Es ist in (10.32) auf Seite 443 zu finden. Aus Platzgründen sind die nichtganzzahligen Koeffizienten auf drei Dezimalstellen angegeben. Das Gleichungssystem wurde mit voller Stellenzahl gelöst.

Die Koeffizientenmatrix des Gleichungssystems ist nicht symmetrisch und besitzt nicht einmal eine symmetrische Besetzungsstruktur, da beispielsweise  $a_{73} \neq 0$ , aber  $a_{37} = 0$  sind. Die Matrix hat Bandstruktur, wobei die linksseitige Bandbreite wegen der elften Gleichung  $m_1 = 5$  und die rechtsseitige Bandbreite  $m_2 = 4$  betragen. Obwohl die Matrix nicht diagonal dominant ist (Gleichung 11!), kann das System mit dem Gauß-Algorithmus unter Verwendung der Diagonalstrategie problemlos gelöst werden. Der Prozess läuft vollständig im Band ab, so dass eine rechen- und speicherökonomische Technik in Analogie zu derjenigen von Abschnitt 2.3.2 angewandt werden kann.

Die auf vier Stellen nach dem Komma gerundete Lösung des Gleichungssystems ist in der ungefähren Anordnung der Gitterpunkte zusammen mit den Randwerten am oberen Rand in (10.33) zusammengestellt.

$$\begin{array}{ccccccc}
 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0.2912 & 0.3006 & 0.3197 & 0.3272 & 0.2983 & 0.2047 & \\
 0.3414 & 0.3692 & 0.4288 & 0.4687 & 0.4391 & & \\
 0.1137 & 0.1840 & 0.3353 & 0.4576 & & & \\
 0.1217 & 0.1337 & & & & &
 \end{array} \tag{10.33}$$

(10.32)

$u_1$	$u_2$	$u_3$	$u_4$	$u_5$	$u_6$	$u_7$	$u_8$	$u_9$	$u_{10}$	$u_{11}$	$u_{12}$	$u_{13}$	$u_{14}$	$u_{15}$	$u_{16}$	$u_{17}$	r. S.
2 -0.5	-1																0.111
-0.5 2 -0.5		-1															0.111
-1 2.67			-1														-0.222
-1	4 -1		-1														0.222
-1	-1 4 -1			-1													0.222
-1	-1.71 13.7 -9.95				-1												0.222
0.704		-57.6 66.4				-1											-3.19
	-1	4 -1		-1	4 -1		-1		-1		-1		-1		-1		0.222
	-1	-1 4 -1			-1.13 4.62 -1.48			-1		-1		-1		-1			0.222
		-1				1.29 2.24				-1.77							0.222
		-7.05 6.82															-0.543
			-1		4 -1		-1		-1		-1		-1		-1		0.222
			-1		-1 4 -1			-1		-1		-1		-1			0.222
				-1		-1 2											0.111
							-1		4 -1	-1	-1						0.222
							-1		-1	2							0.111
									-1		-1		2				0.111

Die Ergebnisse vermitteln ein anschauliches Bild von der Temperaturverteilung, die im Innern des Gebietes ein Maximum annimmt und gegen den Kreisbogen hin abnimmt infolge des Wärmeabflusses.  $\triangle$

Die Herleitung von Differenzengleichungen für Randpunkte mit Neumann- oder Cauchy-Randbedingung ist mühsam und fehleranfällig. Deshalb wird man die Schritte normalerweise von einem Rechner ausführen lassen. Noch eleganter ist die Benutzung leistungsfähiger Softwarepakete, die auch über die Möglichkeit graphischer Eingabe verfügen, siehe Abschnitt 10.4 oder die Beispiele 10.12 und 10.13.

#### 10.1.4 Diskretisierungsfehler

Die berechneten Funktionswerte in den Gitterpunkten als Lösung des linearen Systems von Differenzengleichungen stellen selbstverständlich nur Näherungen für die exakten Werte der Lösungsfunktion der gestellten Randwertaufgabe dar. Um für den Fehler wenigstens qualitative Abschätzungen zu erhalten, bestimmen wir den *lokalen Diskretisierungsfehler der verwendeten Differenzenapproximation*. Der wird wie ein *Residuum* gebildet und sollte nicht verwechselt werden mit dem Diskretisierungsfehler der Lösung in einem Punkt  $u_P - u(P)$ . Den folgenden Betrachtungen legen wir die Poisson-Gleichung und die bisher verwendeten Differenzengleichungen zu Grunde. Analog zu den gewöhnlichen Differenzialgleichungen versteht man unter dem lokalen Diskretisierungsfehler einer Differenzengleichung den Wert, der bei Substitution der exakten Lösung  $u(x, y)$  der Differenzialgleichung in die Differenzengleichung resultiert. Für die Fünf-Punkte-Differenzengleichung (10.12) eines regelmäßigen inneren Gitterpunktes  $P(x, y)$  ist er definiert als

$$d_P := \frac{1}{h^2}[-u(x, y + h) - u(x - h, y) - u(x, y - h) - u(x + h, y) + 4u(x, y)] - f(x, y). \quad (10.34)$$

Für die Funktion  $u(x, y)$  gelten die Taylor-Entwicklungen

$$\begin{aligned} u(x \pm h, y) &= \\ u \pm hu_x + \frac{1}{2}h^2u_{xx} \pm \frac{1}{6}h^3u_{xxx} + \frac{1}{24}h^4u_{xxxx} &\pm \frac{h^5}{120}u_{xxxxx} + \frac{h^6}{720}u_{xxxxxx} \pm \dots \end{aligned} \quad (10.35)$$

$$\begin{aligned} u(x, y \pm h) &= \\ u \pm hu_y + \frac{1}{2}h^2u_{yy} \pm \frac{1}{6}h^3u_{yyy} + \frac{1}{24}h^4u_{yyyy} &\pm \frac{h^5}{120}u_{yyyyy} + \frac{h^6}{720}u_{yyyyyy} \pm \dots \end{aligned}$$

Dabei sind die Werte von  $u$  und der partiellen Ableitungen an der Stelle  $(x, y)$  zu verstehen. Nach ihrer Substitution in (10.34) ergibt sich

$$d_P = [-u_{xx} - u_{yy} - f(x, y)]_P - \frac{h^2}{12}[u_{xxxx} + u_{yyyy}]_P - \frac{h^4}{360}[u_{xxxxxx} + u_{yyyyyy}]_P + \dots$$

Da  $u$  die Poisson-Gleichung erfüllt, verschwindet der erste Klammerausdruck. Der lokale Diskretisierungsfehler der Differenzengleichung in einem regelmäßigen inneren Gitterpunkt

ist somit gegeben durch

$$d_P = -\frac{h^2}{12}[u_{xxxx} + u_{yyyy}]_P - \frac{h^4}{360}[u_{xxxxxxxx} + u_{yyyyyy}]_P - \dots \quad (10.36)$$

Wenn wir im Moment nur den Hauptteil des lokalen Diskretisierungsfehlers betrachten, so besagt (10.36), dass  $d_P = O(h^2)$  ist. Diese Aussage behält ihre Gültigkeit auch für einen Randpunkt mit der Neumann-Randbedingung  $\partial u / \partial n = 0$ , falls der Rand entweder eine Netzgerade oder eine Diagonale des Netzes ist.

Für einen unregelmäßigen, randnahen Gitterpunkt  $P$  nach Abb. 10.7 ist auf Grund der zu (10.35) analogen Taylor-Entwicklungen und der in Beispiel 10.2 durchgeführten Herleitung sofort ersichtlich, dass in der Darstellung des lokalen Diskretisierungsfehlers die dritten partiellen Ableitungen, multipliziert mit der Gitterweite  $h$ , auftreten. Der Hauptteil des lokalen Diskretisierungsfehlers ist folglich proportional zu  $h$ , d.h. es ist  $d_P = O(h)$ . Dasselbe trifft auch zu für die Differenzengleichungen von Randpunkten, die wir in den Beispielen 10.3 bis 10.5 angetroffen haben, da in allen jenen Fällen die dritten partiellen Ableitungen im lokalen Diskretisierungsfehler stehen bleiben.

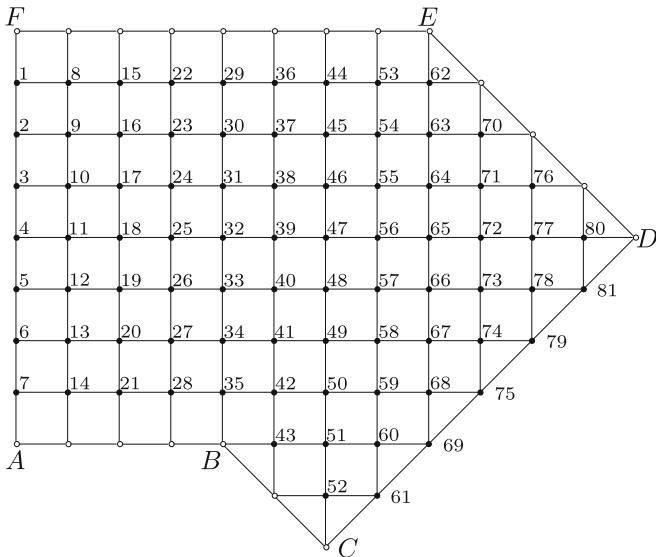
Hier kann wie bei Anfangswertproblemen zu gewöhnlichen Differenzialgleichungen gezeigt werden, dass der globale Diskretisierungsfehler  $e_P := u(x, y) - u_P$  dieselbe Ordnung in  $h$  besitzt wie der lokale. Die *Fehlerordnung* der Fünf-Punkte-Formel (10.12) ist somit zwei. Gleichzeitig gilt damit, dass die Näherungslösungen in den Gitterpunkten für  $h \rightarrow 0$  gegen die exakten Werte der Randwertaufgabe konvergieren.

Konvergenz kann auch bewiesen werden, wenn Differenzengleichungen vorkommen, deren lokaler Diskretisierungsfehler nur  $O(h)$  ist.

Es sei aber nochmal betont, dass das Konvergenzverhalten nur unter der Voraussetzung gilt, dass die Lösungsfunktion in  $\bar{G} = G \cup \Gamma$  mindestens viermal stetig differenzierbar ist. Dies trifft beispielsweise dann nicht zu, wenn Dirichlet-Randbedingungen unstetig sind oder das Gebiet einspringende Ecken aufweist. Partielle Ableitungen niedriger Ordnung der Lösungsfunktion besitzen an diesen Stellen eine Singularität, ein einfaches Beispiel ist der Kreissektor mit Innenwinkel  $\varphi > \pi$ , siehe etwa § 2 in [Bra 03] oder [Tve 02]. Solche Fälle erfordern spezielle Analysen, um das Konvergenzverhalten zu erfassen [Bra 68]. Bei der numerischen Lösung solcher Probleme ist es oft vorteilhaft, die Singularitäten durch geeignete Ansätze zu berücksichtigen [Gla 79, Mit 80].

**Beispiel 10.6.** Zur Steigerung der Genauigkeit der Näherungslösung der Differenzengleichungen kann man die Gitterweite  $h$  verkleinern. Auf Grund der Fehlerordnung  $O(h^2)$  verkleinert sich der Fehler bei Halbierung der Gitterweite  $h$  nur etwa auf den vierten Teil. Die Zahl der Gitterpunkte und damit die Ordnung des linearen Gleichungssystems steigt etwa auf das Vierfache an. Zur Illustration behandeln wir die Randwertaufgabe (10.14) bis (10.17) von Beispiel 10.1 mit der Gitterweite  $h = 0.125$  und erhalten gemäß Abb. 10.13  $n = 81$  Gitterpunkte mit unbekannten Funktionswerten. Es können die Differenzengleichungen (10.18) angewandt werden. Das Gleichungssystem erhält bei spaltenweiser Nummerierung der Gitterpunkte eine Bandstruktur mit der Bandbreite  $m = 9$ .

Zu Vergleichszwecken sind in (10.37) die aus dem Gleichungssystem resultierenden, auf fünf Stellen nach dem Komma gerundeten Näherungswerte an den Gitterpunkten von Abb. 10.6 in deren

Abb. 10.13 Netz und Gitterpunkte für  $h = 0.125$ .

Anordnung zusammengestellt.

$$\begin{array}{cccccc}
 0 & 0 & 0 & 0 & 0 & \\
 0.41771 & 0.41178 & 0.39070 & 0.34227 & 0.23286 & 0 \\
 0.72153 & 0.71270 & 0.68149 & 0.61400 & 0.48858 & 0.28386 & 0 \\
 0.91686 & 0.90979 & 0.88268 & 0.81815 & 0.70244 & 0.52389 & \\
 1 & 1 & 1 & 0.94836 & 0.85602 & \\
 & & & 1 & &
 \end{array} \tag{10.37}$$

Eine Gegenüberstellung mit dem Ergebnis (10.20) für die doppelt so große Gitterweite zeigt eine recht gute Übereinstimmung. Die größte Differenz beträgt maximal acht Einheiten in der dritten Dezimalstelle nach dem Komma. Wenn wir trotz der Ecken des Gebietes die Fehlerordnung zwei annehmen, dann zeigt sich auf Grund einer Extrapolation, dass (10.37) die gesuchte Lösung mit mindestens zweistelliger Genauigkeit darstellt.  $\triangle$

### 10.1.5 Ergänzungen

Jede Verkleinerung der Gitterweite  $h$  bewirkt eine starke Vergrößerung der Zahl der Unbekannten. Es kommt hinzu, dass die Konditionszahl der Matrix  $A$  des Systems von Differenzengleichungen wie  $h^{-2}$  anwächst [Hac 96]. Dies sieht man beispielhaft, wenn man die Spektralnorm von Modellproblemen berechnet, siehe etwa Beispiel 11.16. Eine andere Möglichkeit zur Erhöhung der Genauigkeit der Näherungslösung besteht darin, die Fehlerordnung der Differenzenapproximation zu erhöhen. Zur Bildung der betreffenden Differenzengleichung müssen Funktionswerte an mehr Gitterpunkten verwendet werden. Werden auch zur Bildung der rechten Seite mehrere Werte der Funktion  $f(x, y)$  verwendet, so spricht man auch von *Mehrstellenoperatoren*, siehe etwa [Hac 96].

Die Diskretisierung einer allgemeinen partiellen Differentialgleichung (10.1) vom elliptischen Typus erfolgt nach dem oben vorgezeichneten Vorgehen. Im einfachsten Fall werden die auftretenden partiellen Ableitungen gemäß (10.9) und (10.10) durch Differenzenquotienten approximiert. In einem regelmäßigen inneren Punkt verwendet man für die gemischte partielle Ableitung die Approximation

$$u_{xy}(u_i, y_j) \approx \frac{u_{i+1,j+1} - u_{i-1,j+1} - u_{i+1,j-1} + u_{i-1,j-1}}{4h^2},$$

welche aus zweimaliger Anwendung der zentralen Differenzenquotienten (10.9) resultiert. Damit erhält die einfachste Differenzenapproximation die Struktur einer Neun-Punkte-Formel, wenn in der zu lösenden partiellen Differentialgleichung der Term  $u_{xy}$  auftritt. Wir verweisen wegen weiterer Einzelheiten wieder auf [Hac 96], wo man auch die Anwendung von Differenzenapproximationen auf elliptische Randwertaufgaben höherer Ordnung am Beispiel der *biharmonischen* oder *Plattengleichung*  $\Delta^2 u = f$  findet.

Zur numerischen Lösung von elliptischen Randwertaufgaben ist es in bestimmten Situationen zweckmäßig, ein Netz mit variablen Gitterweiten in  $x$ - und  $y$ -Richtung zu verwenden, um auf diese Weise eine lokal feinere Diskretisierung zu erreichen. Dies ist angezeigt in Teilgebieten, in denen sich die Lösungsfunktion rasch ändert oder die Ableitungen eine Singularität aufweisen, beispielsweise in der Nähe einer einspringenden Ecke. Für das Gebiet  $G$  der Randwertaufgabe (10.14) bis (10.17) trägt beispielsweise das Netz von Abb. 10.14 der einspringenden Ecke  $B$  Rechnung. Die in  $y$ -Richtung in der Nähe von  $B$  gewählte Gitterweite  $h = 1/16$  hat wegen der Neumann-Randbedingung längs  $CD$  eine kleine Gitterweite in einer Region zur Folge, wo es nicht erforderlich wäre.

Zur Approximation der Poisson-Gleichung sind jetzt die Funktionswerte in nicht gleichabständigen Gitterpunkten nach Abb. 10.14 zu verwenden. Aus (10.21) und (10.22) lassen sich für die zweiten partiellen Ableitungen in  $P$  die folgenden Näherungen herleiten

$$u_{xx}(P) \approx 2 \left\{ \frac{u_E}{h_1(h_1 + h_2)} + \frac{u_W}{h_2(h_1 + h_2)} - \frac{u_P}{h_1 h_2} \right\},$$

$$u_{yy}(P) \approx 2 \left\{ \frac{u_N}{h_3(h_3 + h_4)} + \frac{u_S}{h_4(h_3 + h_4)} - \frac{u_P}{h_3 h_4} \right\}.$$

Daraus folgt die Differenzengleichung für den typischen Punkt  $P$

$$\begin{aligned} 2 \left( \frac{1}{h_1 h_2} + \frac{1}{h_3 h_4} \right) u_P &- \frac{2u_N}{h_3(h_3 + h_4)} - \frac{2u_W}{h_2(h_1 + h_2)} \\ &- \frac{2u_S}{h_4(h_3 + h_4)} - \frac{2u_E}{h_1(h_1 + h_2)} = f_P. \end{aligned} \tag{10.38}$$

Der lokale Diskretisierungsfehler der Differenzengleichung (10.38) ist nur  $O(h)$ , falls  $h = \max\{h_1, h_2, h_3, h_4\}$  und  $h_1 \neq h_2$  oder  $h_3 \neq h_4$  ist. Das resultierende System von Differenzengleichungen ist unsymmetrisch.

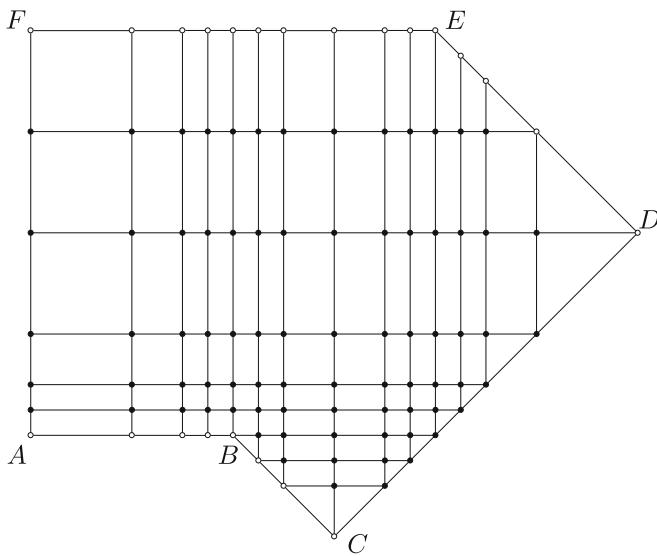
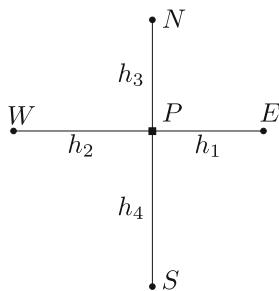


Abb. 10.14 Gebiet mit unregelmäßigem Netz.

Abb. 10.15  
Gitterpunkte im unregelmäßigen Netz.

## 10.2 Parabolische Anfangsrandwertaufgaben

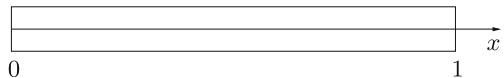
Die mathematische Beschreibung von zeitabhängigen Diffusions- und Wärmeleitungsproblemen führt auf eine parabolische Differenzialgleichung für die gesuchte, von Zeit und von Ortsvariablen abhängige Funktion. Wir behandeln zuerst ausführlich den eindimensionalen Fall, beschreiben zu seiner Lösung zwei Diskretisierungsmethoden mit unterschiedlichen Eigenschaften und betrachten anschließend noch den zweidimensionalen Fall.

### 10.2.1 Eindimensionale Probleme, explizite Methode

Die einfachste parabolische Differenzialgleichung lautet

$$u_t = u_{xx} \quad (10.39)$$

Abb. 10.16  
Wärmeleitung im Stab.



für eine Funktion  $u(x, t)$  der Ortsvariablen  $x$  und der Zeit  $t$ . In der Regel wird die Funktion  $u(x, t)$  gesucht in einem beschränkten Intervall für  $x$ , das wir auf  $(0, 1)$  normieren können, und für positive Werte von  $t$ . Das Gebiet  $G$ , in welchem die Lösung zu bestimmen ist, besteht somit aus einem unendlichen Halbstreifen in der  $(x, t)$ -Ebene. Zur Differenzialgleichung (10.39) treten noch Nebenbedingungen hinzu, die man in zwei Klassen einteilt. So muss eine *Anfangsbedingung*

$$u(x, 0) = f(x), \quad 0 < x < 1, \quad (10.40)$$

gegeben sein, welche die Werte der Lösungsfunktion zur Zeit  $t = 0$  vorschreibt. Weiter müssen sowohl für  $x = 0$  als auch für  $x = 1$  für alle  $t > 0$  *Randbedingungen* vorliegen. Entweder wird der Wert von  $u$  als Funktion der Zeit  $t$  vorgeschrieben (Dirichlet-Randbedingung) oder eine Linearkombination der partiellen Ableitung von  $u$  nach  $x$  und der Funktion  $u$  muss einen im allgemeinen zeitabhängigen Wert annehmen (Cauchy-Randbedingung). Die Randbedingungen können beispielsweise so lauten

$$u(0, t) = \varphi(t), \quad u_x(1, t) + \alpha(t)u(1, t) = \beta(t), \quad t > 0, \quad (10.41)$$

wo  $\varphi(t)$ ,  $\alpha(t)$  und  $\beta(t)$  gegebenen Funktionen der Zeit sind.

Die Anfangsrandwertaufgabe (10.39) bis (10.41) wird nun analog zu den elliptischen Randwertaufgaben diskretisiert, indem zuerst über das Grundgebiet  $G = [0, 1] \times [0, \infty)$  ein Netz mit zwei im Allgemeinen unterschiedlichen Gitterweiten  $h$  und  $k$  in  $x$ - und  $t$ -Richtung gelegt wird. Gesucht werden dann Näherungen der Funktion  $u(x, t)$  in den so definierten diskreten Gitterpunkten. Weiter wird die Differenzialgleichung durch eine Differenzenapproximation ersetzt, wobei gleichzeitig die Randbedingungen berücksichtigt werden. Mit ihrer Hilfe wird die Funktion  $u(x, t)$  näherungsweise mit zunehmender Zeit  $t$  berechnet werden.

**Beispiel 10.7.** Wir betrachten die Wärmeleitung in einem homogenen Stab konstanten Querschnitts mit der Länge Eins. Er sei auf der ganzen Länge wärmeisoliert, so dass keine Wärmeabstrahlung stattfinden kann. An seinem linken Ende (vgl. Abb. 10.16) ändere sich die Temperatur periodisch, während das rechte Ende wärmeisoliert sei. Gesucht wird die Temperaturverteilung im Stab in Abhängigkeit des Ortes  $x$  und der Zeit  $t$  falls zur Zeit  $t = 0$  die Temperaturverteilung bekannt ist.

Die Anfangsrandwertaufgabe für die Temperaturverteilung  $u(x, t)$  lautet:

$$\begin{aligned} u_t &= u_{xx} && \text{für } 0 < x < 1, t > 0; \\ u(x, 0) &= 0 && \text{für } 0 < x < 1; \\ u(0, t) &= \sin(\pi t), \quad u_x(1, t) = 0 && \text{für } t > 0. \end{aligned} \quad (10.42)$$

In Abb. 10.17 ist das Netz in einem Teil des Halbstreifens für die Gitterweiten  $h = 1/n$  und  $k$  eingezeichnet. Die Gitterpunkte, in denen die Funktionswerte entweder durch die Anfangs- oder die Randbedingung bekannt sind, sind durch Kreise markiert, während die Gitterpunkte mit unbekannten Funktionswerten durch ausgefüllte Kreis hervorgehoben sind. Die Gitterpunkte haben die Koordinaten  $x_i = ih, i = 0, 1, \dots, n$ , und  $t_j = jk, j = 0, 1, 2, \dots$ . Die Näherungswerte für die

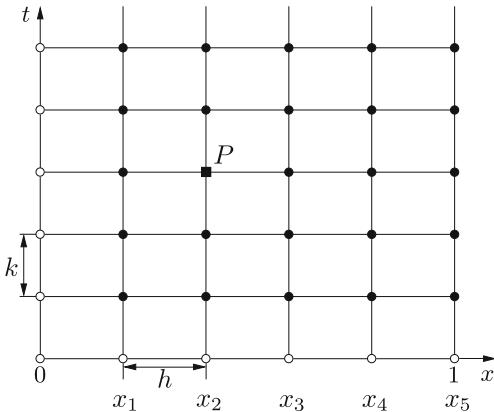


Abb. 10.17  
Netz im Halbstreifen.

gesuchten Funktionswerte  $u(x_i, t_j)$  bezeichnen wir mit  $u_{i,j}$ . Zur Approximation der partiellen Differenzialgleichung in einem inneren Punkt  $P(x_i, t_j)$  ersetzen wir die erste partielle Ableitung nach  $t$  durch den so genannten Vorrücksdifferenzenquotienten

$$u_t(P) \approx \frac{u_{i,j+1} - u_{i,j}}{k}$$

und die zweite partielle Ableitung nach  $x$  durch den zweiten Differenzenquotienten

$$u_{xx}(P) \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2}.$$

Durch Gleichsetzen der beiden Ausdrücke resultiert die Differenzengleichung

$$u_{i,j+1} - u_{i,j} = \frac{k}{h^2} (u_{i+1,j} - 2u_{i,j} + u_{i-1,j}),$$

oder

$$u_{i,j+1} = ru_{i-1,j} + (1 - 2r)u_{i,j} + ru_{i+1,j}, \quad r := \frac{k}{h^2},$$

$$i = 1, 2, \dots, n-1; \quad j = 0, 1, 2, \dots$$

(10.43)

Die Berücksichtigung der Randbedingung am linken Rand ist problemlos, da für  $i = 1$  in (10.43) der bekannte Wert  $u_{0,j} = \sin(\pi jk)$  eingesetzt werden kann. Die Neumann-Randbedingung am rechten Rand wird durch eine Symmetriebetrachtung berücksichtigt, so dass aus (10.43) die Formel folgt

$$u_{n,j+1} = 2ru_{n-1,j} + (1 - 2r)u_{n,j}, \quad j = 0, 1, 2, \dots$$

(10.44)

Zur Zeit  $t = 0$ , d.h. für  $j = 0$ , sind die Funktionswerte  $u_{i,0}$  für  $i = 0, 1, \dots, n$  durch die Anfangsbedingung bekannt. Die Rechenvorschriften (10.43) und (10.44) gestatten, die Näherungen  $u_{i,j+1}$ ,  $i = 1, 2, \dots, n$ , für festes  $j$  aus den Werten  $u_{i,j}$  in expliziter Weise zu berechnen. Somit kann die Näherungslösung mit zunehmendem  $j$ , also in Zeitrichtung fortschreitend, sukzessive ermittelt werden. Die angewandte Diskretisierung der parabolischen Differenzialgleichung führt zur *expliziten Methode von Richardson*.

Wir berechnen Näherungslösungen der Anfangsrandwertaufgabe (10.42) mittels (10.43) und (10.44) für feste Gitterweite  $h = 0.1$  und die Zeitschrittweiten  $k = 0.002$ ,  $k = 0.005$  und  $k = 0.01$ . In Tab. 10.1 bis 10.3 sind die erhaltenen Ergebnisse auszugsweise zusammengestellt.

In den beiden ersten Fällen ( $k = 0.002$  und  $k = 0.005$ ) erhält man qualitativ richtige Näherungen,

Tab. 10.1 Wärmeleitung,  $h = 0.1, k = 0.002, r = 0.2$ ; explizite Methode.

$t$	$j$	$u_{0,j}$	$u_{1,j}$	$u_{2,j}$	$u_{3,j}$	$u_{4,j}$	$u_{6,j}$	$u_{8,j}$	$u_{10,j}$
0	0	0	0	0	0	0	0	0	0
0.1	50	0.3090	0.2139	0.1438	0.0936	0.0590	0.0212	0.0069	0.0035
0.2	100	0.5878	0.4580	0.3515	0.2657	0.1980	0.1067	0.0599	0.0456
0.3	150	0.8090	0.6691	0.5476	0.4441	0.3578	0.2320	0.1611	0.1383
0.4	200	0.9511	0.8222	0.7050	0.6009	0.5107	0.3727	0.2909	0.2639

Tab. 10.2 Wärmeleitung,  $h = 0.1, k = 0.005, r = 0.5$ ; explizite Methode.

$t$	$j$	$u_{0,j}$	$u_{1,j}$	$u_{2,j}$	$u_{3,j}$	$u_{4,j}$	$u_{6,j}$	$u_{8,j}$	$u_{10,j}$
0	0	0	0	0	0	0	0	0	0
0.1	20	0.3090	0.2136	0.1430	0.0927	0.0579	0.0201	0.0061	0.0027
0.2	40	0.5878	0.4578	0.3510	0.2650	0.1970	0.1053	0.0583	0.0439
0.3	60	0.8090	0.6689	0.5472	0.4435	0.3569	0.2306	0.1594	0.1365
0.4	80	0.9511	0.8222	0.7049	0.6006	0.5101	0.3716	0.2895	0.2624
0.5	100	1.0000	0.9007	0.8049	0.7156	0.6350	0.5060	0.4263	0.3994
0.6	120	0.9511	0.8955	0.8350	0.7736	0.7147	0.6142	0.5487	0.5260
0.7	140	0.8090	0.8063	0.7904	0.7661	0.7376	0.6804	0.6387	0.6235
0.8	160	0.5878	0.6408	0.6737	0.6916	0.6985	0.6941	0.6828	0.6776
0.9	180	0.3090	0.4147	0.4954	0.5555	0.5992	0.6510	0.6731	0.6790
1.0	200	0	0.1497	0.2718	0.3699	0.4474	0.5528	0.6076	0.6245

Tab. 10.3 Wärmeleitung,  $h = 0.1, k = 0.001, r = 1.0$ ; explizite Methode.

$t$	$j$	$u_{0,j}$	$u_{1,j}$	$u_{2,j}$	$u_{3,j}$	$u_{4,j}$	$u_{6,j}$	$u_{8,j}$	$u_{10,j}$
0	0	0	0	0	0	0	0	0	0
0.05	5	0.1564	0.0312	0.1256	-0.0314	0.0314	0	0	0
0.10	10	0.3090	5.4638	-8.2955	8.8274	-6.7863	-2.0107	-0.1885	0

wobei im zweiten Fall wegen des größeren Wertes von  $k$  größere Fehler zu erwarten sind. Im dritten Fall mit  $k = 0.01$  braucht man nur wenige Schritte durchzuführen, um zu erkennen, dass die erhaltenen Ergebnisse sinnlos sind. Die explizite Methode ist für diese Kombination von Gitterweiten  $h$  und  $k$  mit  $r = 1.0$  offenbar instabil.  $\triangle$

Um die Eigenschaften der expliziten Methode von Richardson zu untersuchen, beginnen wir mit der Bestimmung des *lokalen Diskretisierungsfehlers* der Rechenvorschrift (10.43). Mit der Lösungsfunktion  $u(x, t)$  der Aufgabe (10.42) ist dieser definiert durch

$$\begin{aligned}
d_{i,j+1} &:= u(x_i, t_{j+1}) - ru(x_{i-1}, t_j) - (1-2r)u(x_i, t_j) - ru(x_{i+1}, t_j) \\
&= u + ku_t + \frac{1}{2}k^2u_{tt} + \dots \\
&\quad - r \left\{ u - hu_x + \frac{1}{2}h^2u_{xx} - \frac{1}{6}h^3u_{xxx} + \frac{1}{24}h^4u_{xxxx} \mp \dots \right\} \\
&\quad - (1-2r)u \\
&\quad - r \left\{ u + hu_x + \frac{1}{2}h^2u_{xx} + \frac{1}{6}h^3u_{xxx} + \frac{1}{24}h^4u_{xxxx} + \dots \right\} \\
&= k\{u_t - u_{xx}\} + \frac{1}{2}k^2u_{tt} - \frac{1}{2}kh^2u_{xxxx} + \dots,
\end{aligned}$$

worin wir  $k = rh^2$  verwendet haben. Der Koeffizient von  $k$  ist gleich null, weil  $u(x, t)$  die Differenzialgleichung erfüllt. Somit gilt für den lokalen Diskretisierungsfehler

$$d_{i,j+1} = \frac{1}{2}k^2u_{tt}(x_i, t_j) - \frac{1}{12}kh^2u_{xxxx}(x_i, t_j) + \dots = O(k^2) + O(kh^2). \quad (10.45)$$

Um weiter den *globalen Diskretisierungsfehler*  $g_{i,j+1}$  des Verfahrens abschätzen zu können, verwenden wir die Tatsache, dass mit der Methode eine Integration in Zeitrichtung erfolgt. Die Rechenvorschriften (10.43) und (10.44) entsprechen der *Methode von Euler* (8.12) zur Integration eines Systems von gewöhnlichen Differenzialgleichungen. Man gelangt zu diesem System, wenn man die partielle Differenzialgleichung nur bezüglich der Ortsvariablen  $x$  diskretisiert. Die zweite partielle Ableitung ersetzen wir dabei durch den zweiten Differenzenquotienten, berücksichtigen die Randbedingungen am linken und am rechten Rand und definieren die  $n$  Funktionen  $y_i(t) := u(x_i, t)$ ,  $(i = 1, 2, \dots, n)$ , zugehörig zu den diskreten Stellen  $x_i$ . Dann lautet das System von gewöhnlichen Differenzialgleichungen erster Ordnung

$$\begin{aligned}
\dot{y}_1(t) &= \frac{1}{h^2}\{-2y_1(t) + y_2(t) + \sin(\pi t)\}, \\
\dot{y}_i(t) &= \frac{1}{h^2}\{y_{i-1}(t) - 2y_i(t) + y_{i+1}(t)\}, \quad i = 2, 3, \dots, n-1, \\
\dot{y}_n(t) &= \frac{1}{h^2}\{2y_{n-1}(t) - 2y_n(t)\}.
\end{aligned} \quad (10.46)$$

Integriert man (10.46) mit der Methode von Euler mit dem Zeitschritt  $k$ , so resultieren (10.43) und (10.44). Auf Grund dieses Zusammenhangs erkennt man, dass der globale Fehler gegenüber dem lokalen eine Potenz in  $k$  verliert. Es gilt somit  $g_{i,j-1} = O(k) + O(h^2)$ . Die explizite Methode von Richardson ist von erster Ordnung bezüglich der Zeitintegration und zweiter Ordnung bezüglich der Ortsdiskretisation.

Es bleibt noch das zentrale Problem der *absoluten Stabilität* (vgl. Abschnitt 8.4) der expliziten Methode abzuklären. Zu diesem Zweck schreiben wir die Rechenvorschriften (10.43) und (10.44) unter Berücksichtigung der Randbedingung am linken Rand wie folgt:

$$\mathbf{u}_{j+1} = \mathbf{A}\mathbf{u}_j + \mathbf{b}_j, \quad j = 0, 1, 2, \dots \quad (10.47)$$

Darin bedeuten

$$\mathbf{A} := \begin{pmatrix} 1-2r & r & & & \\ r & 1-2r & r & & \\ & r & 1-2r & r & \\ & & \ddots & \ddots & \ddots \\ & & r & 1-2r & r \\ & & & 2r & 1-2r \end{pmatrix}, \quad (10.48)$$

$$\mathbf{u}_j := \begin{pmatrix} u_{1,j} \\ u_{2,j} \\ u_{3,j} \\ \vdots \\ u_{n-1,j} \\ u_{n,j} \end{pmatrix}, \quad \mathbf{b}_j := \begin{pmatrix} r \sin(\pi j k) \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}.$$

Die Matrix  $\mathbf{A}$  ist tridiagonal und ist durch den Parameter  $r$  von  $k$  und  $h$  abhängig. Notwendig und hinreichend für die absolute Stabilität ist die Bedingung, dass die Eigenwerte  $\lambda_\nu$  der Matrix  $\mathbf{A}$  betragsmäßig kleiner Eins sind. Um Aussagen über die Eigenwerte in Abhängigkeit von  $r$  zu gewinnen, setzen wir

$$\mathbf{A} = \mathbf{I} - r\mathbf{J} \text{ mit } \mathbf{J} := \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & \ddots & \ddots & \ddots \\ & & & -1 & 2 & -1 \\ & & & & -2 & 2 \end{pmatrix} \in \mathbb{R}^{n,n}. \quad (10.49)$$

Die Eigenwerte  $\lambda_\nu$  von  $\mathbf{A}$  sind durch die Eigenwerte  $\mu_\nu$  von  $\mathbf{J}$  gegeben durch  $\lambda_\nu = 1 - r\mu_\nu$ ,  $\nu = 1, 2, \dots, n$ . Die Eigenwerte von  $\mathbf{J}$  sind reell, denn  $\mathbf{J}$  ist ähnlich zu der symmetrischen Matrix  $\hat{\mathbf{J}} := \mathbf{D}^{-1}\mathbf{J}\mathbf{D}$  mit  $\mathbf{D} := \text{diag}(1, 1, \dots, 1, \sqrt{2})$ . Die Matrix  $\hat{\mathbf{J}}$  ist positiv definit, denn der Gauß-Algorithmus für  $\hat{\mathbf{J}}$  ist mit Diagonalstrategie mit positiven Pivotelementen durchführbar. Folglich sind die Eigenwerte von  $\mathbf{J}$  positiv und auf Grund der Zeilenmaximumnorm höchstens gleich vier. Die Matrix  $\hat{\mathbf{J}} - 4\mathbf{I}$  ist negativ definit, und somit ist der Wert vier nicht Eigenwert von  $\hat{\mathbf{J}}$ . Für die Eigenwerte von  $\mathbf{A}$  gilt folglich wegen  $r > 0$

$$1 - 4r < \lambda_\nu < 1,$$

und die Bedingung der absoluten Stabilität ist erfüllt, falls

$$r \leq \frac{1}{2} \text{ oder } k \leq \frac{1}{2}h^2$$

(10.50)

gilt. Im Beispiel 10.7 wurde mit  $h = 0.1, k = 0.001$  und  $r = 1$  die hinreichende Bedingung (10.50) klar verletzt, was die erhaltenen Zahlenwerte in Tab. 10.3 erklärt.

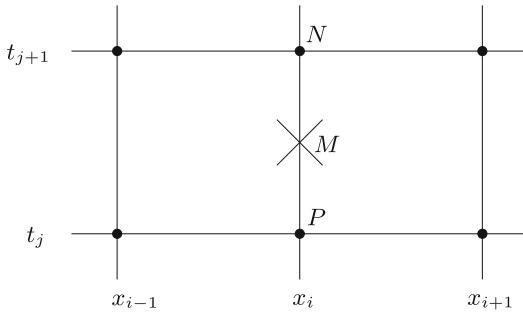


Abb. 10.18 Netzausschnitt.

Die Beschränkung des Zeitschrittes  $k$  durch (10.50) zur Sicherstellung der Stabilität der expliziten Methode ist für kleine Gitterweiten  $h$  sehr restriktiv. Zur Lösung der Anfangsrandwertaufgabe bis zu einem Zeitpunkt  $T \gg 1$  ist in diesem Fall eine derart große Anzahl von Schritten notwendig, dass der gesamte Rechenaufwand prohibitiv groß werden kann. Deshalb sind andersgeartete Differenzenapproximationen mit besseren Eigenschaften hinsichtlich der absoluten Stabilität nötig.

Die Untersuchung der absoluten Stabilität erfolgte für die konkrete Aufgabe (10.42). Die Bedingung (10.50) bleibt bei der Differenzialgleichung  $u_t = u_{xx}$  auch für andere Randbedingungen bestehen [Smi 85].

### 10.2.2 Eindimensionale Probleme, implizite Methode

Aus der Sicht der Differenzenapproximation ist bei der Herleitung der expliziten Methode nachteilig, dass die beiden verwendeten Differenzenquotienten die zugehörigen Ableitungen an verschiedenen Stellen des Gebietes  $G$  am besten approximieren. Um die Approximation unter diesem Gesichtspunkt zu verbessern, soll  $u_{xx}$  durch das arithmetische Mittel der beiden zweiten Differenzenquotienten ersetzt werden, welche zu den Punkten  $P(x_i, t_j)$  und  $N(x_i, t_{j+1})$  in zwei aufeinanderfolgenden Zeitschichten gebildet werden (vgl. Abb. 10.18). Damit erfolgt eine Approximation von  $u_t = u_{xx}$  bezüglich des Mittelpunktes  $M$ . Mit

$$\begin{aligned} u_{xx} &\approx \frac{1}{2h^2} \{u_{i+1,j} - 2u_{i,j} + u_{i-1,j} + u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}\} \\ u_t &\approx \frac{1}{k} \{u_{i,j+1} - u_{i,j}\} \end{aligned}$$

erhalten wir durch Gleichsetzen der beiden Differenzenapproximationen, nach Multiplikation mit  $2k$  und nachfolgendem Ordnen folgende Differenzengleichung für einen inneren Punkt  $P$ .

$$\begin{aligned} -ru_{i-1,j+1} + (2+2r)u_{i,j+1} - ru_{i+1,j+1} \\ = ru_{i-1,j} + (2-2r)u_{i,j} + ru_{i+1,j}; \quad r = \frac{k}{h^2}. \end{aligned} \tag{10.51}$$

Für die folgenden Betrachtungen legen wir die Aufgabe (10.42) zu Grunde. Die beiden Randbedingungen führen zu den zusätzlichen Differenzengleichungen

$$(2 + 2r) u_{1,j+1} - ru_{2,j+1} \quad (10.52)$$

$$= (2 - 2r) u_{1,j} + ru_{2,j} + r\{\sin(\pi jk) + \sin(\pi(j+1)k)\},$$

$$-2ru_{n-1,j+1} + (2 + 2r) u_{n,j+1} = 2ru_{n-1,j} + (2 - 2r) u_{n,j}. \quad (10.53)$$

Schreibt man sich die Gleichungen (10.51) bis (10.53) für einen festen Index  $j$  auf, entsteht ein lineares Gleichungssystem für die  $n$  Unbekannten  $u_{1,j+1}, u_{2,j+1}, \dots, u_{n,j+1}$ , dessen Koeffizientenmatrix tridiagonal ist. Da in jedem Zeitschritt ein Gleichungssystem zu lösen ist, ist die dargestellte *Methode von Crank-Nicolson* implizit.

Der *lokale Diskretisierungsfehler* der Rechenvorschrift (10.51) ist definiert als

$$\begin{aligned} d_{i,j+1} := & -ru(x_{i-1}, t_{j+1}) + (2 + 2r)u(x_i, t_{j+1}) - ru(x_{i+1}, t_{j+1}) \\ & -ru(x_{i-1}, t_j) - (2 - 2r)u(x_i, t_j) - ru(x_{i+1}, t_j). \end{aligned}$$

Setzt man darin die Taylor-Entwicklungen bezüglich  $P(x_i, t_j)$  ein, so erhält man die folgende Darstellung für  $d_{i,j+1}$ .

$$\begin{aligned} d_{i,j+1} = & 2k\{u_t - u_{xx}\} + k^2\{u_{tt} - u_{xxt}\} \\ & + \frac{1}{3}k^3u_{ttt} - \frac{1}{6}h^2ku_{xxxx} - \frac{1}{2}k^3u_{xxtt} + \frac{1}{12}k^4u_{tttt} + \dots \end{aligned}$$

Die erste geschweifte Klammer ist gleich null, denn  $u(x, t)$  ist nach Voraussetzung Lösung von  $u_t = u_{xx}$ . Auch die zweite geschweifte Klammer ist gleich null, denn der Ausdruck ist gleich der partiellen Ableitung nach  $t$  von  $u_t - u_{xx} = 0$ . Folglich gilt wegen  $u_{ttt} = u_{xxtt}$

$$d_{i,j+1} = -\frac{1}{6}k^3u_{xxtt} - \frac{1}{6}h^2ku_{xxxx} + \dots = O(k^3) + O(h^2k). \quad (10.54)$$

Die Beziehung zum *globalen Diskretisierungsfehler*  $g_{i,j+1}$  der impliziten Methode von Crank-Nicolson wird hergestellt durch die Feststellung, dass die Formeln (10.51) bis (10.53) der Integration des Differenzialgleichungssystems (10.46) nach der *Trapezmethode* (8.16) mit dem Zeitschritt  $k$  entsprechen. Somit gilt  $g_{i,j+1} = O(k^2) + O(h^2)$ , und die implizite Methode von Crank-Nicolson ist von zweiter Ordnung bezüglich  $h$  und  $k$ .

Als nächstes zeigen wir die *absolute Stabilität* der impliziten Methode für die Aufgabe (10.42). Mit Vektoren  $\mathbf{u}_j$  gemäß (10.48) und der Matrix  $\mathbf{J}$  (10.49) lauten die Rechenvorschriften (10.51) bis (10.53)

$$(2\mathbf{I} + r\mathbf{J})\mathbf{u}_{j+1} = (2\mathbf{I} - r\mathbf{J})\mathbf{u}_j + \mathbf{b}_j, \quad (10.55)$$

wo  $\mathbf{b}_j = r\{\sin(\pi jk) + \sin(\pi(j+1)k)\}\mathbf{e}_1$  ist. Die Matrix  $2\mathbf{I} + r\mathbf{J}$  ist wegen  $r > 0$  diagonal dominant und folglich regulär. Mit ihrer Inversen lautet (10.55) formal

$$\mathbf{u}_{j+1} = (2\mathbf{I} + r\mathbf{J})^{-1}(2\mathbf{I} - r\mathbf{J})\mathbf{u}_j + (2\mathbf{I} + r\mathbf{J})^{-1}\mathbf{b}_j. \quad (10.56)$$

Die Methode ist absolut stabil, falls die Eigenwerte  $\lambda_\nu$  der Matrix

$$\mathbf{B} := (2\mathbf{I} + r\mathbf{J})^{-1}(2\mathbf{I} - r\mathbf{J})$$

betragsmäßig kleiner als Eins sind. Wie oben bereits festgestellt worden ist, gilt  $0 < \mu_\nu < 4$  für die Eigenwerte  $\mu_\nu$  von  $\mathbf{J}$ , und somit sind die Eigenwerte von  $\mathbf{B}$

$$-1 < \lambda_\nu = \frac{2 - r\mu_\nu}{2 + r\mu_\nu} < 1 \quad \text{für alle } \nu \text{ und alle } r > 0.$$

Die implizite Methode von Crank-Nicolson ist absolut stabil, denn der Wert  $r = k/h^2$  unterliegt keiner Einschränkung bezüglich Stabilität. Natürlich darf  $k$  nicht beliebig groß gewählt werden, da sonst der globale Diskretisierungsfehler zu groß wird. Wegen (10.54) ist oft die Wahl  $k = h$ , also  $r = 1/h$  durchaus sinnvoll. Die Integration in Zeitrichtung erfolgt dann in bedeutend größeren Zeitschritten als dies bei der expliziten Methode möglich wäre.

Die in jedem Zeitschritt durchzuführende Berechnung des Vektors  $\mathbf{u}_{j+1}$  aus dem Gleichungssystem (10.55) ist nicht sehr aufwändig, weil dessen Koeffizientenmatrix  $(2\mathbf{I} + r\mathbf{J})$  erstens tridiagonal und diagonal dominant und zweitens konstant für alle  $j$  ist. Deshalb ist die  $LR$ -Zerlegung nur einmal, und zwar mit Diagonalstrategie durchzuführen, wozu etwa  $2n$  wesentliche Operationen nötig sind (vgl. Abschnitt 2.3.3). Für jeden Integrationsschritt sind nur die Vorwärts- und Rücksubstitution mit etwa  $3n$  multiplikativen Operationen für die jeweilige rechte Seite auszuführen, deren Berechnung weitere  $2n$  Multiplikationen erfordert, falls man ihre  $i$ -te Komponente in der Darstellung  $\varrho u_{i,j} + r(u_{i-1,j} + u_{i+1,j})$  mit  $\varrho = 2 - 2r$  ausrechnet. Ist  $r = 1$ , dann vereinfacht sich diese Formel wegen  $\varrho = 0$ . Der Rechenaufwand für einen Schritt mit der impliziten Methode von Crank-Nicolson beträgt somit

$$Z_{CN} \cong 5n$$

wesentliche Operationen. Nach (10.43) erfordert ein Schritt mit der expliziten Methode von Richardson etwa  $2n$  Multiplikationen. Da aber der Zeitschritt  $k$  der impliziten Methode keiner Stabilitätsbedingung unterliegt, ist sie bedeutend effizienter, da  $k$  viel größer gewählt werden kann. Der Mehraufwand pro Schritt wird durch die geringere Zahl von Schritten bei weitem kompensiert.

**Beispiel 10.8.** Die Anfangsrandwertaufgabe (10.42) behandeln wir mit der impliziten Methode von Crank-Nicolson für verschiedene Kombinationen von  $h$  und  $k$ , um die oben behandelten Eigenschaften zu illustrieren. In Tab. 10.4 und 10.5 sind die Ergebnisse der Rechnung auszugsweise für  $h = 0.1$  und  $k = 0.01(r = 1)$ , bzw.  $k = 0.1(r = 10)$  zusammengestellt. Im ersten Fall erhalten wir eine Näherungslösung, welche mit den Ergebnissen von Tab. 10.1 oder 10.2 gut übereinstimmt, und dies trotz einer größeren Schrittweite  $k$ . Das ist eine Folge der höheren Fehlerordnung bezüglich  $k$ . Im zweiten Fall mit  $r = 10$  ist die implizite Methode zwar stabil, doch sind die Diskretisierungsfehler für den Zeitschritt  $k = 0.1$  erwartungsgemäß recht groß. Sie treten hauptsächlich in den Näherungswerten für die ersten diskreten Zeitwerte deutlich in Erscheinung. Der Zeitschritt  $k$  muss der zeitlichen Änderung der Randbedingung für  $x = 0$  angemessen sein und zudem auch der Größe von  $h$  angepasst sein, damit die beiden Hauptteile des globalen Fehlers vergleichbar groß sind.

Zu Vergleichszwecken ist die Aufgabe mit  $h = 0.05, k = 0.01$ , also  $r = 4.0$  behandelt worden. Das Ergebnis ist auszugsweise in Tab. 10.6 für die gleichen diskreten Stellen  $x_i$  wie in den vorhergehenden Tabellen angegeben. Die Näherungen stellen die exakte Lösung mit einer maximalen Abweichung von drei Einheiten in der vierten Dezimalstelle dar. Bei dieser feinen Ortsdiskretisation zeigt sich die Überlegenheit der impliziten gegenüber der expliziten Methode bereits deutlich. Denn bei dieser müsste  $k \leq 0.00125$  gewählt werden und somit wären achtmal mehr Schritte notwendig. Da der Rechenaufwand der impliziten Methode nur 2.5mal größer ist, ist sie mehr als dreimal effizienter.

Tab. 10.4 Wärmeleitung,  $h = 0.1, k = 0.01, r = 1$ ; implizite Methode.

$t$	$j$	$u_{0,j}$	$u_{1,j}$	$u_{2,j}$	$u_{3,j}$	$u_{4,j}$	$u_{6,j}$	$u_{8,j}$	$u_{10,j}$
0	0	0	0	0	0	0	0	0	0
0.1	10	0.3090	0.2141	0.1442	0.0942	0.0597	0.0219	0.0075	0.0039
0.2	20	0.5878	0.4582	0.3518	0.2662	0.1986	0.1076	0.0609	0.0467
0.3	30	0.8090	0.6691	0.5478	0.4445	0.3583	0.2328	0.1622	0.1395
0.4	40	0.9511	0.8222	0.7051	0.6011	0.5110	0.3733	0.2918	0.2649
0.5	50	1.0000	0.9005	0.8048	0.7156	0.6353	0.5069	0.4276	0.4008
0.6	60	0.9511	0.8952	0.8345	0.7730	0.7141	0.6140	0.5487	0.5262
0.7	70	0.8090	0.8057	0.7894	0.7649	0.7363	0.6791	0.6374	0.6224
0.8	80	0.5878	0.6401	0.6725	0.6899	0.6966	0.6918	0.6803	0.6751
0.9	90	0.3090	0.4140	0.4940	0.5535	0.5968	0.6479	0.6697	0.6754
1.0	100	0	0.1490	0.2704	0.3678	0.4448	0.5492	0.6036	0.6203

Tab. 10.5 Wärmeleitung,  $h = 0.1, k = 0.1, r = 10$ ; implizite Methode.

$t$	$j$	$u_{0,j}$	$u_{1,j}$	$u_{2,j}$	$u_{3,j}$	$u_{4,j}$	$u_{6,j}$	$u_{8,j}$	$u_{10,j}$
0	0	0	0	0	0	0	0	0	0
0.1	1	0.3090	0.1983	0.1274	0.0818	0.0527	0.0222	0.0104	0.0073
0.2	2	0.5878	0.4641	0.3540	0.2637	0.1934	0.1025	0.0587	0.0459
0.3	3	0.8090	0.6632	0.5436	0.4422	0.3565	0.2295	0.1575	0.1344
0.4	4	0.9511	0.8246	0.7040	0.5975	0.5064	0.3684	0.2866	0.2594
0.5	5	1.0000	0.8969	0.8022	0.7132	0.6320	0.5017	0.4215	0.3946
$\vdots$	$\vdots$								
1.0	10	0	0.1498	0.2701	0.3672	0.4440	0.5477	0.6014	0.6179

Tab. 10.6 Wärmeleitung,  $h = 0.05, k = 0.01, r = 4$ ; implizite Methode.

$t$	$j$	$u_{0,j}$	$u_{2,j}$	$u_{4,j}$	$u_{6,j}$	$u_{8,j}$	$u_{12,j}$	$u_{16,j}$	$u_{20,j}$
0	0	0	0	0	0	0	0	0	0
0.1	10	0.3090	0.2140	0.1439	0.0938	0.0592	0.0214	0.0071	0.0037
0.2	20	0.5878	0.4581	0.3516	0.2659	0.1982	0.1070	0.0602	0.0460
0.3	30	0.8090	0.6691	0.5477	0.4442	0.3580	0.2323	0.1615	0.1387
0.4	40	0.9511	0.8222	0.7050	0.6010	0.5108	0.3729	0.2912	0.2642
0.5	50	1.0000	0.9006	0.8048	0.7156	0.6352	0.5067	0.4273	0.4005
0.6	60	0.9511	0.8953	0.8346	0.7732	0.7143	0.6140	0.5487	0.5261
0.7	70	0.8090	0.8058	0.7897	0.7652	0.7366	0.6794	0.6377	0.6226
0.8	80	0.5878	0.6403	0.6728	0.6903	0.6971	0.6924	0.6809	0.6757
0.9	90	0.3090	0.4142	0.4943	0.5540	0.5974	0.6487	0.6705	0.6763
1.0	100	0	0.1492	0.2707	0.3684	0.4454	0.5501	0.6046	0.6214

Schließlich ist die sich periodisch wiederholende Temperaturverteilung im Stab bestimmt worden.

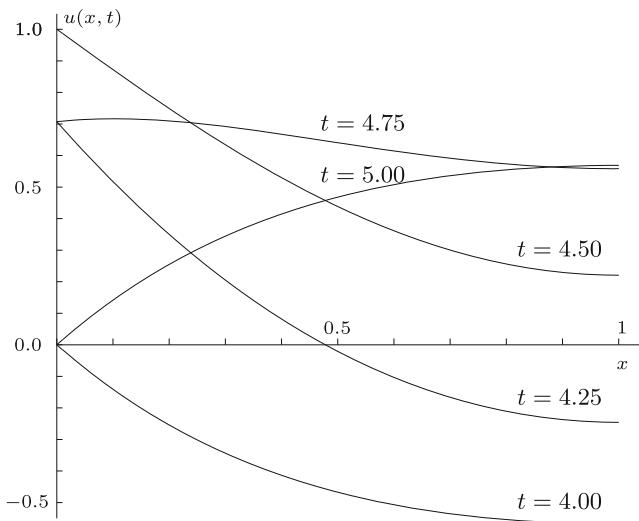


Abb. 10.19 Temperaturverteilungen im stationären Zustand.

Tab. 10.7 Zum stationären Temperaturablauf,  $h = 0.05, k = 0.01, r = 4$ .

$t$	$j$	$u_{0,j}$	$u_{2,j}$	$u_{4,j}$	$u_{6,j}$	$u_{8,j}$	$u_{12,j}$	$u_{16,j}$	$u_{20,j}$
2.00	200	0	-0.1403	-0.2532	-0.3425	-0.4120	-0.5040	-0.5505	-0.5644
2.25	225	0.7071	0.5176	0.3505	0.2057	0.0825	-0.1018	-0.2089	-0.2440
2.50	250	1.0000	0.8726	0.7495	0.6344	0.5301	0.3620	0.2572	0.2217
2.75	275	0.7071	0.7167	0.7099	0.6921	0.6679	0.6148	0.5739	0.5588
3.00	300	0	0.1410	0.2546	0.3447	0.4148	0.5080	0.5551	0.5693
3.25	325	-0.7071	-0.5171	-0.3497	-0.2045	-0.0810	0.1039	0.2114	0.2467
3.50	350	-1.0000	-0.8724	-0.7491	-0.6338	-0.5293	-0.3609	-0.2559	-0.2203
3.75	375	-0.7071	-0.7165	-0.7097	-0.6918	-0.6674	-0.6142	-0.5732	-0.5580
4.00	400	0	-0.1410	-0.2545	-0.3445	-0.4146	-0.5076	-0.5547	-0.5689
4.25	425	0.7071	0.5172	0.3497	0.2046	0.0811	-0.1037	-0.2112	-0.2464
4.50	450	1.0000	0.8724	0.7491	0.6338	0.5293	0.3610	0.2560	0.2204
4.75	475	0.7071	0.7165	0.7097	0.6918	0.6675	0.6142	0.5732	0.5581
5.00	500	0	0.1410	0.2545	0.3445	0.4146	0.5076	0.5547	0.5689

Dieser stationäre Zustand ist nach zwei Perioden ( $t = 4$ ) bereits erreicht. In Tab. 10.7 sind die Temperaturnäherungen für  $t \geq 2$  angegeben, und in Abb. 10.19 sind die Temperaturverteilungen für einige äquidistante Zeitpunkte einer halben Periode dargestellt.  $\triangle$

### 10.2.3 Diffusionsgleichung mit variablen Koeffizienten

Diffusionsprozesse mit ortsabhängigen Diffusionskennzahlen und Quellendichten werden beschrieben durch parabolische Differenzialgleichungen für die Konzentrationsfunktion  $u(x, t)$

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x} \left( a(x) \frac{\partial u}{\partial x} \right) + p(x)u + q(x), \quad 0 < x < 1, \quad t > 0, \quad (10.57)$$

wo  $a(x) > 0$ ,  $p(x)$  und  $q(x)$  gegebene Funktionen von  $x$  sind. Zu (10.57) gehören selbstverständlich Anfangs- und Randbedingungen. Zur Diskretisierung der Aufgabe verwenden wir ein Netz nach Abb. 10.17. Den Differenzialausdruck auf der rechten Seite von (10.57) approximieren wir im Gitterpunkt  $P(x_i, t_j)$  durch zweimalige Anwendung des ersten zentralen Differenzenquotienten zur Schrittweite  $h/2$ , wobei die Funktionswerte  $a(x_i + h/2) =: a_{i+\frac{1}{2}}$  und  $a(x_i - h/2) =: a_{i-\frac{1}{2}}$  auftreten.

$$\frac{\partial}{\partial x} \left( a(x) \frac{\partial u}{\partial x} \right)_P \approx \frac{1}{h^2} \left\{ a_{i+\frac{1}{2}}(u_{i+1,j} - u_{i,j}) - a_{i-\frac{1}{2}}(u_{i,j} - u_{i-1,j}) \right\}$$

Weiter bezeichnen wir zur Abkürzung mit  $p_i := p(x_i)$ ,  $q_i := q(x_i)$  die bekannten Werte der Funktionen. Die Differenzenapproximation nach dem impliziten Schema von Crank-Nicolson liefert für (10.57)

$$\begin{aligned} \frac{u_{i,j+1} - u_{i,j}}{k} &= \frac{1}{2} \left[ \frac{1}{h^2} \left\{ a_{i+\frac{1}{2}}(u_{i+1,j+1} - u_{i,j+1}) - a_{i-\frac{1}{2}}(u_{i,j+1} - u_{i-1,j+1}) \right\} \right. \\ &\quad \left. + p_i u_{i,j+1} + q_i \right] + p_i u_{i,j} + q_i \\ &\quad + \frac{1}{h^2} \left\{ a_{i+\frac{1}{2}}(u_{i+1,j} - u_{i,j}) - a_{i-\frac{1}{2}}(u_{i,j} - u_{i-1,j}) \right\}. \end{aligned}$$

Nach Multiplikation mit  $2k$  fassen wir zusammen und erhalten für einen inneren Punkt mit  $r = k/h^2$  die Gleichung

$$\begin{aligned} &-ra_{i-\frac{1}{2}}u_{i-1,j+1} + \left\{ 2 + r \left( a_{i-\frac{1}{2}} + a_{i+\frac{1}{2}} - h^2 p_i \right) \right\} u_{i,j+1} - ra_{i+\frac{1}{2}}u_{i+1,j+1} \\ &= ra_{i-\frac{1}{2}}u_{i-1,j} + \left\{ 2 - r \left( a_{i-\frac{1}{2}} + a_{i+\frac{1}{2}} - h^2 p_i \right) \right\} u_{i,j} + ra_{i+\frac{1}{2}}u_{i+1,j} + 2kq_i, \\ &\quad i = 1, 2, \dots, n-1; \quad j = 0, 1, 2, \dots \end{aligned} \quad (10.58)$$

Wenn man noch die Randbedingungen berücksichtigt, resultiert aus (10.58) ein tridiagonales Gleichungssystem für die Unbekannten  $u_{i,j+1}$ ,  $j$  fest. Die Matrix des Systems ist diagonal dominant, falls  $2 - kp(x_i) > 0$  für alle Punkte  $x_i$ .

**Beispiel 10.9.** Zu lösen sei

$$\begin{aligned} \frac{\partial u}{\partial t} &= \frac{\partial}{\partial x} \left( (1 + 2x^2) \frac{\partial u}{\partial x} \right) + 4x(1-x)u + 5\sin(\pi x), \quad 0 < x < 1; \\ u(x, 0) &= 0, \quad 0 < x < 1; \end{aligned} \quad (10.59)$$

$$u(0, t) = 0, u_x(1, t) + 0.4u(1, t) = 0, \quad t > 0.$$

Tab. 10.8 Diffusionsproblem,  $h = 0.1$ ,  $k = 0.01$ ,  $r = 1$ .

$t$	$j$	$u_{1,j}$	$u_{2,j}$	$u_{3,j}$	$u_{4,j}$	$u_{5,j}$	$u_{6,j}$	$u_{8,j}$	$u_{10,j}$
0	0	0	0	0	0	0	0	0	0
0.1	10	0.1044	0.1963	0.2660	0.3094	0.3276	0.3255	0.2888	0.2533
0.2	20	0.1591	0.3010	0.4124	0.4872	0.5265	0.5365	0.5037	0.4560
0.3	30	0.1948	0.3695	0.5085	0.6044	0.6581	0.6765	0.6470	0.5915
0.4	40	0.2185	0.4150	0.5722	0.6821	0.7454	0.7695	0.7421	0.6814
0.5	50	0.2342	0.4451	0.6145	0.7336	0.8033	0.8311	0.8052	0.7410
$\vdots$									
1.0	100	0.2612	0.4969	0.6871	0.8222	0.9029	0.9371	0.9136	0.8435
$\vdots$									
1.5	150	0.2647	0.5035	0.6964	0.8336	0.9157	0.9507	0.9276	0.8567
$\vdots$									
2.0	200	0.2651	0.5044	0.6976	0.8351	0.9173	0.9525	0.9294	0.8584
$\vdots$									
2.5	250	0.2652	0.5045	0.6978	0.8353	0.9175	0.9527	0.9296	0.8586

Die Dirichlet-Randbedingung ist in (10.58) für  $i = 1$  mit  $u_{0,j} = u_{0,j+1} = 0$  einfach zu berücksichtigen. Die Cauchy-Randbedingung am rechten Rand wird approximiert mit Hilfe des zentralen Differenzenquotienten unter der Annahme, dass die Funktion  $u(x, t)$  auch außerhalb des Intervalls definiert ist, durch

$$\frac{u_{n+1,j} - u_{n-1,j}}{2h} + 0.4 u_{n,j} = 0 \quad \text{oder} \quad u_{n+1,j} = u_{n-1,j} - 0.8 h u_{n,j}.$$

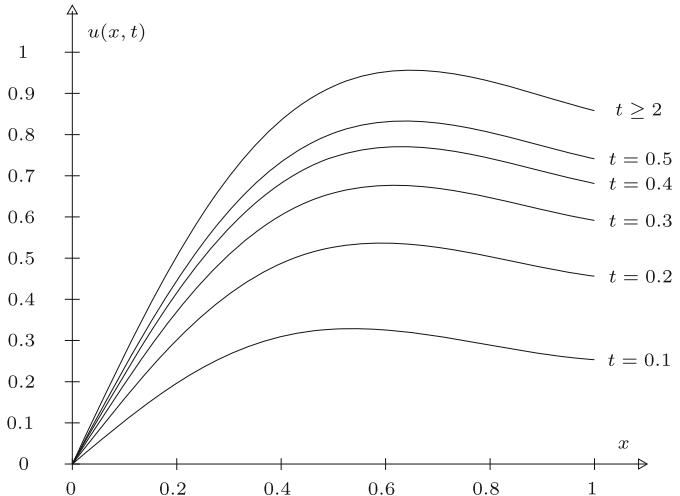
Nach Elimination von  $u_{n+1,j}$  und  $u_{n+1,j+1}$  in (10.58) lautet die Differenzengleichung unter der Voraussetzung, dass die Funktion  $a(x)$  auch außerhalb des  $x$ -Intervalls definiert ist, für die Gitterpunkte des rechten Randes

$$\begin{aligned} & -r \left( a_{n-\frac{1}{2}} + a_{n+\frac{1}{2}} \right) u_{n-1,j+1} + \left\{ 2 + r \left( a_{n-\frac{1}{2}} + (1 + 0.8h)a_{n+\frac{1}{2}} - h^2 p_n \right) \right\} u_{n,j+1} \\ & = r \left( a_{n-\frac{1}{2}} + a_{n+\frac{1}{2}} \right) u_{n-1,j} + \left\{ 2 - r \left( a_{n-\frac{1}{2}} + (1 + 0.8h)a_{n+\frac{1}{2}} - h^2 p_n \right) \right\} u_{n,j} + 2kq_n. \end{aligned}$$

Die diskrete Form der Anfangswertaufgabe (10.59) ist für  $n = 10$ ,  $h = 0.1$ ,  $k = 0.01$  und  $r = 1$  numerisch gelöst worden. Wenn wir die Matrix  $\mathbf{A}$  des tridiagonalen Gleichungssystems analog zu (10.55) als  $\mathbf{A} = 2\mathbf{I} + r\tilde{\mathbf{J}}$  schreiben, lautet die Matrix  $\tilde{\mathbf{J}} \in \mathbb{R}^{10,10}$  auszugsweise

$$\tilde{\mathbf{J}} = \begin{pmatrix} 2.0464 & -1.045 & & & & & & & \\ -1.045 & 2.1636 & -1.125 & & & & & & \\ & -1.125 & 2.3616 & -1.245 & & & & & \\ & & \ddots & \ddots & \ddots & & & & \\ & & & -2.125 & 4.5636 & -2.445 & & & \\ & & & & -2.445 & 5.2464 & -2.805 & & \\ & & & & & -6.01 & 6.2664 & & \end{pmatrix}.$$

Auf Grund einer analogen Betrachtung, wie sie zur Abschätzung der Eigenwerte der Matrix  $\mathbf{J}$

Abb. 10.20 Konzentrationsverteilung in Abhängigkeit der Zeit  $t$ .

(10.49) angewandt worden ist, folgt für die Eigenwerte  $\mu_\nu$  von  $\tilde{J}$

$$0 < \mu_\nu < 12.2764.$$

Für die zugehörige explizite Methode von Richardson ergibt sich daraus die Bedingung der absoluten Stabilität zu  $r \leq 1/6.1382 \doteq 0.163$ . Somit muss der Zeitschritt der Bedingung  $k \leq 0.00163$  genügen. Für die implizite, absolut stabile Methode darf der etwa sechsmal größere Zeitschritt  $k = 0.01$  verwendet werden. Die Ergebnisse sind auszugsweise in Tab. 10.8 wiedergegeben. Der stationäre Zustand wird innerhalb der angegebenen Stellenzahl bei etwa  $t = 2.0$  erreicht. Die Funktion  $u(x, t)$  ist in Abb. 10.20 für einige Zeitwerte dargestellt.  $\triangle$

### 10.2.4 Zweidimensionale Probleme

Die klassische parabolische Differenzialgleichung für eine Funktion  $u(x, y, t)$  der zwei Ortsvariablen  $x, y$  und der Zeitvariable  $t$  lautet

$$u_t = u_{xx} + u_{yy}. \quad (10.60)$$

Sie ist zu lösen in einem Gebiet  $G \subset \mathbb{R}^2$  der  $(x, y)$ -Ebene mit dem Rand  $\Gamma$  für Zeiten  $t > 0$ . Zur Differenzialgleichung (10.60) gehört sowohl eine *Anfangsbedingung*

$$u(x, y, 0) = f(x, y) \quad \text{in } G \quad (10.61)$$

als auch *Randbedingungen* auf dem Rand  $\Gamma$ , wie wir sie von den elliptischen Randwertaufgaben kennen. Da die Funktion  $u(x, y, t)$  zeitabhängig ist, können die in den Dirichletschen, Neumannschen und Cauchyschen Randbedingungen (10.6) bis (10.8) auftretenden Funktionen auch von der Zeit  $t$  abhängen. Die Argumentmenge  $(x, y, t)$ , für welche die Lösungsfunktion gesucht ist, besteht im  $\mathbb{R}^3$  aus dem Halbzyylinder über dem Gebiet  $G$ .

Zur Diskretisierung der Anfangsrandwertaufgabe verwenden wir ein regelmäßiges dreidimensionales Gitter, welches sich aufbaut aus einem regelmäßigen Netz im Gebiet  $G$  mit der

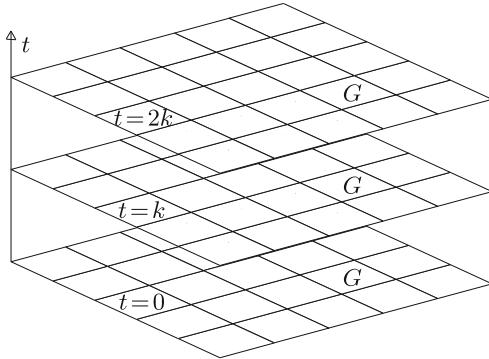


Abb. 10.21  
Ortsgitter in fortschreitenden Zeitschichten.

Gitterweite  $h$  (vgl. Abb. 10.2), das sich in gleichen Zeitschichtabständen  $k$  in Zeitrichtung fortsetzt, siehe Abb. 10.21. Gesucht werden Näherungen  $u_{\mu,\nu,j}$  der Funktionswerte in den Gitterpunkten  $P(x_\mu, y_\nu, t_j)$ .

Die Approximation von (10.60) erfolgt in zwei Teilen. Der Differenzialausdruck  $u_{xx} + u_{yy}$ , welcher nur partielle Ableitungen bezüglich der Ortsvariablen umfasst, wird für eine feste Zeitschicht  $t_j$  nach dem Vorgehen von Abschnitt 10.1 für jeden dieser Gitterpunkte durch einen entsprechenden Differenzenausdruck angenähert. Im einfachen Fall eines regelmäßigen inneren Gitterpunktes  $P(x_\mu, y_\nu, t_j)$  setzen wir

$$(u_{xx} + u_{yy})_P \approx \frac{1}{h^2} \{u_{\mu,\nu+1,j} + u_{\mu-1,\nu,j} + u_{\mu,\nu-1,j} + u_{\mu+1,\nu,j} - 4u_{\mu,\nu,j}\}$$

während für einen randnahen oder einen auf dem Rand liegenden Gitterpunkt Approximationen gemäß Abschnitt 10.1.3 zu verwenden sind.

Die partielle Ableitung nach der Zeit  $t$  kann beispielsweise durch den Vorwärtsdifferenzenquotienten in  $P$

$$u_{tt} \approx \frac{1}{k} (u_{\mu,\nu,j+1} - u_{\mu,\nu,j})$$

approximiert werden. Dies führt zur *expliziten Methode von Richardson* mit der Rechenvor- schrift für einen regelmäßigen inneren Gitterpunkt

$$u_{\mu,\nu,j+1} = u_{\mu,\nu,j} + r \{u_{\mu,\nu+1,j} + u_{\mu-1,\nu,j} + u_{\mu,\nu-1,j} + u_{\mu+1,\nu,j} - 4u_{\mu,\nu,j}\}. \quad (10.62)$$

Darin haben wir wieder  $r = k/h^2$  gesetzt. Um für das Folgende die Indizes zu vereinfachen, setzen wir voraus, dass die Gitterpunkte mit unbekanntem Wert  $u$  in jeder Zeitschicht von 1 bis  $n$  durchnumbert seien, wie dies im Abschnitt 10.1.2 beschrieben ist. Dann fassen wir die Näherungswerte in der  $j$ -ten Zeitschicht mit  $t_j = jk$  zum Vektor

$$\mathbf{u}_j := (u_{1,j}, u_{2,j}, \dots, u_{n,j})^T \in \mathbb{R}^n \quad (10.63)$$

zusammen, wo sich der erste Index  $i$  von  $u_{i,j}$  auf die Nummer des Gitterpunktes bezieht. Dann lässt sich (10.62) zusammenfassen zu

$$\mathbf{u}_{j+1} = (\mathbf{I} - r\mathbf{A})\mathbf{u}_j + \mathbf{b}_j, \quad j = 0, 1, 2, \dots \quad (10.64)$$

Die Matrix  $\mathbf{A} \in \mathbb{R}^{n,n}$  ist die Koeffizientenmatrix des Gleichungssystems der Differenzengleichungen zur Lösung der Poisson-Gleichung im Gebiet  $G$ ; dabei enthält der Vektor  $\mathbf{b}_j$  von

den Randbedingungen herrührende Konstanten. Die Bedingung für die absolute Stabilität der expliziten Methode besteht darin, dass die Eigenwerte der Matrix  $(\mathbf{I} - r\mathbf{A})$  dem Betrag nach kleiner als Eins sind. Die daraus für  $r$  zu beachtende Bedingung kann allgemein nur für symmetrische und positiv definite Matrizen  $\mathbf{A}$  angegeben werden. In diesem Fall gilt für die Eigenwerte  $\lambda_\nu$  von  $(\mathbf{I} - r\mathbf{A})$ , falls  $\mu_\nu$  die Eigenwerte von  $\mathbf{A}$  sind,

$$\lambda_\nu = 1 - r\mu_\nu, \quad \nu = 1, 2, \dots, n; \quad \mu_\nu > 0.$$

Damit ergibt sich aus  $1 - r\mu_\nu > -1$  für alle  $\nu$  die Bedingung

$$r < 2 / \max_\nu(\mu_\nu). \quad (10.65)$$

Für eine Matrix  $\mathbf{A}$ , welche durch die Fünf-Punkte-Formel (10.12) definiert ist wie z.B. (10.19), ist auf Grund der Zeilenmaximumnorm  $\max_\nu(\mu_\nu) \leq 8$ , so dass die Bedingung

$$r < \frac{1}{4}, \quad \text{d.h. } k < \frac{1}{4}h^2 \quad (10.66)$$

für die absolute Stabilität der expliziten Methode (10.62) zu beachten ist. Der größte Eigenwert von  $\mathbf{A}$  ist stets kleiner als 8, weshalb in (10.66) auch Gleichheit zulässig ist. Durch  $k \leq \frac{1}{4}h^2$  wird aber die Größe des Zeitschrittes  $k$  sehr stark eingeschränkt. Deshalb ist wiederum das *implizite Verfahren von Crank-Nicolson* anzuwenden. In (10.62) wird die geschweifte Klammer durch das arithmetische Mittel der Ausdrücke der  $j$ -ten und  $(j+1)$ -ten Zeitschicht ersetzt. An die Stelle von (10.64) tritt – mit anderem  $\mathbf{b}_j$  – die Rechenvorschrift

$$(2\mathbf{I} + r\mathbf{A})\mathbf{u}_{j+1} = (2\mathbf{I} - r\mathbf{A})\mathbf{u}_j + \mathbf{b}_j, \quad j = 0, 1, 2, \dots \quad (10.67)$$

Sie ist absolut stabil für symmetrische und positiv definite Matrizen  $\mathbf{A}$  oder für unsymmetrische Matrizen  $\mathbf{A}$ , deren Eigenwerte  $\mu_\nu$  positiven Realteil haben, denn dann sind die Eigenwerte  $\lambda_\nu$  von  $(2\mathbf{I} + r\mathbf{A})^{-1}(2\mathbf{I} - r\mathbf{A})$  für alle  $r > 0$  betragsmäßig kleiner als Eins.

Die Berechnung der Näherungswerte  $\mathbf{u}_{j+1}$  in den Gitterpunkten der  $(j+1)$ -ten Zeitschicht erfordert nach (10.67) die Lösung eines linearen Gleichungssystems mit der in der Regel diagonal dominanten, und für alle Zeitschritte konstanten Matrix  $(2\mathbf{I} + r\mathbf{A})$ . Nach einer einmal erfolgten *LR*-Zerlegung ist für die bekannte rechte Seite von (10.67) die Vorwärts- und die Rücksubstitution auszuführen. Bei kleiner Gitterweite  $h$  sind die Ordnung der Matrix  $(2\mathbf{I} + r\mathbf{A})$  und ihre Bandbreite recht groß, so dass sowohl ein beträchtlicher Speicherplatz als auch ein großer Rechenaufwand pro Zeitschritt notwendig sind.

Um den Aufwand hinsichtlich beider Gesichtspunkte wesentlich zu verringern, haben *Pearceman* und *Rachford* [Pea 55] eine Diskretisierung vorgeschlagen, die zum Ziel hat, in jedem Zeitschritt eine Folge von tridiagonalen Gleichungssystemen lösen zu müssen. Die Idee besteht darin, pro Schritt zwei verschiedene Differenzenapproximationen miteinander zu kombinieren. Dazu wird der Zeitschritt  $k$  halbiert, und es werden Hilfswerte  $u_{\mu,\nu,j+1/2} := u_{\mu\nu}^*$  zum Zeitpunkt  $t_j + \frac{1}{2}k = t_{j+1/2}$  als Lösung der Differenzengleichungen

$$\begin{aligned} \frac{2}{k}(u_{\mu,\nu}^* - u_{\mu,\nu,j}) = & \frac{1}{h^2}(u_{\mu+1,\nu}^* - 2u_{\mu\nu}^* + u_{\mu-1,\nu}^*) \\ & + \frac{1}{h^2}(u_{\mu,\nu+1,j} - 2u_{\mu,\nu,j} + u_{\mu,\nu-1,j}) \end{aligned} \quad (10.68)$$

definiert. Zur Approximation von  $u_{xx}$  wird der zweite Differenzenquotient mit Hilfswerten der Zeitschicht  $t_{j+1/2}$  verwendet, die zweite partielle Ableitung  $u_{yy}$  wird hingegen mit Hilfe von (bekannten) Näherungswerten der Zeitschicht  $t_j$  approximiert, und die Ableitung  $u_t$  durch den gewöhnlichen ersten Differenzenquotienten, aber natürlich mit der halben Schrittweite  $k/2$ . Fassen wir die Hilfswerte  $u_{\mu,\nu}^*$  für festes  $\nu$ , d.h. die Werte, die zu Gitterpunkten längs einer zur  $x$ -Achse parallelen Netzlinie gehören, zu Gruppen zusammen, so ergibt (10.68) für sie ein tridiagonales Gleichungssystem mit der typischen Gleichung

$$\begin{aligned} & -ru_{\mu-1,\nu}^* + (2 + 2r)u_{\mu,\nu}^* - ru_{\mu+1,\nu}^* \\ = & \quad ru_{\mu,\nu-1,j} + (2 - 2r)u_{\mu,\nu,j} + ru_{\mu,\nu+1,j}; \quad r = k/h^2. \end{aligned} \quad (10.69)$$

Zur Bestimmung der Gesamtheit aller Hilfswerte  $u_{\mu,\nu}^*$  ist somit für jede zur  $x$ -Achse parallele Linie des Netzes ein tridiagonales Gleichungssystem zu lösen. Mit den so berechneten Hilfswerten werden die Näherungen  $u_{\mu,\nu,j+1}$  der Zeitschicht  $t_{j+1}$  aus den Differenzengleichungen

$$\begin{aligned} \frac{2}{k}(u_{\mu,\nu,j+1} - u_{\mu,\nu}^*) &= \frac{1}{h^2}(u_{\mu+1,\nu}^* - 2u_{\mu,\nu}^* + u_{\mu-1,\nu}^*) \\ &\quad + \frac{1}{h^2}(u_{\mu,\nu+1,j+1} - 2u_{\mu,\nu,j+1} + u_{\mu,\nu-1,j+1}) \end{aligned} \quad (10.70)$$

bestimmt. Darin ist jetzt  $u_{xx}$  mit bekannten Hilfswerten und  $u_{yy}$  durch die gesuchten Näherungswerte der  $(j+1)$ -ten Zeitschicht approximiert. Nun ist wichtig, dass wir in (10.70) die unbekannten Werte  $u_{\mu,\nu,j+1}$  für festes  $\mu$ , d.h. für Gitterpunkte, die auf einer zur  $y$ -Achse parallelen Netzlinie liegen, zusammenfassen. Für jede dieser Gruppen stellt (10.70) wiederum ein tridiagonales Gleichungssystem mit der typischen Gleichung

$$\begin{aligned} & -ru_{\mu,\nu-1,j+1} + (2 + 2r)u_{\mu,\nu,j+1} - ru_{\mu,\nu+1,j+1} \\ = & \quad ru_{\mu-1,\nu}^* + (2 - 2r)u_{\mu,\nu}^* + ru_{\mu+1,\nu}^*, \quad r = k/h^2 \end{aligned} \quad (10.71)$$

dar. Damit ist wiederum eine Folge von tridiagonalen Gleichungssystemen für die Unbekannten  $u_{\mu,\nu,j+1}$  in den Gitterpunkten, die zu Netzlinien parallel zur  $y$ -Achse gehören, zu lösen. Wegen des Wechsels der Richtung, in welcher die Gitterpunkte zusammengefasst werden, heisst das Verfahren von Peaceman und Rachford auch *Methode der alternierenden Richtungen*. Ihre Genauigkeit ist bei entsprechenden Bedingungen  $O(h^2 + k^2)$ .

Die tridiagonalen Gleichungssysteme (10.69) und (10.71) sind von der Art, wie sie bei eindimensionalen Problemen auftreten. Die Matrizen sind diagonal dominant. Der Speicherbedarf ist minimal, und der Rechenaufwand zur Lösung von allen tridiagonalen Systemen in einem Zeitschritt ist nur proportional zur Zahl der Gitterpunkte pro Zeitschicht.

**Beispiel 10.10.** Eine besonders einfache und durchsichtige Situation ergibt sich für ein Rechteckgebiet  $G$  gemäß Abb. 10.22, falls die Anfangswertaufgabe (10.60) unter Dirichletschen Randbedingungen zu lösen ist. Die Gitterweite  $h$  sei so wählbar, dass  $h = a/(N+1) = b/(M+1)$  mit  $N, M \in \mathbb{N}^*$  gilt. Es ergeben sich somit  $n = N \cdot M$  innere Gitterpunkte mit unbekanntem Funktionswert. Die tridiagonalen Gleichungssysteme für die Hilfswerte  $u^*$ , die für jede Netzlinie parallel

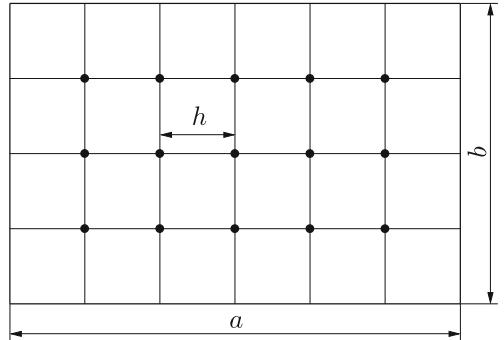


Abb. 10.22  
Rechteckiges Gebiet.

zur  $x$ -Achse zu lösen sind, haben die gleiche Matrix, welche im Fall  $N = 5$

$$\mathbf{H} := \begin{pmatrix} 2 + 2r & -r & & & \\ -r & 2 + 2r & -r & & \\ & -r & 2 + 2r & -r & \\ & & -r & 2 + 2r & -r \\ & & & -r & 2 + 2r \end{pmatrix}$$

lautet. Es genügt somit, bei gewähltem  $r$  für diese Matrix als Vorbereitung die  $LR$ -Zerlegung zu berechnen, um später für die Bestimmung der  $M$  Gruppen von Hilfswerten nur die Vorwärts- und Rücksubstitution auszuführen. Zur Berechnung der Näherungen  $u$  der  $(j+1)$ -ten Zeitschicht sind dann  $N$  tridiagonale Gleichungssysteme mit je  $M$  Unbekannten mit der festen Matrix ( $M = 3$ )

$$\mathbf{V} := \begin{pmatrix} 2 + 2r & -r & & \\ -r & 2 + 2r & -r & \\ & -r & 2 + 2r & \end{pmatrix}$$

zu lösen, für die ebenfalls die  $LR$ -Zerlegung bereitzustellen ist. Werden die bekannten rechten Seiten der Gleichungssysteme (10.69) und (10.71) mit einem Minimum an Multiplikationen berechnet, sind für einen Integrationsschritt total nur etwa  $10n$  wesentliche Operationen nötig.  $\triangle$

**Beispiel 10.11.** Für ein weniger einfaches Gebiet  $G$  und andere Randbedingungen besitzt die Methode der alternierenden Richtungen eine entsprechend aufwändigere Realisierung. Wir betrachten dazu die parabolische Differenzialgleichung  $u_t = u_{xx} + u_{yy}$  für das Gebiet  $G$  der Abb. 10.6, Seite 433, mit den Randbedingungen (10.15) bis (10.17) und der Anfangsbedingung  $u(x, y, 0) = 0$  in  $G$ . Wir verwenden die Nummerierung der Gitterpunkte von Abb. 10.6 und stellen die Matrizen der tridiagonalen Gleichungssysteme zusammen, die für die Hilfswerte zu Gitterpunkten in den vier horizontalen Linien zu lösen sind. Bei der Berücksichtigung der Randbedingungen auf den Randstücken  $FA$  und  $CD$  sind die Differenzenapproximationen nicht durch zwei zu dividieren. Zu den Matrizen sind die zugehörigen  $u^*$ -Werte angegeben.

$$\mathbf{H}_1 := \begin{pmatrix} 2 + 2r & -2r & & & \\ -r & 2 + 2r & -r & & \\ & -r & 2 + 2r & -r & \\ & & -r & 2 + 2r & -r \\ & & & -r & 2 + 2r \end{pmatrix},$$

$$(u_1^*, u_4^*, u_7^*, u_{10}^*, u_{14}^*)^T,$$

$$\begin{aligned}
\mathbf{H}_2 &:= \begin{pmatrix} 2+2r & -2r & & & & \\ -r & 2+2r & -r & & & \\ & -r & 2+2r & -r & & \\ & & -r & 2+2r & -r & \\ & & & -r & 2+2r & -r \\ & & & & -r & 2+2r \end{pmatrix}, \\
&\quad (u_2^*, u_5^*, u_8^*, u_{11}^*, u_{15}^*, u_{18}^*)^T, \\
\mathbf{H}_3 &:= \begin{pmatrix} 2+2r & -2r & & & & \\ -r & 2+2r & -r & & & \\ & -r & 2+2r & -r & & \\ & & -r & 2+2r & -r & \\ & & & -r & 2+2r & -r \\ & & & & -2r & 2+2r \end{pmatrix}, \\
&\quad (u_3^*, u_6^*, u_9^*, u_{12}^*, u_{16}^*, u_{19}^*)^T, \\
\mathbf{H}_4 &:= \begin{pmatrix} 2+2r & -r & \\ -2r & 2+2r & \end{pmatrix}, \\
&\quad (u_{13}^*, u_{17}^*)^T.
\end{aligned}$$

Für den zweiten Halbschritt entstehen für die sechs vertikalen Linien nun vier verschiedene tri-diagonale Matrizen, denn für die ersten drei Linien sind die dreireihigen Matrizen identisch. Ihre Aufstellung sei dem Leser überlassen.  $\triangle$

## 10.3 Methode der finiten Elemente

Zur Lösung von elliptischen Randwertaufgaben betrachten wir im Folgenden die Energiemethode, welche darin besteht, eine zugehörige Variationsaufgabe näherungsweise zu lösen. Wir werden die grundlegende Idee der Methode der finiten Elemente zur Diskretisierung der Aufgabe darlegen und das Vorgehen für einen ausgewählten Ansatz vollständig darstellen. Für eine ausführliche Behandlung der Methode sei auf [Bra 03, Cia 02, Hac 96, Mit 85, Gro 05, Sch 91b, Zie 05] verwiesen, Hinweise auf rechnerische Lösungen findet man im Abschnitt 10.4 und in den Beispielen 10.12 und 10.13.

### 10.3.1 Grundlagen

In der  $(x, y)$ -Ebene sei ein beschränktes Gebiet  $G$  gegeben, welches begrenzt wird vom stückweise stetig differenzierbaren Rand  $\Gamma$ , der auch aus mehreren geschlossenen Kurven

bestehen darf (vgl. Abb. 10.1). Wir betrachten den Integralausdruck

$$\begin{aligned} I(u) := & \iint_G \left\{ \frac{1}{2}(u_x^2 + u_y^2) + \frac{1}{2}\varrho(x, y)u^2 - f(x, y)u \right\} dx dy \\ & + \oint_{\Gamma} \left\{ \frac{1}{2}\alpha(s)u^2 - \beta(s)u \right\} ds, \end{aligned} \quad (10.72)$$

wo  $\varrho(x, y)$  und  $f(x, y)$  auf  $G$  definierte Funktionen bedeuten,  $s$  die Bogenlänge auf  $\Gamma$  darstellt, und  $\alpha(s)$  und  $\beta(s)$  gegebene Funktionen der Bogenlänge sind. Zusätzlich zu (10.72) seien auf einem Teil  $\Gamma_1$  des Randes  $\Gamma$ , der auch den ganzen Rand umfassen oder auch leer sein kann, für die Funktion  $u(x, y)$  Randwerte vorgegeben.

$$u = \varphi(s) \quad \text{auf } \Gamma_1, \quad \Gamma_1 \subset \Gamma. \quad (10.73)$$

Wir wollen nun zeigen, dass diejenige Funktion  $u(x, y)$ , welche den Integralausdruck  $I(u)$  unter der Nebenbedingung (10.73) stationär macht, eine bestimmte elliptische Randwertaufgabe löst unter der Voraussetzung, dass  $u(x, y)$  hinreichend oft stetig differenzierbar ist. So mit wird es möglich sein, eine Extremalaufgabe für  $I(u)$  (10.72) unter der Nebenbedingung (10.73) zu behandeln, um auf diese Weise die Lösung einer elliptischen Randwertaufgabe zu bestimmen. Der Integralausdruck  $I(u)$  hat in den meisten Anwendungen die Bedeutung einer Energie und nimmt auf Grund von Extremalprinzipien (Hamiltonsches, Rayleighsches oder Fermatsches Prinzip) [Fun 70] nicht nur einen stationären Wert, sondern ein Minimum an. Dies trifft insbesondere dann zu, falls  $\varrho(x, y) \geq 0$  in  $G$  und  $\alpha(s) \geq 0$  auf  $\Gamma$  sind. Wegen des erwähnten Zusammenhangs spricht man auch von der *Energiemethode*.

Damit die Funktion  $u(x, y)$  den Integralausdruck  $I(u)$  stationär macht, muss notwendigerweise seine erste Variation verschwinden. Nach den Regeln der Variationsrechnung [Akh 88, Cou 93, Fun 70, Kli 88] erhalten wir

$$\begin{aligned} \delta I = & \iint_G \{u_x \delta u_x + u_y \delta u_y + \varrho(x, y)u \delta u - f(x, y)\delta u\} dx dy \\ & + \oint_{\Gamma} \{\alpha(s)u \delta u - \beta(s)\delta u\} ds. \end{aligned} \quad (10.74)$$

Da  $u_x \delta u_x + u_y \delta u_y = \operatorname{grad} u \cdot \operatorname{grad} \delta u$  ist, können wir die Greensche Formel unter der Voraussetzung  $u(x, y) \in C^2(G \cup \Gamma)$ ,  $v(x, y) \in C^1(G \cup \Gamma)$

$$\iint_G \operatorname{grad} u \cdot \operatorname{grad} v dx dy = - \iint_G \{u_{xx} + u_{yy}\}v dx dy + \oint_{\Gamma} \frac{\partial u}{\partial \mathbf{n}} v ds$$

anwenden, wo  $\partial u / \partial \mathbf{n}$  die Ableitung von  $u$  in Richtung der äußeren Normalen  $\mathbf{n}$  auf dem Rand  $\Gamma$  bedeutet, und erhalten aus (10.74)

$$\begin{aligned} \delta I = & \iint_G \{-\Delta u + \varrho(x, y)u - f(x, y)\}\delta u dx dy + \\ & \oint_{\Gamma} \left( \frac{\partial u}{\partial \mathbf{n}} + \alpha(s)u - \beta(s) \right) \delta u ds. \end{aligned} \quad (10.75)$$

Die erste Variation  $\delta I$  muss für jede zulässige Änderung  $\delta u$  der Funktion  $u$  verschwinden. Mit Hilfe einer Konkurrenz einschränkung mit  $\delta u = 0$  auf  $\Gamma$  folgt aus (10.75) als erste Bedingung die Eulersche Differenzialgleichung

$$-\Delta u + \varrho(x, y)u = f(x, y) \quad \text{in } G. \quad (10.76)$$

Die Funktion  $u(x, y)$ , welche als zweimal stetig differenzierbar vorausgesetzt ist, und den Integralausdruck  $I(u)$  (10.72) stationär macht, erfüllt notwendigerweise die elliptische Differenzialgleichung (10.76) in  $G$ .

Falls der Teil  $\Gamma_1$  mit vorgegebenen Randwerten (10.73), wo natürlich  $\delta u = 0$  sein muss, nicht den ganzen Rand  $\Gamma$  umfasst, folgt für den Rest  $\Gamma_2$  des Randes die weitere notwendige Bedingung

$$\frac{\partial u}{\partial n} + \alpha(s)u = \beta(s) \quad \text{auf } \Gamma_2 = \Gamma \setminus \Gamma_1. \quad (10.77)$$

Die Randbedingung (10.77) ist eine *natürliche Randbedingung*, der die Lösungsfunktion  $u(x, y)$  der Variationsaufgabe (10.72), (10.73) notwendigerweise genügen muss.

Falls  $u(x, y) \in C^2(G \cup \Gamma)$  das Funktional (10.72) unter der Nebenbedingung (10.73) stationär macht, folgt also, dass  $u(x, y)$  eine Lösung der elliptischen Differenzialgleichung (10.76) unter den Randbedingungen (10.73) und (10.77) ist. Es stellt sich natürlich die Frage nach der Existenz einer Lösung des Variationsproblems, d.h. nach der Existenz einer Funktion  $u(x, y)$ , welche  $I(u)$  unter der Nebenbedingung (10.73) stationär macht. Diese Problematik ist verknüpft mit der Wahl des Raumes der zulässigen Funktionen zur Lösung des Variationsproblems. Beim Betrachten von  $I(u)$  erkennt man, dass dieses Funktional bereits unter viel schwächeren Voraussetzungen an  $u(x, y)$  definiert ist, als oben vorausgesetzt wurde. So genügt es beispielsweise, dass  $u(x, y)$  stückweise stetig differenzierbar ist. Durch eine Erweiterung des Raumes der zulässigen Funktionen erhält man einen bestimmten *Sobolev-Raum*. Für diesen kann mit relativ einfachen funktionalanalytischen Hilfsmitteln die Existenz einer Lösung des Variationsproblems bewiesen werden. Als Element des Sobolev-Raumes braucht die Lösungsfunktion aber nicht zweimal stetig differenzierbar und damit auch nicht eine Lösung der elliptischen Randwertaufgabe zu sein. Sie heißt daher *schwache Lösung*. Diese Existenzaussage ist unter sehr allgemeinen Voraussetzungen an die Problemdaten  $\varrho(x, y)$ ,  $f(x, y)$ ,  $\alpha(s)$ ,  $\beta(s)$  und  $G$  gültig. Falls wir zusätzlich voraussetzen, dass diese Daten hinreichend glatt sind, so lässt sich, allerdings unter großen mathematischen Schwierigkeiten, zeigen, dass die Lösung der Variationsaufgabe tatsächlich auch eine Lösung des elliptischen Randwertproblems ist, das also die schwache Lösung eine *reguläre Lösung* ist.

Vom praktischen Gesichtspunkt aus ist die zuletzt geschilderte Problematik von geringer Bedeutung. Da in vielen Fällen die Variationsaufgabe die natürliche Art der Beschreibung eines physikalischen Sachverhalts darstellt, ist es nämlich überhaupt nicht notwendig auf das Randwertproblem zurückzugehen. Die Extremalaufgabe wird im Folgenden approximativ gelöst, indem eine Näherung für  $u(x, y)$  in einem endlich dimensionalen Funktionenraum von bestimmten stückweise stetig differenzierbaren Funktionen ermittelt wird.

Mit der Betrachtung der Variationsaufgabe zeigt sich eine für die Anwendung der Energiemethode wesentliche Unterscheidung der Randbedingungen. Die natürliche Randbedingung (10.77), welche die Normalableitung betrifft, ist in der Formulierung als Extremalproblem implizit im Randintegral von  $I(u)$  enthalten und braucht daher nicht explizit berücksichtigt

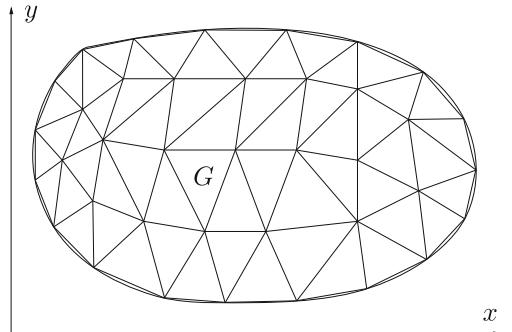


Abb. 10.23  
Triangulierung eines Gebietes  $G$ .

zu werden. Durch Spezialisierung der im Integralausdruck auftretenden Funktionen erhalten wir die elliptischen Randwertaufgaben (10.2) bis (10.4).

Spezielle natürliche Randbedingungen sind

$$\frac{\partial u}{\partial \mathbf{n}} = \beta(s) \quad \text{Neumann-Randbedingung } (\alpha = 0),$$

$$\frac{\partial u}{\partial \mathbf{n}} + \alpha(s)u = 0 \quad \text{Cauchy-Randbedingung } (\beta = 0).$$

### 10.3.2 Prinzip der Methode der finiten Elemente

Wir beschreiben zuerst das grundsätzliche Vorgehen der Methode und gehen anschließend auf die Detailausführung ein. Der Integralausdruck  $I(u)$  (10.72) bildet den Ausgangspunkt. Als erstes wollen wir diesen in geeigneter Weise approximieren, um dann die Bedingung des Stationärwerdens unter Berücksichtigung der Dirichletschen Randbedingung (10.73) zu formulieren.

In einem ersten Lösungsschritt erfolgt eine Diskretisierung des Gebietes  $G$  in einfache Teilegebiete, den so genannten *Elementen*. Wir wollen im Folgenden nur *Triangulierungen* betrachten, in denen das Gebiet  $G$  durch Dreieckelemente so überdeckt wird, dass aneinander grenzende Dreiecke eine ganze Seite oder nur einen Eckpunkt gemeinsam haben (vgl. Abb. 10.23). Das Grundgebiet  $G$  wird durch die Gesamtfläche der Dreiecke ersetzt. Ein krummlinig berandetes Gebiet kann sehr flexibel durch eine Triangulierung approximiert werden, wobei allenfalls am Rand eine lokale feinere Einteilung angewandt werden muss. Die Triangulierung sollte keine allzu stumpfwinkligen Dreiecke enthalten, um numerische Schwierigkeiten zu vermeiden.

Im zweiten Schritt wählt man für die gesuchte Funktion  $u(x, y)$  in jedem Dreieck einen bestimmten Ansatz  $\tilde{u}(x, y)$ . Dafür eignen sich lineare, quadratische und auch kubische Polynome in den beiden Variablen  $x$  und  $y$

$$\tilde{u}(x, y) = c_1 + c_2x + c_3y, \tag{10.78}$$

$$\tilde{u}(x, y) = c_1 + c_2x + c_3y + c_4x^2 + c_5xy + c_6y^2, \tag{10.79}$$

$$\tilde{u}(x, y) = c_1 + c_2x + c_3y + c_4x^2 + c_5xy + c_6y^2 + c_7x^3 + c_8x^2y + c_9xy^2 + c_{10}y^3. \tag{10.80}$$

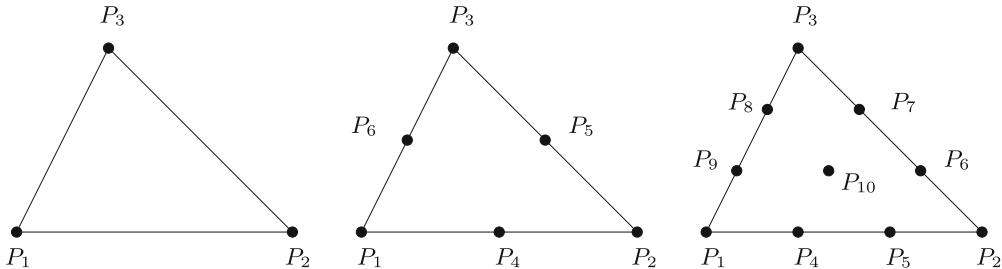


Abb. 10.24 Knotenpunkte im Dreieck bei linearem, quadratischen und kubischen Ansatz.

Diese für jedes Element gültigen Ansatzfunktionen müssen beim Übergang von einem Dreieck ins benachbarte zumindest stetig sein, damit eine für die Behandlung der Extremalaufgabe zulässige, d.h. stetige und einmal stückweise stetig differenzierbare Gesamtfunktion resultiert. Um diese Stetigkeitsbedingung zu erfüllen, sind entweder die Koeffizienten  $c_k$  in (10.78) oder (10.79) durch Funktionswerte in bestimmten *Knotenpunkten* des Dreiecks auszudrücken, oder aber man verwendet direkt einen geeigneten Ansatz für  $\tilde{u}(x, y)$  mit so genannten *Basisfunktionen*, die analog zu den Lagrange-Polynomen mit entsprechenden Interpolationseigenschaften bezüglich der Knotenpunkte definiert werden.

Im Fall des linearen Ansatzes (10.78) ist die Funktion  $\tilde{u}(x, y)$  im Dreieck eindeutig bestimmt durch die drei Funktionswerte in den Eckpunkten. Die Stetigkeit der linearen Ansätze beim Übergang in benachbarte Dreiecke folgt aus der Tatsache, dass sie auf den Dreiecksseiten lineare Funktionen der Bogenlänge sind, welche durch die Funktionswerte in den Endpunkten eindeutig bestimmt sind.

Die quadratische Ansatzfunktion (10.79) ist in einem Dreieck eindeutig festgelegt durch die sechs Funktionswerte in den drei Eckpunkten und den drei Mittelpunkten der Seiten. Die kubische Ansatzfunktion (10.80) wird durch ihre Werte in den Eckpunkten, in den Drittelpunkten und Zweidrittelpunkten auf den Dreiecksseiten und im Schwerpunkt eindeutig festgelegt. Die Ansatzfunktionen sind beim Übergang in benachbarte Elemente stetig, da sie auf der gemeinsamen Seite quadratische bzw. kubische Funktionen der Bogenlänge sind, die eindeutig bestimmt sind durch die Funktionswerte im den Interpolationspunkten.

Der dritte Schritt besteht darin, den Integralausdruck  $I$  in Abhängigkeit der Funktionswerte in den Knotenpunkten, den *Knotenvariablen*, für den gewählten Ansatz darzustellen. Dazu sind die Beiträge der einzelnen Dreieckelemente sowie der Randkanten bereitzustellen und zu addieren. Um das letztere systematisch vornehmen zu können, werden die Knotenpunkte durchnummerniert. Wir bezeichnen mit  $u_j$  den Funktionswert im Punkt mit der Nummer  $j$ . Einerseits sind die Integranden entweder quadratische oder lineare Funktionen in  $u$  und andererseits ist der Ansatz für  $\tilde{u}(x, y)$  linear in den Koeffizienten  $c_k$  und deshalb linear in den Knotenvariablen  $u_j$ . Deshalb ist der Integralausdruck  $I(\tilde{u}(x, y))$  eine quadratische Funktion der Knotenvariablen  $u_j$ . Sie beschreibt den Integralausdruck für einen linearen Funktionenraum, definiert durch die elementweise erklärten Funktionen, dessen Dimension gleich der Anzahl der Knotenpunkte der Gebietsdiskretisierung ist.

Im nächsten Schritt erfolgt die Berücksichtigung der Dirichletschen Randbedingung (10.73),

welche in bestimmten Randknotenpunkten die Werte der betreffenden Knotenvariablen vorschreibt. Diese bekannten Größen sind im Prinzip in der quadratischen Funktion für  $I$  einzusetzen. Für die verbleibenden unbekannten Knotenvariablen  $u_1, u_2, \dots, u_n$ , die wir im Vektor  $\mathbf{u} := (u_1, u_2, \dots, u_n)^T$  zusammenfassen, resultiert eine quadratische Funktion der Form

$$F := \frac{1}{2} \mathbf{u}^T \mathbf{A} \mathbf{u} - \mathbf{b}^T \mathbf{u} + d, \quad \mathbf{A} \in \mathbb{R}^{n,n}, \quad \mathbf{b} \in \mathbb{R}^n. \quad (10.81)$$

Darin ist  $\mathbf{A}$  eine symmetrische Matrix, die positiv definit ist, falls der Integralausdruck  $I$  (10.72) einer Energie entspricht oder  $\varrho(x, y) \geq 0$  und  $\alpha(s) \geq 0$  sind und hinreichende Zwangsbedingungen (10.73) gegeben sind. Der Vektor  $\mathbf{b}$  entsteht einerseits aus den linearen Anteilen von  $I$  und andererseits durch Beiträge beim Einsetzen von bekannten Werten von Knotenvariablen. Dasselbe gilt für die Konstante  $d$  in (10.81).

Die Bedingung des Stationärwerdens von  $F$  führt in bekannter Weise auf ein lineares Gleichungssystem

$$\mathbf{A} \mathbf{u} = \mathbf{b} \quad (10.82)$$

mit *symmetrischer* und *positiv definiter Matrix  $\mathbf{A}$* . Nach seiner Auflösung erhält man Werte  $u_j$ , die Näherungen für die exakten Funktionswerte von  $u(x, y)$  in den betreffenden Knotenpunkten darstellen. Durch den gewählten Funktionsansatz (10.78), (10.79) oder (10.80) ist der Verlauf der Näherungslösung  $\tilde{u}(x, y)$  in den einzelnen Dreieckelementen definiert, so dass beispielsweise Niveaulinien der Näherungslösung konstruiert werden können.

### 10.3.3 Elementweise Bearbeitung

Für die gewählte Triangulierung sind die Beiträge der Integrale der einzelnen Dreieckelemente und Randstücke für den betreffenden Ansatz in Abhängigkeit der Knotenvariablen zu bestimmen. Diese Beiträge sind quadratische oder lineare Funktionen in den  $u_j$ , und wir wollen die Matrizen der quadratischen Formen und die Koeffizienten der linearen Formen herleiten. Sie bilden die wesentliche Grundlage für den Aufbau des später zu lösenden linearen Gleichungssystems (10.82). Im Folgenden betrachten wir den quadratischen Ansatz (10.79) und setzen zur Vereinfachung voraus, dass die Funktionen  $\varrho(x, y)$ ,  $f(x, y)$ ,  $\alpha(s)$  und  $\beta(s)$  im Integralausdruck  $I$  (10.72) zumindest für jedes Element, bzw. jede Randkante konstant seien, so dass die betreffenden Werte  $\varrho$ ,  $f$ ,  $\alpha$  und  $\beta$  vor die entsprechenden Integrale gezogen werden können.

Wir betrachten ein Dreieck  $T_i$  in allgemeiner Lage mit den sechs Knotenpunkten  $P_1$  bis  $P_6$  für den quadratischen Ansatz. Wir setzen fest, dass die Eckpunkte  $P_1, P_2$  und  $P_3$  im Gegenuhrzeigersinn nach Abb. 10.25 angeordnet sein sollen. Die Koordinaten der Eckpunkte  $P_j$  seien  $(x_j, y_j)$ . Um den Wert des Integrals über ein solches Element

$$\iint_{T_i} (\tilde{u}_x^2 + \tilde{u}_y^2) dx dy \quad (10.83)$$

für einen der Ansätze am einfachsten zu bestimmen, wird  $T_i$  mittels einer linearen Transformation auf ein gleichschenklig rechtwinkliges *Normaldreieck*  $T$  abgebildet (Abb. 10.25). Die zugehörige Transformation lautet

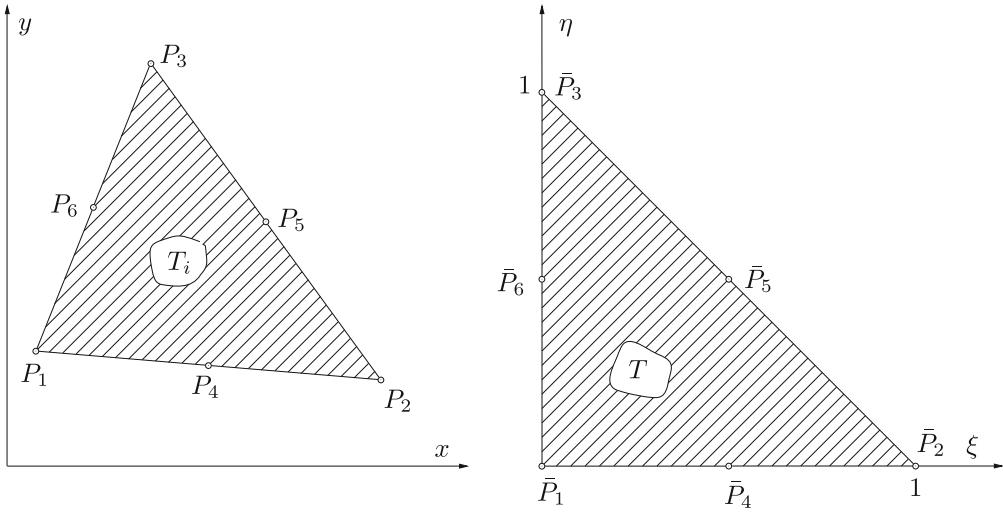


Abb. 10.25 Dreieckelement in beliebiger Lage und Normaldreieck mit Knotenpunkten für quadratischen Ansatz.

$$\begin{aligned} x &= x_1 + (x_2 - x_1)\xi + (x_3 - x_1)\eta, \\ y &= y_1 + (y_2 - y_1)\xi + (y_3 - y_1)\eta. \end{aligned} \quad (10.84)$$

Das Gebietsintegral (10.83) für  $T_i$  ist nach den Regeln der Analysis zu transformieren. Falls wir die transformierte Funktion gleich bezeichnen, gelten

$$\tilde{u}_x = \tilde{u}_\xi \xi_x + \tilde{u}_\eta \eta_x, \quad \tilde{u}_y = \tilde{u}_\xi \xi_y + \tilde{u}_\eta \eta_y. \quad (10.85)$$

Weiter folgen auf Grund der Transformation (10.84)

$$\xi_x = \frac{y_3 - y_1}{J}, \quad \eta_x = -\frac{y_2 - y_1}{J}, \quad \xi_y = -\frac{x_3 - x_1}{J}, \quad \eta_y = \frac{x_2 - x_1}{J}, \quad (10.86)$$

wo

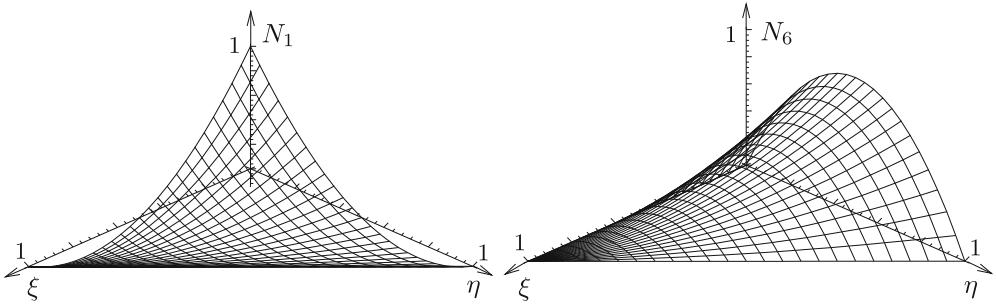
$$J := (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1) > 0 \quad (10.87)$$

die Jacobi-Determinante der Abbildung (10.84) bedeutet und wegen unserer Festsetzung über die Eckpunkte gleich der doppelten Fläche des Dreiecks  $T_i$  ist. Damit ergibt sich nach Substitution von (10.85) und (10.86) in (10.83)

$$\begin{aligned} \iint_{T_i} (\tilde{u}_x^2 + \tilde{u}_y^2) dx dy &= \iint_T [a\tilde{u}_\xi^2 + 2b\tilde{u}_\xi\tilde{u}_\eta + c\tilde{u}_\eta^2] d\xi d\eta \\ &=: aI_1 + bI_2 + cI_3 \end{aligned} \quad (10.88)$$

mit den konstanten, vom Element  $T_i$  abhängigen Koeffizienten

$$\begin{aligned} a &= [(x_3 - x_1)^2 + (y_3 - y_1)^2]/J, \\ b &= -[(x_3 - x_1)(x_2 - x_1) + (y_3 - y_1)(y_2 - y_1)]/J, \\ c &= [(x_2 - x_1)^2 + (y_2 - y_1)^2]/J. \end{aligned} \quad (10.89)$$

Abb. 10.26 Formfunktionen  $N_1(\xi, \eta)$  und  $N_6(\xi, \eta)$ .

Der quadratische Ansatz (10.79) in den  $x, y$ -Variablen geht durch die lineare Transformation (10.84) in einen quadratischen Ansatz derselben Form in den  $\xi, \eta$ -Variablen über. Zu seiner Darstellung wollen wir *Basisfunktionen* oder sog. *Formfunktionen* verwenden, welche gestatten,  $\tilde{u}(\xi, \eta)$  in Abhängigkeit der Funktionswerte  $u_j$  in den Knotenpunkten  $\bar{P}_i$  des Normaldreieckelementes  $T$  anzugeben. Zu diesem Zweck definieren wir in Analogie zu den Lagrange-Polynomen (siehe Abschnitt 3.1.2) für das Normaldreieck  $T$  und die sechs Knotenpunkte  $\bar{P}_j(\xi_j, \eta_j)$  die sechs Basisfunktionen  $N_i(\xi, \eta)$  mit den Interpolationseigenschaften

$$N_i(\xi_j, \eta_j) = \begin{cases} 1 & \text{falls } i = j \\ 0 & \text{falls } i \neq j \end{cases} \quad i, j = 1, 2, \dots, 6. \quad (10.90)$$

Die Formfunktionen  $N_i(\xi, \eta)$  können leicht angegeben werden, wenn man beachtet, dass eine solche Funktion auf einer ganzen Dreiecksseite gleich null ist, falls sie in den drei auf dieser Seite liegenden Knotenpunkten verschwinden muss. Somit sind

$\boxed{\begin{array}{l} N_1(\xi, \eta) = (1 - \xi - \eta)(1 - 2\xi - 2\eta) \\ N_2(\xi, \eta) = \xi(2\xi - 1) \\ N_3(\xi, \eta) = \eta(2\eta - 1) \end{array}}$	$\boxed{\begin{array}{l} N_4(\xi, \eta) = 4\xi(1 - \xi - \eta) \\ N_5(\xi, \eta) = 4\eta(1 - \xi - \eta) \\ N_6(\xi, \eta) = 4\eta(1 - \xi - \eta) \end{array}}$
--	--

(10.91)

In Abb. 10.26 sind zwei Formfunktionen veranschaulicht.

Mit den Formfunktionen  $N_i(\xi, \eta)$  (10.91) lautet die quadratische Ansatzfunktion  $\tilde{u}(\xi, \eta)$  im Normaldreieck  $T$

$$\tilde{u}(\xi, \eta) = \sum_{i=1}^6 u_i N_i(\xi, \eta) = \mathbf{u}_e^T \mathbf{N}(\xi, \eta), \quad (10.92)$$

wo  $u_i$  den Funktionswert im Knotenpunkt  $\bar{P}_i$  bedeutet. Diese sechs Funktionswerte fassen wir im Vektor  $\mathbf{u}_e$  des Elementes zusammen, und die Formfunktionen im Vektor  $\mathbf{N}$ :

$$\begin{aligned} \mathbf{u}_e &:= (u_1, u_2, \dots, u_6)^T, \\ \mathbf{N}(\xi, \eta) &:= (N_1(\xi, \eta), N_2(\xi, \eta), \dots, N_6(\xi, \eta))^T. \end{aligned}$$

Mit (10.92) sind dann die partiellen Ableitungen

$$\tilde{u}_\xi = \mathbf{u}_e^T \mathbf{N}_\xi(\xi, \eta), \quad \tilde{u}_\eta = \mathbf{u}_e^T \mathbf{N}_\eta(\xi, \eta). \quad (10.93)$$

Für die drei Teilintegrale in (10.88), welche nicht von der Geometrie des Dreieckelementes abhängen, erhalten wir mit der Identität

$$\begin{aligned}
 (\mathbf{u}_e^T \mathbf{N}_\xi)^2 &= (\mathbf{u}_e^T \mathbf{N}_\xi)(\mathbf{N}_\xi^T \mathbf{u}_e) = \mathbf{u}_e^T \mathbf{N}_\xi \mathbf{N}_\xi^T \mathbf{u}_e \\
 I_1 &:= \iint_T \tilde{u}_\xi^2 d\xi d\eta = \iint_T \{\mathbf{u}_e^T \mathbf{N}_\xi\}^2 d\xi d\eta \\
 &= \mathbf{u}_e^T \left\{ \iint_T \mathbf{N}_\xi \mathbf{N}_\xi^T d\xi d\eta \right\} \mathbf{u}_e = \mathbf{u}_e^T \mathbf{S}_1 \mathbf{u}_e, \\
 I_2 &:= 2 \iint_T \tilde{u}_\xi \tilde{u}_\eta d\xi d\eta = \mathbf{u}_e^T \left\{ \iint_T [\mathbf{N}_\xi \mathbf{N}_\eta^T + \mathbf{N}_\eta \mathbf{N}_\xi^T] d\xi d\eta \right\} \mathbf{u}_e = \mathbf{u}_e^T \mathbf{S}_2 \mathbf{u}_e, \\
 I_3 &:= \iint_T \tilde{u}_\eta^2 d\xi d\eta = \mathbf{u}_e^T \left\{ \iint_T \mathbf{N}_\eta \mathbf{N}_\eta^T d\xi d\eta \right\} \mathbf{u}_e = \mathbf{u}_e^T \mathbf{S}_3 \mathbf{u}_e.
 \end{aligned} \tag{10.94}$$

Im Sinn des Matrizenproduktes stellt beispielsweise  $\mathbf{N}_\xi \mathbf{N}_\xi^T$  als Produkt eines Spaltenvektors mit einem Zeilenvektor eine Matrix der Ordnung sechs dar, und das Integral ist komponentenweise zu verstehen. Um auch für  $I_2$  darstellungsgemäß eine symmetrische Matrix als Integranden zu erhalten, ist  $2\tilde{u}_\xi \tilde{u}_\eta$  in zwei Summanden aufgeteilt worden.  $\mathbf{S}_1, \mathbf{S}_2$  und  $\mathbf{S}_3$  sind symmetrische, sechsreihige Matrizen. Sie müssen einmal ausgerechnet werden für den quadratischen Ansatz und bilden dann die Grundlage zur Berechnung des Beitrages des Dreieckelementes  $T_i$  gemäß (10.88) mit den Koeffizienten (10.89). Die drei Matrizen  $\mathbf{S}_i$  in (10.94) erhält man mit den partiellen Ableitungen der Formfunktionen nach einer längeren, aber elementaren Rechnung unter Verwendung der Integrationsformel

$$I_{p,q} := \iint_T \xi^p \eta^q d\xi d\eta = \frac{p! q!}{(p+q+2)!}, \quad p, q \in \mathbb{N}.$$

Wir fassen das Ergebnis zusammen in der so genannten *Steifigkeitselementmatrix*  $\mathbf{S}_e \in \mathbb{R}^{6,6}$  (auch Steifigkeitsmatrix genannt) eines Dreieckelementes  $T_i$  mit quadratischem Ansatz.

$$\begin{aligned}
 &\iint_{T_i} (\tilde{u}_x^2 + \tilde{u}_y^2) dx dy = \mathbf{u}_e^T \mathbf{S}_e \mathbf{u}_e \\
 \mathbf{S}_e = \frac{1}{6} &\begin{pmatrix} 3(a+2b+c) & a+b & b+c & -4(a+b) & 0 & -4(b+c) \\ a+b & 3a & -b & -4(a+b) & 4b & 0 \\ b+c & -b & 3c & 0 & 4b & -4(b+c) \\ -4(a+b) & -4(a+b) & 0 & 8(a+b+c) & -8(b+c) & 8b \\ 0 & 4b & 4b & -8(b+c) & 8(a+b+c) & -8(a+b) \\ -4(b+c) & 0 & -4(b+c) & 8b & -8(a+b) & 8(a+b+c) \end{pmatrix}
 \end{aligned} \tag{10.95}$$

Die Geometrie der Dreieckelemente  $T_i$  ist in den Koeffizienten  $a, b$  und  $c$  (10.89) enthalten. Zueinander ähnliche, aber beliebig gedrehte Dreieckelemente besitzen wegen (10.89) identische Steifigkeitselementmatrizen.

Für das Integral über  $u^2$  in (10.72) ergibt sich nach seiner Transformation auf das Normaldreieck  $T$  die Darstellung

$$\begin{aligned} I_4 := \iint_{T_i} \tilde{u}^2(x, y) dx dy &= J \iint_T \tilde{u}^2(\xi, \eta) d\xi d\eta = J \iint_T \{\mathbf{u}_e^T \mathbf{N}\}^2 d\xi d\eta \\ &= \mathbf{u}_e^T \left\{ J \iint_T \mathbf{N} \mathbf{N}^T d\xi d\eta \right\} \mathbf{u}_e = \mathbf{u}_e^T \mathbf{M}_e \mathbf{u}_e. \end{aligned}$$

Die Berechnung der Matrixelemente der so genannten *Massenelementmatrix*  $\mathbf{M}_e \in \mathbb{R}^{6,6}$  eines Dreieckelementes  $T_i$  mit quadratischem Ansatz ist elementar, wenn auch aufwändig. Wir fassen wieder zusammen.

$$\iint_{T_i} \tilde{u}^2(x, y) dx dy = \mathbf{u}_e^T \mathbf{M}_e \mathbf{u}_e$$

$$\mathbf{M}_e = \frac{J}{360} \begin{pmatrix} 6 & -1 & -1 & 0 & -4 & 0 \\ -1 & 6 & -1 & 0 & 0 & -4 \\ -1 & -1 & 6 & -4 & 0 & 0 \\ 0 & 0 & -4 & 32 & 16 & 16 \\ -4 & 0 & 0 & 16 & 32 & 16 \\ 0 & -4 & 0 & 16 & 16 & 32 \end{pmatrix}$$

(10.96)

Die Geometrie des Dreieckelementes erscheint in der Massenelementmatrix  $\mathbf{M}_e$  allein in Form des gemeinsamen Faktors  $J$ , der doppelten Fläche des Dreiecks. Die Form des Dreiecks beeinflusst die Zahlenwerte nicht.

Das Integral in (10.72) mit einem in  $u$  linearen Term ergibt sich zu

$$\begin{aligned} I_5 := \iint_{T_i} \tilde{u}(x, y) dx dy &= J \iint_T \tilde{u}(\xi, \eta) d\xi d\eta \\ &= J \iint_T \mathbf{u}_e^T \mathbf{N} d\xi d\eta = \mathbf{u}_e^T \mathbf{b}_e = \mathbf{b}_e^T \mathbf{u}_e. \end{aligned}$$

Die Komponenten des *Elementvektors*  $\mathbf{b}_e$  erhält man sehr einfach durch Integration der Formfunktionen über das Normaldreieck  $T$ . Das Ergebnis ist

$$\iint_{T_i} \tilde{u}(x, y) dx dy = \mathbf{b}_e^T \mathbf{u}_e, \quad \mathbf{b}_e = \frac{J}{6} (0, 0, 0, 1, 1, 1)^T.$$

(10.97)

Das Resultat (10.97) stellt eine interpolatorische Quadraturformel für ein Dreieck auf der Basis einer quadratischen Interpolation dar. Bemerkenswert ist die Tatsache, dass nur die Funktionswerte in den Seitenmittelpunkten mit dem gleichen Gewicht in die Formel eingehen, und dass die Geometrie des Dreiecks einzig in Form der doppelten Fläche erscheint.

Jetzt bleiben noch die beiden Randintegrale zu behandeln. Auf Grund unserer Triangulierung wird der Rand stets durch einen Polygonzug approximiert. Somit müssen wir uns

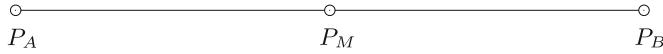


Abb. 10.27 Randkante mit Knotenpunkten.

mit der Betrachtung der Beiträge der Randintegrale für eine Dreiecksseite befassen. Die betrachtete Ansatzfunktion ist dort eine quadratische Funktion der Bogenlänge und ist durch die Funktionswerte in den drei Knotenpunkten eindeutig bestimmt. Wir betrachten deshalb eine Randkante  $R_i$  in allgemeiner Lage mit der Länge  $L$ . Ihre Endpunkte seien  $P_A$  und  $P_B$  und der Mittelpunkt  $P_M$  (Abb. 10.27). Die Funktionswerte in diesen Knotenpunkten bezeichnen wir mit  $u_A, u_B$  und  $u_M$  und fassen sie im Vektor  $\mathbf{u}_R := (u_A, u_M, u_B)^T$  zusammen. Zur Berechnung der Randintegrale führen wir die Substitution  $s = L\sigma$  durch, womit eine Abbildung auf das Einheitsintervall erfolgt. Zur Darstellung der quadratischen Funktion  $\tilde{u}(\sigma)$  verwenden wir die drei Lagrange-Polynome

$$N_1(\sigma) = (1 - \sigma)(1 - 2\sigma), \quad N_2(\sigma) = 4\sigma(1 - \sigma), \quad N_3(\sigma) = \sigma(2\sigma - 1),$$

die man als Basis- oder *Formfunktionen* für die Randstücke bezeichnet. Wir fassen sie im Vektor  $\mathbf{N}(\sigma) := (N_1(\sigma), N_2(\sigma), N_3(\sigma))^T \in \mathbb{R}^3$  zusammen, und erhalten so für das erste Randintegral

$$\begin{aligned} I_6 &:= \int_{R_i} \tilde{u}^2(s) ds = L \int_0^1 \tilde{u}^2(\sigma) d\sigma = L \int_0^1 \{\mathbf{u}_R^T \mathbf{N}(\sigma)\}^2 d\sigma \\ &= \mathbf{u}_R^T \left\{ L \int_0^1 \mathbf{N} \mathbf{N}^T d\sigma \right\} \mathbf{u}_R = \mathbf{u}_R^T \mathbf{M}_R \mathbf{u}_R. \end{aligned}$$

Als Resultat der Integration der dreireihigen Matrix erhalten wir die *Massenelementmatrix*  $\mathbf{M}_R$  einer geradlinigen Randkante  $R_i$  mit quadratischer Ansatzfunktion

$$\int_{R_i} \tilde{u}^2(s) ds = \mathbf{u}_R^T \mathbf{M}_R \mathbf{u}_R, \quad \mathbf{M}_R = \frac{L}{30} \begin{pmatrix} 4 & 2 & -1 \\ 2 & 16 & 2 \\ -1 & 2 & 4 \end{pmatrix}. \quad (10.98)$$

Das zweite Randintegral ist durch die Simpson-Regel (7.11) gegeben, da  $\tilde{u}(s)$  eine quadratische Funktion der Bogenlänge  $s$  und damit gleich dem Interpolationspolynom ist. Mit dem Randelementvektor  $\mathbf{b}_R \in \mathbb{R}^3$  gilt somit

$$\int_{R_i} \tilde{u}(s) ds = \mathbf{b}_R^T \mathbf{u}_R, \quad \mathbf{b}_R = \frac{L}{6} (1, 4, 1)^T. \quad (10.99)$$

### 10.3.4 Aufbau und Behandlung der linearen Gleichungen

Für die praktische Durchführung der Methode der finiten Elemente auf einem Computer ist es am zweckmäßigsten, alle  $N$  Knotenpunkte der vorgenommenen Triangulierung durchzunummerieren, also auch diejenigen, in denen die Funktionswerte durch eine Dirichletsche Randbedingung (10.73) vorgegeben sind. Die Nummerierung der Knotenpunkte sollte so erfolgen, dass die maximale Differenz zwischen Nummern, welche zu einem Element gehören, minimal ist, damit die Bandbreite der Matrix  $\mathbf{A}$  im System (10.82) möglichst klein ist. Es existieren heuristische Algorithmen, die eine nahezu optimale Nummerierung in akzeptabler Zeit  $O(N)$  systematisch finden [Duf 86, Sch 91b]. Mit dieser Nummerierung erfolgt die Summation der Beiträge der einzelnen Dreieckselemente und Randkanten zur Matrix  $\mathbf{A}$  und rechten Seite  $\mathbf{b}$  des Gleichungssystems (10.82) für das gesamte triangulierte Gebiet

$$\mathbf{A}\mathbf{u} = \mathbf{b}, \quad \mathbf{A} \in \mathbb{R}^{N,N}, \quad \mathbf{b}, \mathbf{u} \in \mathbb{R}^N. \quad (10.100)$$

Zunächst baut sich  $\mathbf{A}$  aus den Steifigkeits- und Massenelementmatrizen und  $\mathbf{b}$  aus den Elementvektoren entsprechend der Nummerierung auf. Die Unbekannten in (10.100) sind die Werte der  $N$  Knotenvariablen  $u_1, u_2, \dots, u_N$ . Man bezeichnet diesen Schritt, in welchem  $\mathbf{A}$  und  $\mathbf{b}$  gebildet werden, als *Kompilationsprozess*. Die *Gesamtsteifigkeitsmatrix*  $\mathbf{A}$  ist symmetrisch, kann aber noch singulär sein. Regulär und sogar positiv definit wird sie durch die Berücksichtigung der Randbedingungen (10.73) in den betreffenden Randknotenpunkten. Dies kann auf unterschiedliche Art und Weise erfolgen. Wir wollen hier die einfachste Methode vorschlagen. Im Knotenpunkt mit der Nummer  $k$  sei  $u_k = \varphi_k$  gegeben. Das bedeutet, dass das  $\varphi_k$ -fache der  $k$ -ten Spalte von  $\mathbf{A}$  vom Vektor  $\mathbf{b}$  abzuziehen ist. Im Prinzip wären danach in  $\mathbf{A}$  die  $k$ -te Spalte und die  $k$ -te Zeile und in  $\mathbf{b}$  die  $k$ -te Komponente zu streichen. Den Aufwand dieser beiden Schritte erspart man sich, wenn stattdessen in  $\mathbf{A}$  die  $k$ -te Spalte und Zeile durch Einheitsvektoren ersetzt werden, und  $b_k := \varphi_k$  gesetzt wird. Sobald alle Randknotenpunkte mit Dirichletscher Randbedingung auf diese Weise behandelt wurden, entsteht ein Gleichungssystem mit positiv definiter Matrix  $\mathbf{A} \in \mathbb{R}^{N,N}$ . Allerdings enthält das Gleichungssystem eine Reihe von trivialen Gleichungen, die den Randbedingungen entsprechen. Der Lösungsvektor  $\mathbf{u} \in \mathbb{R}^N$  enthält auf diese Weise aber auch diejenigen Knotenvariablen, deren Werte durch Dirichletsche Randbedingungen gegeben sind, was im Fall einer Weiterverarbeitung sehr zweckmäßig ist, z.B. zur Bestimmung von Niveaulinien.

Das symmetrische positiv definite Gleichungssystem  $\mathbf{A}\mathbf{u} = \mathbf{b}$  kann mit der Methode von Cholesky unter Ausnutzung der Bandstruktur gelöst werden. Die Bandbreite von  $\mathbf{A}$  variiert allerdings bei den meisten Anwendungen sehr stark, so dass die so genannte *hüllenorientierte* Rechentechnik Vorteile bringt. Bei größeren Gleichungssystemen wird man spezielle iterative Methoden anwenden, die wir im Kapitel 11 behandeln.

### 10.3.5 Beispiele

In diesem Abschnitt sollen zwei Beispiele rechnerisch behandelt werden. Dabei wollen wir die unterschiedlichen Möglichkeiten verschiedener Softwaresystem demonstrieren, siehe auch Abschnitt 10.4. Es gibt gerade in einem auch für industrielle Anwendungen so wichtigen Gebiet, wie es die Anwendungen von partiellen Differentialgleichungen und die Lösungen mit der Methode der finiten Elemente darstellen, einen erheblichen Unterschied zwischen

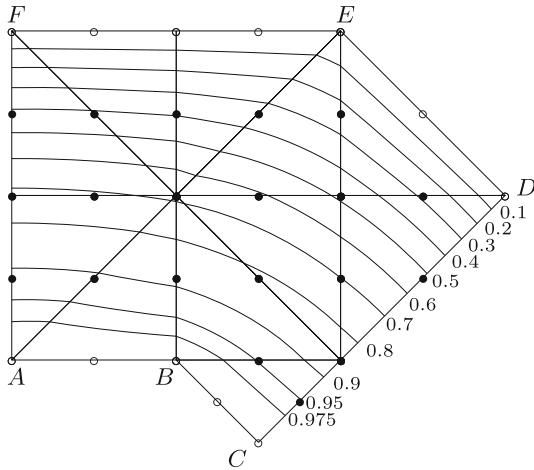


Abb. 10.28 Gebiet mit Triangulierung und Niveaulinien der Näherungslösung.

frei zugänglicher, oft durch eigene Programme ergänzter Software einerseits und teurer, aber auch komfortabler kommerzieller Software andererseits. Dabei gibt es innerhalb beider Gruppen noch erhebliche Unterschiede. So arbeitet der überwiegende Anteil der zahlreichen Programm pakete ausschließlich mit linearen Ansätzen, obwohl quadratische Ansätze in der Regel genauere Lösungen und bessere graphische Darstellungen ermöglichen. Andererseits liefern gerade die kleineren frei zugänglichen Pakete oft nur Zahlen, aber keine graphische Darstellung als Ergebnis.

**Beispiel 10.12.** Wir wollen zunächst ein Beispiel mit einem einfachen selbst geschriebenen Programm, wie es auch in [Sch 91a] zu finden ist, rechnen. Die berechneten Lösungswerte haben wir in MATLAB importiert und graphisch ausgewertet. Dieser einfachen Vorgehensweise stellen wir die Ergebnisse des Pakets PLTMG gegenüber.

Im Gebiet  $G$  der Abb. 10.6 soll die Randwertaufgabe (10.14) bis (10.17) mit quadratischen Ansätzen auf Dreiecken gelöst werden. Die zugehörige Formulierung als Variationsproblem lautet

$$I = \iint_G \left\{ \frac{1}{2}(u_x^2 + u_y^2) - 2u \right\} dx dy \longrightarrow \text{Min!} \quad (10.101)$$

unter den Randbedingungen

$$\begin{aligned} u &= 0 && \text{auf } DE \text{ und } EF, \\ u &= 1 && \text{auf } AB \text{ und } BC. \end{aligned} \quad (10.102)$$

Zu Vergleichszwecken mit Beispiel 10.1 verwenden wir eine recht grobe Triangulierung des Gebietes nach Abb. 10.28 in elf Dreieckelemente. Von den insgesamt  $N = 32$  Knotenpunkten haben zwölf bekannte Randwerte (als leere Kreise gekennzeichnet), so verbleiben  $n = 20$  Knotenpunkte mit unbekannten Funktionswerten, die als ausgefüllte Kreise hervorgehoben sind. Da alle Dreiecke gleichschenklig rechtwinklig sind, sind alle Steifigkeitselementmatrizen identisch. Beginnen wir die

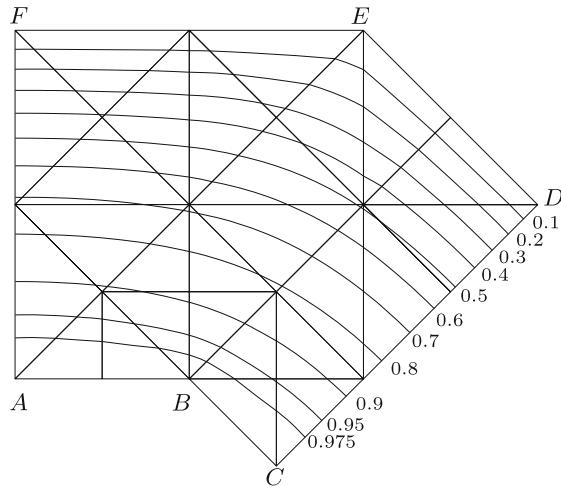


Abb. 10.29 Lokal verfeinerte Triangulierung und Niveaulinien.

Nummerierung stets im Eckpunkt mit dem rechten Winkel, so folgt wegen  $a = c = 1, b = 0$

$$\boldsymbol{S}_e = \frac{1}{6} \begin{pmatrix} 6 & 1 & 1 & -4 & 0 & -4 \\ 1 & 3 & 0 & -4 & 0 & 0 \\ 1 & 0 & 3 & 0 & 0 & -4 \\ -4 & -4 & 0 & 16 & -8 & 0 \\ 0 & 0 & 0 & -8 & 16 & -8 \\ -4 & 0 & -4 & 0 & -8 & 16 \end{pmatrix}.$$

Mit  $\boldsymbol{S}_e$  und dem Elementvektor  $\boldsymbol{b}_e$  kann das Gleichungssystem relativ leicht aufgebaut werden. Die resultierende Näherungslösung für die Randwertaufgabe ist in (10.103) in der Anordnung der Knotenpunkte zusammengestellt. Die Diskretisierungsfehler in dieser Näherung sind vergleichbar mit denjenigen von (10.20). In Abb. 10.28 sind einige Niveaulinien der Näherungslösung  $\tilde{u}(x, y)$  eingezeichnet. Man erkennt die Stetigkeit der Näherungslösung, aber auch die Unstetigkeit der ersten partiellen Ableitungen beim Übergang von einem Dreieckselement ins benachbarte. Besonders auffällig ist der Verlauf der Niveaulinien in der Nähe der einspringenden Ecke des Gebietes  $G$ . Dies ist bedingt durch die Singularität der partiellen Ableitungen der Lösungsfunktion in der Ecke.

$$\begin{array}{cccccc}
 0 & 0 & 0 & 0 & 0 \\
 0.41761 & 0.41129 & 0.38986 & 0.34098 & 0.23714 & 0 \\
 0.72286 & 0.71269 & 0.68215 & 0.61191 & 0.48260 & 0.28252 & 0 \\
 0.91668 & 0.90944 & 0.88338 & 0.81692 & 0.69997 & 0.52250 \\
 1 & 1 & 1 & 0.94741 & 0.85287 \\
 & & & 1 & 0.95255 \\
 & & & & 1
 \end{array} \tag{10.103}$$

Um dieser Tatsache besser Rechnung zu tragen, ist die Randwertaufgabe mit der feineren Triangulation von Abb. 10.29 behandelt worden. In der Nähe der einspringenden Ecke ist die Einteilung zusätzlich verfeinert worden. Bei insgesamt  $N = 65$  Knotenpunkten sind die Funktionswerte in

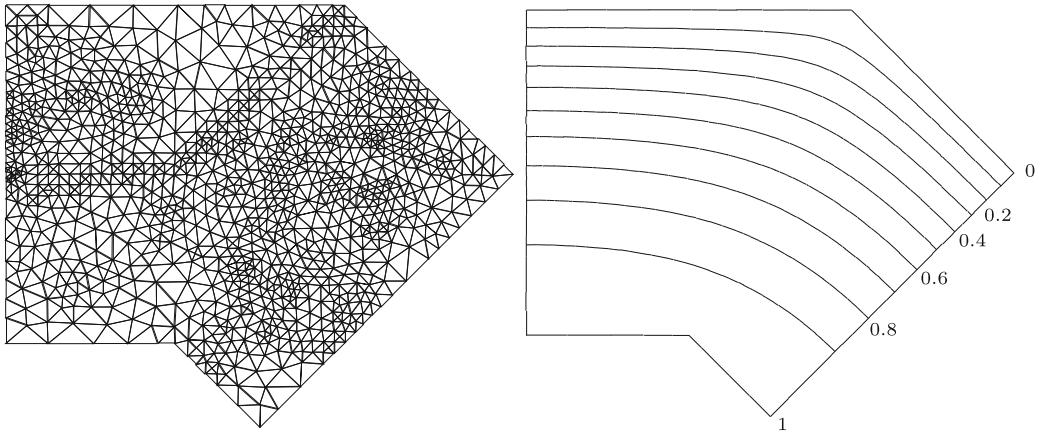


Abb. 10.30 PLTMG-Triangulierung und Niveaulinien.

$n = 49$  Knotenpunkten unbekannt. Die Niveaulinien verlaufen jetzt glatter, obwohl der Einfluss der Ecken immer noch vorhanden ist. Die Näherungswerte in denjenigen Knotenpunkten, die (10.103) entsprechen, sind in (10.104) zusammengestellt.

$$\begin{array}{cccccc}
 0 & 0 & 0 & 0 & 0 \\
 0.41793 & 0.41165 & 0.39081 & 0.3396 & 0.22979 & 0 \\
 0.72144 & 0.71286 & 0.68078 & 0.61282 & 0.48765 & 0.28239 & 0 \\
 0.91712 & 0.91006 & 0.88167 & 0.81686 & 0.70061 & 0.52275 & \\
 1 & 1 & 1 & 0.94683 & 0.85439 & \\
 & & & 1 & 0.95312 & \\
 & & & & 1 &
 \end{array} \tag{10.104}$$

Hier wäre also eine weitere – möglichst adaptive – Verfeinerung notwendig. Wir rechnen deshalb das Beispiel noch einmal mit dem Paket PLTMG, siehe Abschnitt 10.4. Wir lassen PLTMG das Gebiet triangulieren und anschließend dreimal adaptiv verfeinern. Das resultierende Dreiecksnetz und einige Niveaulinien sind in Abb. 10.30 zu sehen. Trotz nur linearen Ansatzes sind auf Grund der feinen Triangulierung die Niveaulinien sehr glatt. Natürlich sind auch die Lösungswerte genauer.

△

**Beispiel 10.13.** Wir betrachten die Randwertaufgabe (10.29) von Beispiel 10.5 für das Gebiet  $G$  mit dem Kreisbogen  $AB$  als Randstück (vgl. Abb. 10.10). Die zugehörige Formulierung als Variationsproblem lautet

$$\begin{aligned}
 I &= \iint_G \left\{ \frac{1}{2}(u_x^2 + u_y^2) - 2u \right\} dx dy + \int_{AB} \{u^2 + u\} ds \longrightarrow \text{Min!} \\
 u &= 0 \quad \text{auf } CD.
 \end{aligned} \tag{10.105}$$

Das Randintegral erstreckt sich nur über den Kreisbogen  $AB$  mit  $\alpha(s) = 2$  und  $\beta(s) = -1$ .

Wir wollen diese Aufgabe mit dem kommerziellen Paket FEMLAB bearbeiten, siehe Abschnitt 10.4, auch um zu demonstrieren, wie komfortabel ein solches Werkzeug zur Lösung partieller Differenzialgleichungen sein kann.

Zunächst definieren wir das Grundgebiet  $G$  als Differenz eines Dreiecks und eines Kreises. Das gelingt mit fünf Mausklicks im geometrischen Eingabefenster. Dann erzeugen wir mit einem Mausklick eine Anfangstriangulierung, die 375 Knotenpunkte enthält. Das mag viel erscheinen, liegt aber an den Standard-Vorgaben des FEMLAB-Systems, die man auch leicht ändern kann. Wir definieren die Differenzialgleichung, die Randbedingungen und die Wahl der quadratischen Ansatzfunktionen über entsprechende Parameter-Eingabefenster, bestimmen eine Lösung mit einem Mausklick und bestimmen in einem Fenster mit einer großen Auswahl an Parametern, dass wir die Lösung mit Hilfe von Niveaulinien zu vorgegebenen Lösungswerten darstellen wollen. Dies ergibt die obere der Abb. 10.31. Eine adaptive Verfeinerung führt zu einer Triangulierung mit 1429 Knotenpunkten und der zugehörigen Lösung. In der Nähe der Ecken kann man deutlich die Verbesserung durch die Verfeinerung erkennen.

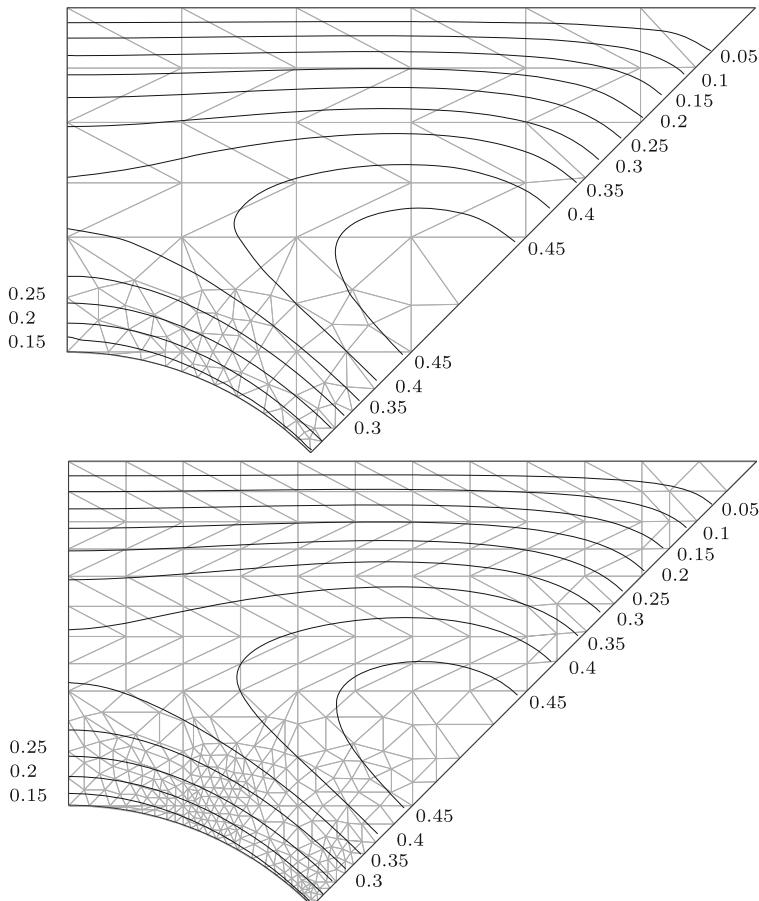


Abb. 10.31 Zwei FEMLAB-Gitter mit zugehörigen Niveaulinien.



## 10.4 Software

Die numerische Lösung partieller Differenzialgleichungen setzt sich aus vielen Anteilen zusammen. Nachdem wir in diesem Kapitel einige Methoden zur Diskretisierung partieller Differenzialgleichungen kennen gelernt haben, sollen Rüdes Grundprinzipien zur effizienten Lösung von partiellen Differenzialgleichungen einmal genannt werden, [Rüd 93]:

- Gute Diskretisierungsmethoden (verschieden hoher Ordnung).
- Schnelle Lösungsmethoden für die entstehenden Gleichungssysteme.
- Adaptivität.
- Hochwertige Informatikanteile (Hardware, Algorithmen und Software).

Bei der Diskretisierung, insbesondere bei der Methode der finiten Elemente, entsteht das Problem der Gittererzeugung bzw. Netzgenerierung, mit dem wir uns im Rahmen dieses Bandes nicht beschäftigen können. Mit der Lösung der bei der Diskretisierung entstehenden großen, schwach besetzten Gleichungssysteme werden wir uns in Kapitel 11 beschäftigen. Die Adaptivität spielt bei beiden Problemklassen eine große Rolle. Wir haben sie im letzten Beispiel 10.13 schon praktisch kennen gelernt. Weiter kann sie hier nicht behandelt werden. Dasselbe gilt für die Verwendung hochwertiger Hardware – wie Parallelrechner – und den Einsatz von Software, die strengen Anforderungen des Softwareengineering genügt, und die die Möglichkeiten spezieller Hardware algorithmisch nutzt. Alle diese Kriterien spielen bei der Auswahl von Software eine Rolle. Hinzu kommt die Berücksichtigung des Anwendungsbereichs.

PLTMG<sup>1</sup> ist ein frei erhältliches Paket, das seit einigen Jahrzehnten im Bereich von Universitäten und Forschungslabors beliebt ist. Der Name *Piecewise Linear Triangle Multi-Grid* verrät die verwendete Methode. Es erscheinen regelmäßig neue verbesserte Versionen, oft gefolgt von Buch-Neuerscheinungen bei SIAM [Ban 98]. PLTMG löst Variationsprobleme, die über (10.72) weit hinausgehen. So ist die Einbeziehung von Parametern und damit die Lösung von Eigenwertproblemen möglich. Seine Benutzung ist allerdings sehr gewöhnungsbedürftig, da die Definition des Gebietes und die Festlegung vieler die Lösung beeinflussender Parameter über vom Benutzer zu schreibende FORTRAN-Programme geschieht. Dabei werden die vielen Kontrollparameter über lange Felder ohne mnemotechnische Hilfestellung eingegeben. Belohnt wird man mit flexiblen Möglichkeiten bei der graphischen Darstellung der Lösung.

Für einfache lineare Beispielprobleme ist die *Partial Differential Equation Toolbox pdetool* des MATLAB-Systems eine große Hilfe. Sie benutzt die Methode der finiten Elemente, für die sie ein Dreiecksnetz erzeugt. Das Netz kann homogen oder adaptiv verfeinert werden. Eine graphische Benutzeroberfläche macht die Konstruktion eines Gebietes  $G$ , die Problemdefinition und die Eingabe von Kontrollparametern ausgesprochen einfach.

Auch FEMLAB kann als MATLAB-Toolbox betrachtet werden, baut es doch auf der MATLAB-Sprache und Funktionalität auf. Es ist als komfortabler Ersatz der Partial Differential Equation Toolbox gedacht und verfügt über eine bis auf Erweiterungen nahezu identische graphische Benutzeroberfläche. FEMLAB wird aus MATLAB heraus aufgerufen; die Ergebnisse können in MATLAB weiter verarbeitet werden. Es gibt aber eine stand-alone-Version, die

---

<sup>1</sup><http://ccom.ucsd.edu/~reb/software.html>

unabhängig von MATLAB ist. Sie hat C++-basierte Löser, die wesentlich schneller sind als die der MATLAB-basierten Version. Der Leistungsumfang von FEMLAB übersteigt den von `pdetool` bei weitem, insbesondere bei Anwendungsproblemen. Es gibt zahlreiche Spezialmodule u.a. für strukturmechanische, chemische und elektromagnetische Anwendungen.

Ein eigenständiges Paket riesigen Umfangs mit starker Industrie-Orientierung ist ANSYS. Zu seinen Anwendungsmodulen zählt die Luft- und Raumfahrt, der Automobilbau, die Elektronik und die Biomedizin. Der Hersteller CAD-FEM gibt die eigene Zeitschrift *Infoplaner* heraus und hält Anwenderschulungen ab. Die graphischen Möglichkeiten von ANSYS z.B. in der Strömungsdynamik sind beeindruckend.

Die NAG-FORTRAN-Bibliothek enthält seit Mark 20 ein Kapitel D06 mit einer Reihe von Routinen zur Netzerzeugung. Natürlich kann man mit dieser Bibliothek die entstehenden schwach besetzten Gleichungssysteme optimal lösen. Es fehlt allerdings das Bindeglied des Kompilationsprozesses, den man selbst programmieren muss. Und es fehlt die Möglichkeit komfortabler graphischer Ausgabe.

Eine Reihe von Anwendungsmodulen für Wärmeleitung, Reaktions-Diffusionsgleichungen, Struktur- und Strömungsmechanik sowie Akustik stellt auch FASTFLO von CSIRO bereit. FASTFLO verfügt über bequeme graphische Ein- und Ausgabewerkzeuge und erlaubt lineare und quadratische Ansätze.

Die Aufzählung von Softwaresystemen zur Lösung von partiellen Differenzialgleichungen ließe sich noch lange fortsetzen. Stattdessen verweisen wir auf die Internetseiten

[http://homepage.usask.ca/~ijm451/finite/fe\\_resources/fe\\_resources.html](http://homepage.usask.ca/~ijm451/finite/fe_resources/fe_resources.html)

und komfortabel, bunt und schön

[http://comp.uark.edu/~jjrencis/femur/main\\_menu.html](http://comp.uark.edu/~jjrencis/femur/main_menu.html).

Bei der Suche beachte man, dass die Programme manchmal nur unter dem Namen des Herstellers statt unter dem Programmnamen verzeichnet sind, aber Strukturierung und Suchmöglichkeit erleichtern die Orientierung. Man findet grundsätzlich Links zu den Seiten der Hersteller. Wir haben deshalb auf diese Angaben hier weitgehend verzichtet.

## 10.5 Aufgaben

**10.1.** Für das Gebiet  $G$  in Abb. 10.32 sei die Randwertaufgabe gegeben:

$$\begin{aligned} -\Delta u &= 1 \quad \text{in } G, \\ u &= 1 \quad \text{auf } AB, \quad \frac{\partial u}{\partial n} = 0 \quad \text{auf } BC, \\ u &= 0 \quad \text{auf } CD, \quad \frac{\partial u}{\partial n} + 2u = 1 \quad \text{auf } DA. \end{aligned}$$

Man löse sie näherungsweise mit dem Differenzenverfahren unter Verwendung der Fünf-Punkte-Approximation für Gitterweiten  $h = 1/4$ ,  $h = 1/6$  und  $h = 1/8$ . Wie lauten die Differenzengleichungen für die Gitterpunkte auf der Seite  $DA$ ? Für welche Nummerierung der Gitterpunkte hat die Matrix des Systems von Differenzengleichungen eine minimale Bandbreite?

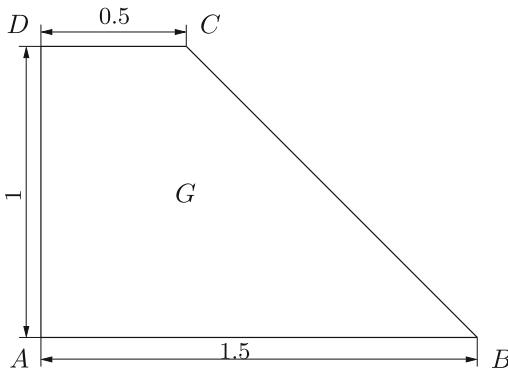


Abb. 10.32  
Trapezgebiet.

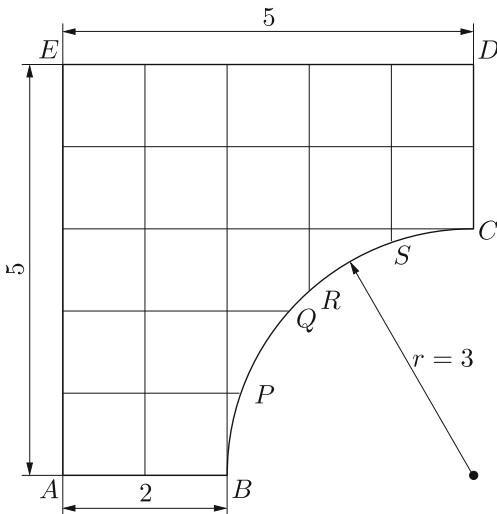


Abb. 10.33  
Gebiet mit Kreisrand.

**10.2.** Im Gebiet  $G$  in Abb. 10.33, dessen Randstück  $BC$  ein Viertelkreis mit dem Radius  $r = 3$  ist, soll die elliptische Randwertaufgabe gelöst werden:

$$\begin{aligned} -\Delta u &= 10 \quad \text{in } G, \\ \frac{\partial u}{\partial n} &= 0 \quad \text{auf } AB, \quad u = 4 \quad \text{auf } BC, \quad \frac{\partial u}{\partial n} = 0 \quad \text{auf } CD, \\ \frac{\partial u}{\partial n} + 4u &= 1 \quad \text{auf } DE, \quad u = 0 \quad \text{auf } EA. \end{aligned}$$

Wie lauten die Differenzengleichungen für die Gitterpunkte im Gitter von Abb. 10.33 mit der Gitterweite  $h = 1$ ? Ist das System der Differenzengleichungen symmetrisch?

**10.3.** Im Randwertproblem von Aufgabe 10.2 ersetze man die Dirichlet-Randbedingung auf  $BC$  durch die Cauchy-Randbedingung

$$\frac{\partial u}{\partial n} + 3u = 2.$$

Für die Randpunkte  $B, P, Q, R, S$  und  $C$  sind weitere Differenzengleichungen herzuleiten. Welche Struktur erhält das System von Differenzengleichungen? Die Berechnung seiner Lösung kann mit

dem Gauß-Algorithmus mit Diagonalstrategie erfolgen.

**10.4.** Man löse die parabolische Anfangsrandwertaufgabe

$$\begin{aligned} u_t &= u_{xx} + 1 \quad \text{für } 0 < x < 1, t > 0; \\ u(x, 0) &= 0 \quad \text{für } 0 < x < 1, \\ u_x(0, t) - 0.5u(0, t) &= 0, \quad u_x(1, t) + 0.2u(1, t) = 0 \quad \text{für } t > 0, \end{aligned}$$

mit der expliziten und der impliziten Methode für  $h = 0.1$  und  $h = 0.05$  und für verschiedene Zeitschritte  $k$ . Man leite die Bedingung der absoluten Stabilität im Fall der expliziten Methode für die obigen Randbedingungen her. Die stationäre Lösung  $u(x, t)$  für  $t \rightarrow \infty$  mit  $u_t = 0$  ist analytisch zu bestimmen und mit den berechneten Näherungen zu vergleichen.

**10.5.** Ein Diffusionsproblem besitzt folgende Formulierung:

$$\begin{aligned} \frac{\partial u}{\partial t} &= \frac{\partial}{\partial x} \left( (x^2 - x + 1) \frac{\partial u}{\partial x} \right) + (2 - x) \sin(t), \quad 0 < x < 1, t > 0, \\ u(x, 0) &= 0, \quad 0 < x < 1, \\ u(0, t) &= 0, \quad u_x(1, t) + 0.3u(1, t) = 0, \quad t > 0. \end{aligned}$$

Die Diffusionskennzahl ist ortsabhängig, und die Quellendichte ist orts- und zeitabhängig. Mit der Schrittweite  $h = 0.1$  bestimme man die Lösungsfunktion näherungsweise mit der expliziten und der impliziten Methode. Wie lautet die Bedingung der absoluten Stabilität für die Methode von Richardson?

**10.6.** Es soll die parabolische Differenzialgleichung

$$u_t = u_{xx} + u_{yy} + 1 \quad \text{in } G \times (0, \infty)$$

unter der Anfangsbedingung

$$u(x, y, 0) = 0 \quad \text{in } G$$

und den zeitunabhängigen Randbedingungen

$$\begin{aligned} u &= 1 \quad \text{auf } AB, \quad \frac{\partial u}{\partial n} = 0 \quad \text{auf } BC, \\ u &= 0 \quad \text{auf } CD, \quad \frac{\partial u}{\partial n} + 2u = 1 \quad \text{auf } DA \end{aligned}$$

gelöst werden, wo  $G$  das Gebiet von Abb. 10.32 ist. Man verwende dazu die explizite Methode von Richardson, das implizite Verfahren von Crank-Nicolson sowie die Methode der alternierenden Richtungen. Als Gitterweite wähle man  $h = 1/4$  und  $h = 1/6$ . Die stationäre Lösung für  $t \rightarrow \infty$  ist gleich der Lösung von Aufgabe 10.1.

**10.7.** Wie lauten die Steifigkeitselementmatrizen  $\mathbf{S}_e$  im Fall des quadratischen Ansatzes für  
a) ein gleichseitiges Dreieck;

b) ein rechtwinkliges Dreieck mit den Kathetenlängen  $\overline{P_1 P_2} = \alpha h$  und  $\overline{P_1 P_3} = \beta h$ ;

c) ein gleichschenkliges Dreieck mit der Schenkellänge  $h$  und dem Zwischenwinkel  $\gamma$ ?

Was folgt für die Matrixelemente von  $\mathbf{S}_e$  in den Fällen b) und c), falls  $\beta \ll \alpha$ , bzw.  $\gamma$  sehr klein ist?

Welchen Einfluss haben solche spitzwinkligen Dreieckelemente auf die Gesamtsteifigkeitsmatrix  $\mathbf{A}$ ?

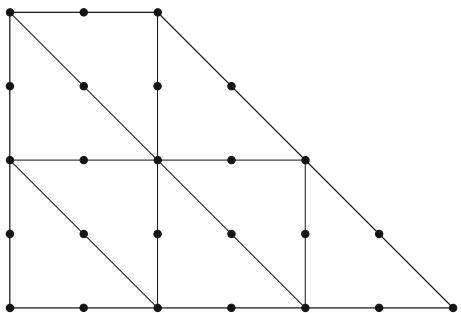


Abb. 10.34  
Grobe Triangulierung des Trapezgebietes.

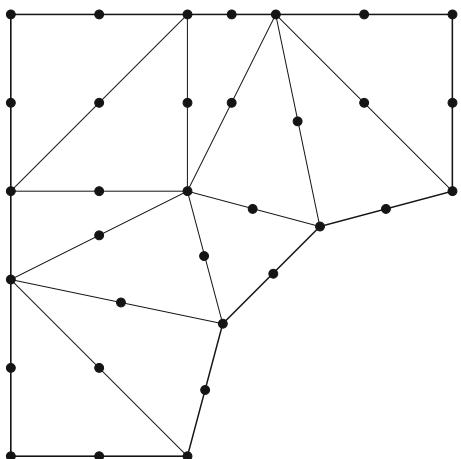


Abb. 10.35  
Triangulierung des Gebietes mit Kreisrand.

### 10.8. Linearer Ansatz in der Methode der finiten Elemente.

a) Wie lauten die Formfunktionen für den linearen Ansatz? Mit ihrer Hilfe leite man die Elementmatrizen zur Approximation des Integralausdrucks (10.72) unter der Annahme von konstanten Funktionen  $\varrho(x, y)$ ,  $f(x, y)$ ,  $\alpha(s)$  und  $\beta(s)$  her.

b) Welches sind die Steifigkeitsmatrizen  $S_e$  für ein gleichschenklig rechtwinkliges Dreieck, ein gleichseitiges Dreieck, ein rechtwinkliges Dreieck mit den Kathetenlängen  $\alpha h$  und  $\beta h$  sowie für ein gleichschenkliges Dreieck mit der Schenkellänge  $h$  und dem Zwischenwinkel  $\gamma$ ? Was passiert für spitzwinklige Dreiecke?

c) Man verifiziere, dass die Methode der finiten Elemente für die Poissonsche Differenzialgleichung  $-\Delta u = f(x, y)$  im Fall der linearen Elemente für jeden im Innern liegenden Knotenpunkt bei Verwendung einer regelmäßigen Triangulierung in kongruente, rechtwinklig gleichschenklige Dreiecke die Fünf-Punkte-Differenzengleichung (10.12) liefert, ganz unabhängig davon, wieviele Dreieckelemente im betreffenden Knotenpunkt zusammenstoßen.

**10.9.** Wie lauten die zu minimierenden Integralausdrücke zu den Randwertproblemen der Aufgaben 10.1 bis 10.3? Man löse jene Randwertaufgaben mit der Methode der finiten Elemente beispielsweise für die Triangulierungen der Abb. 10.34 und Abb. 10.35 unter Verwendung der quadratischen Ansätze.

## 11 Lineare Gleichungssysteme, iterative Verfahren

Die Behandlung von linearen elliptischen Randwertaufgaben mit dem Differenzenverfahren oder mit finiten Elementen führt auf die Aufgabe, lineare Gleichungssysteme mit symmetrischer oder gelegentlich unsymmetrischer Matrix für die unbekannten Funktionswerte in den Gitterpunkten zu lösen. Bei feiner Diskretisierung des Grundgebietes sind die Systeme einerseits von hoher Ordnung und besitzen andererseits die Eigenschaft, sehr schwach besetzt (engl. *sparse*) zu sein. Grundsätzlich können sie mit den direkten Methoden von Kapitel 2 gelöst werden, wobei bei geeigneter Nummerierung der Unbekannten die resultierende Bandstruktur ausgenutzt werden kann. Im Verlauf des Eliminationsprozesses erfolgt aber im Inneren des Bandes ein oft vollständiger Auffüllprozess (das sog. fill-in), bei welchem Matrixelemente, die ursprünglich gleich null sind, durch von null verschiedene Werte ersetzt werden. Dadurch kann für sehr große Gleichungssysteme neben dem Rechenaufwand insbesondere der Speicherbedarf prohibitiv groß werden. Deshalb erweisen sich iterative Verfahren zur Lösung von sehr großen, schwach besetzten linearen Gleichungssystemen als geeignete Alternativen, mit denen die schwache Besetzung voll ausgenutzt wird. Im Folgenden betrachten wir die klassischen Iterationsmethoden und zeigen einige ihrer wichtigsten Eigenschaften auf. Darauf aufbauend werden Mehrgittermethoden beschrieben, die zu den schnellsten Lösern für die genannten Probleme gehören. Dann wird die Methode der konjugierten Gradienten für symmetrische und positiv definite Gleichungssysteme ausführlich unter Einschluss der zentralen, die Konvergenz verbessern Vorkonditionierung behandelt. Daraus wird anschließend die Methode der verallgemeinerten minimierten Residuen zur Lösung von unsymmetrischen Gleichungssystemen entwickelt. Ausführlichere Darstellungen von Iterationsmethoden findet man etwa in [Bey 98, Bra 93, Bri 00, McC 88, Hac 85, Hac 93, Hag 04, Sto 05, Var 00, Wes 04, You 71].

### 11.1 Diskretisierung partieller Differenzialgleichungen

Im Abschnitt 10.1.2 wurde die Diskretisierung partieller Differenzialgleichungen mit dividierten Differenzen bereits behandelt. Hier soll ein einfaches Beispiel eine typische Struktur des entstehenden linearen Gleichungssystems zeigen.

**Beispiel 11.1.** Der Laplace-Operator wird auf einem quadratischen Gitter  $(x_i, y_j)$  mit der Gitterweite  $h$  approximiert durch einen Fünf-Punkte-Stern (siehe Abb. 10.3)

$$-\Delta u = -\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} \quad (11.1)$$

$$\approx \frac{1}{h^2}[-u(x_{i-1}, y_j) - u(x_{i+1}, y_j) + 4u(x_i, y_j) - u(x_i, y_{j-1}) - u(x_i, y_{j+1})].$$

Als Gebiet im  $\mathbb{R}^2$  wählen wir jetzt das Einheitsquadrat  $G := (0, 1) \times (0, 1)$  mit Rand  $\Gamma$ . Zu gegebener rechter Seite  $f(x, y) \in C(G)$  ist eine Funktion  $u(x, y) \in C^2(G)$  gesucht, die die Differenzialgleichung und die Randbedingung

$$\begin{aligned} -\Delta u &= f && \text{in } G \\ u &= 0 && \text{auf } \Gamma \end{aligned} \tag{11.2}$$

erfüllt. Mit  $x_i := i h$ ,  $y_j := j h$ ,  $h := 1/(N+1)$ , ist ein Gitter mit den Punkten  $P_{ij} := (x_i, y_j)$  auf  $G$  definiert, auf dem (11.2) diskretisiert werden soll.

Mit  $u_{ij} := u(x_i, y_j)$  bekommt man für alle inneren Punkte  $P_{ij}$  die Gleichungen

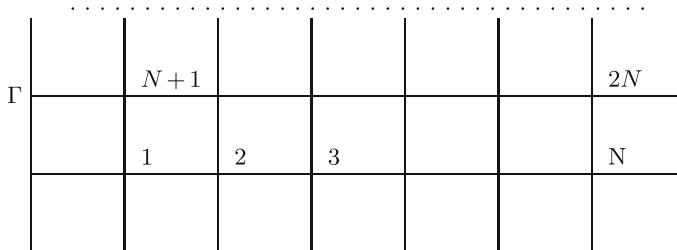
$$-u_{i-1,j} - u_{i+1,j} + 4u_{i,j} - u_{i,j-1} - u_{i,j+1} = h^2 f_{ij},$$

wobei in Randpunkten  $u_{i,j} = 0$  gesetzt wird.

Zur eindimensionalen Nummerierung können die Punkte in der Index-Reihenfolge

$$(1, 1), (2, 1), (3, 1), \dots, (N, 1), (1, 2), (2, 2), (3, 2), \dots, (N, N),$$

also geometrisch zeilenweise von unten nach oben geordnet werden:



Damit ergibt sich die  $N^2 \times N^2$  - Matrix

$$A = \begin{pmatrix} B & -I & 0 & \cdots & 0 \\ -I & B & -I & 0 & \\ 0 & \ddots & \ddots & \ddots & \ddots \\ \ddots & & & & -I \\ 0 & \cdots & 0 & -I & B \end{pmatrix} \tag{11.3}$$

mit den Blöcken

$$B = \begin{pmatrix} 4 & -1 & 0 & \cdots & 0 \\ -1 & 4 & -1 & 0 & \\ 0 & \ddots & \ddots & \ddots & \\ \vdots & \ddots & -1 & 4 & -1 \\ 0 & \cdots & 0 & -1 & 4 \end{pmatrix} \in \mathbb{R}^{N,N}, \quad I \in \mathbb{R}^{NN}.$$

Die Matrix hat die Bandbreite  $m = N$ , siehe Def. 2.19. Allerdings hat jede Zeile nur maximal fünf Elemente ungleich null. Da in Anwendungsproblemen  $N$  recht groß ist, sollten zur Lösung spezielle Iterationsverfahren angewendet werden.  $\triangle$

An diesem Beispiel wird deutlich, dass die Bandbreite bei diesem Problemtyp in der Größenordnung von  $\sqrt{n}$  liegt, wenn  $n$  die Ordnung des linearen Gleichungssystems ist. Wird das System z.B. mit einem Band-Cholesky-Verfahren gelöst, so werden wegen des fill-in  $n\sqrt{n}$  Speicherplätze benötigt und das Verfahren hat einen Aufwand  $O(n^2)$ .

Bei der iterativen Lösung schwach besetzter linearer Gleichungssysteme ist es das Ziel Verfahren zu entwickeln, die *asymptotisch optimal* sind, d.h.:

1. Es werden nur  $O(n)$  Operationen benötigt.
2. Der Aufwand ist unabhängig von der Diskretisierungsgröße  $h$ .

Nur wenige Verfahren erfüllen diese Eigenschaft; dazu gehören die Mehrgittermethoden. Oft ist sie nur für Modellprobleme beweisbar, wird aber im Experiment für allgemeine Probleme bestätigt.

## 11.2 Relaxationsverfahren

In diesem Abschnitt sollen die Verfahren von Jacobi und Gauß-Seidel in relaxierter und nicht relaxierter Form entwickelt werden. Diese iterativen Verfahren konvergieren unter Voraussetzungen, die für die wichtigsten Anwendungen erfüllt sind, aber sie konvergieren viel zu langsam, um als Einzelverfahren zur Lösung großer, schwach besetzter linearer Gleichungssysteme in Frage zu kommen. Andererseits haben sie Eigenschaften, die sie als ideale Verfahrensteile für eine der schnellsten Löser-Klassen, die Mehrgitter- oder Mehrstufenmethoden, auszeichnen. Mehrgittermethoden werden im nächsten Abschnitt behandelt.

### 11.2.1 Konstruktion der Iterationsverfahren

Wir betrachten ein allgemeines lineares Gleichungssystem

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \quad \mathbf{A} \in \mathbb{R}^{n,n}, \quad \mathbf{x}, \mathbf{b} \in \mathbb{R}^n \quad (11.4)$$

in  $n$  Unbekannten mit der *regulären* Matrix  $\mathbf{A}$ , so dass die Existenz und Eindeutigkeit der Lösung  $\mathbf{x}$  gewährleistet ist. Damit ein Gleichungssystem (11.4) iterativ gelöst werden kann, muss es in einer ersten Klasse von Verfahren zuerst in eine äquivalente Fixpunktform übergeführt werden, z.B.

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{B}^{-1}(\mathbf{b} - \mathbf{A}\mathbf{x}^{(k)}) = (\mathbf{I} - \mathbf{B}^{-1}\mathbf{A})\mathbf{x}^{(k)} + \mathbf{B}^{-1}\mathbf{b}. \quad (11.5)$$

Für  $\mathbf{B} = \mathbf{A}$  ist  $\mathbf{x}^{(1)}$  für einen beliebigen Startvektor  $\mathbf{x}^{(0)}$  die exakte Lösung. Deshalb sollte die Matrix  $\mathbf{B}$  so gewählt werden, dass einerseits  $\mathbf{B} \approx \mathbf{A}$  und andererseits die Inverse von  $\mathbf{B}$  leicht berechenbar ist. Unter den zahlreichen möglichen Varianten werden wir im Folgenden nur einige klassische Methoden betrachten. Zu ihrer Herleitung treffen wir die Zusatzannahme, dass im zu lösenden Gleichungssystem (11.4)

$$\sum_{j=1}^n a_{ij}x_j = b_i, \quad i = 1, 2, \dots, n, \quad (11.6)$$

für die Diagonalelemente von  $\mathbf{A}$

$$a_{ii} \neq 0, \quad i = 1, 2, \dots, n, \quad (11.7)$$

gilt. Die Voraussetzung (11.7) ist in der Regel bei einer zweckmäßigen Formulierung des Gleichungssystems automatisch erfüllt oder kann andernfalls durch eine geeignete Anordnung der Gleichungen erfüllt werden. Somit kann die  $i$ -te Gleichung von (11.6) nach der  $i$ -ten Unbekannten  $x_i$  aufgelöst werden.

$$x_i = -\frac{1}{a_{ii}} \left[ \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij} x_j - b_i \right], \quad i = 1, 2, \dots, n, \quad (11.8)$$

(11.8) und (11.6) stellen offensichtlich äquivalente Beziehungen dar. Durch (11.8) wird eine lineare Abbildung des  $\mathbb{R}^n$  in den  $\mathbb{R}^n$  definiert, für welche der Lösungsvektor  $\mathbf{x}$  von (11.4) ein Fixpunkt ist. Auf Grund dieser Tatsache können wir eine erste Iterationsvorschrift definieren gemäß

$$x_i^{(k+1)} = -\frac{1}{a_{ii}} \left[ \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij} x_j^{(k)} - b_i \right], \quad i = 1, 2, \dots, n; \quad k = 0, 1, 2, \dots \quad (11.9)$$

Hier ist offenbar in (11.5)  $\mathbf{B} = \mathbf{D} := \text{diag}(a_{11}, a_{22}, \dots, a_{nn})$ .

Da der iterierte Vektor  $\mathbf{x}^{(k)}$  in (11.9) als Ganzes in der rechten Seite eingesetzt wird, nennt man die Iterationsvorschrift das *Gesamtschrittverfahren*. Es ist jedoch üblich, die Methode (11.9) als *Jacobi-Verfahren* oder kurz als *J-Verfahren* zu bezeichnen.

Anstatt in der rechten Seite von (11.8) den alten iterierten Vektor einzusetzen, besteht eine naheliegende Modifikation darin, diejenigen Komponenten  $x_j^{(k+1)}$ , die schon neu berechnet wurden, zu verwenden. Das ergibt im Fall  $n = 4$  folgende geänderte Iterationsvorschrift:

$$\begin{aligned} x_1^{(k+1)} &= -[a_{12}x_2^{(k)} + a_{13}x_3^{(k)} + a_{14}x_4^{(k)} - b_1]/a_{11} \\ x_2^{(k+1)} &= -[a_{21}x_1^{(k+1)} + a_{23}x_3^{(k)} + a_{24}x_4^{(k)} - b_2]/a_{22} \\ x_3^{(k+1)} &= -[a_{31}x_1^{(k+1)} + a_{32}x_2^{(k+1)} + a_{34}x_4^{(k)} - b_3]/a_{33} \\ x_4^{(k+1)} &= -[a_{41}x_1^{(k+1)} + a_{42}x_2^{(k+1)} + a_{43}x_3^{(k+1)} - b_4]/a_{44} \end{aligned} \quad (11.10)$$

Allgemein lautet das *Einzelschrittverfahren* oder *Gauß-Seidel-Verfahren*

$$x_i^{(k+1)} = -\frac{1}{a_{ii}} \left[ \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} + \sum_{j=i+1}^n a_{ij} x_j^{(k)} - b_i \right], \quad i = 1, 2, \dots, n; \quad k = 0, 1, 2, \dots \quad (11.11)$$

Hier ist in (11.5)  $\mathbf{B}$  die untere Dreiecksmatrix von  $\mathbf{A}$ , wie wir unten sehen werden.

Die Reihenfolge, in welcher die Komponenten  $x_i^{(k+1)}$  des iterierten Vektors  $\mathbf{x}^{(k+1)}$  gemäß (11.11) berechnet werden, ist wesentlich, denn nur so ist diese Iterationsvorschrift explizit.

Die Rechenpraxis zeigt, und die später folgende Analyse wird dies bestätigen, dass das Konvergenzverhalten der Iterationsvektoren  $\boldsymbol{x}^{(k)}$  gegen den Fixpunkt  $\boldsymbol{x}$  oft ganz wesentlich verbessert werden kann, falls die Korrekturen der einzelnen Komponenten mit einem festen *Relaxationsparameter*  $\omega \neq 1$  multipliziert und dann addiert werden. Falls  $\omega > 1$  ist, spricht man von *Überrelaxation*, andernfalls von *Unterrelaxation*. Die geeignete Wahl des Relaxationsparameters  $\omega > 0$  ist entweder abhängig von Eigenschaften des zu lösenden Gleichungssystems oder aber von speziellen Zielsetzungen, wie etwa im Zusammenhang mit der so genannten Glättung bei Mehrgittermethoden, siehe Abschnitt 11.3.

Die Korrektur der  $i$ -ten Komponente im Fall des Jacobi-Verfahrens ist gemäß (11.9) gegeben durch

$$\Delta x_i^{(k+1)} = x_i^{(k+1)} - x_i^{(k)} = - \left[ \sum_{j=1}^n a_{ij} x_j^{(k)} - b_i \right] / a_{ii}, \quad i = 1, 2, \dots, n,$$

und das *JOR-Verfahren*, auch *gedämpfte Jacobi-Iteration* genannt, ist definiert durch

$$\begin{aligned} x_i^{(k+1)} &:= x_i^{(k)} + \omega \cdot \Delta x_i^{(k+1)} \\ &= x_i^{(k)} - \frac{\omega}{a_{ii}} \left[ \sum_{j=1}^n a_{ij} x_j^{(k)} - b_i \right] \\ &= (1 - \omega)x_i^{(k)} - \frac{\omega}{a_{ii}} \left[ \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij} x_j^{(k)} - b_i \right], \\ i &= 1, 2, \dots, n; \quad k = 0, 1, 2, \dots. \end{aligned} \tag{11.12}$$

In Analogie dazu resultiert aus dem Einzelschrittverfahren mit den aus (11.11) folgenden Korrekturen

$$\Delta x_i^{(k+1)} = x_i^{(k+1)} - x_i^{(k)} = - \left[ \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} + \sum_{j=i}^n a_{ij} x_j^{(k)} - b_i \right] / a_{ii}$$

die *Methode der sukzessiven Überrelaxation* (successive overrelaxation) oder abgekürzt das *SOR-Verfahren*

$$\begin{aligned} x_i^{(k+1)} &:= x_i^{(k)} - \frac{\omega}{a_{ii}} \left[ \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} + \sum_{j=i}^n a_{ij} x_j^{(k)} - b_i \right] \\ &= (1 - \omega)x_i^{(k)} - \frac{\omega}{a_{ii}} \left[ \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} + \sum_{j=i+1}^n a_{ij} x_j^{(k)} - b_i \right], \\ i &= 1, 2, \dots, n; \quad k = 0, 1, 2, \dots. \end{aligned} \tag{11.13}$$

Das JOR- und das SOR-Verfahren enthalten für  $\omega = 1$  als Spezialfälle das J-Verfahren beziehungsweise das Einzelschrittverfahren.

Als Vorbereitung für die nachfolgenden Konvergenzbetrachtungen sollen die Iterationsverfahren, welche komponentenweise und damit auf einem Computer unmittelbar implementierbar formuliert worden sind, auf eine einheitliche Form gebracht werden. Da die Diagonalelemente und die Nicht-Diagonalelemente der unteren und der oberen Hälfte der gegebenen Matrix  $\mathbf{A}$  eine zentrale Rolle spielen, wird die Matrix  $\mathbf{A}$  als Summe von drei Matrizen dargestellt gemäß

$$\boxed{\mathbf{A} := \mathbf{D} - \mathbf{L} - \mathbf{U} = \begin{pmatrix} \ddots & & -\mathbf{U} \\ & \mathbf{D} & \\ -\mathbf{L} & & \ddots \end{pmatrix}.} \quad (11.14)$$

Darin bedeutet  $\mathbf{D} := \text{diag}(a_{11}, a_{22}, \dots, a_{nn}) \in \mathbb{R}^{n,n}$  eine Diagonalmatrix, gebildet mit den Diagonalelementen von  $\mathbf{A}$ , die wegen der Voraussetzung (11.7) regulär ist.  $\mathbf{L}$  ist eine *strikt untere Linksdreiecksmatrix* mit den Elementen  $-a_{i,j}$ ,  $i > j$ , und  $\mathbf{U}$  ist eine *strikt obere Rechtsdreiecksmatrix* mit den Elementen  $-a_{i,j}$ ,  $i < j$ .

Die Iterationsvorschrift (11.9) des Gesamtschrittverfahrens ist nach Multiplikation mit  $a_{ii}$  äquivalent zu

$$\mathbf{D}\mathbf{x}^{(k+1)} = (\mathbf{L} + \mathbf{U})\mathbf{x}^{(k)} + \mathbf{b},$$

und infolge der erwähnten Regularität von  $\mathbf{D}$  ist dies gleichwertig zu

$$\mathbf{x}^{(k+1)} = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})\mathbf{x}^{(k)} + \mathbf{D}^{-1}\mathbf{b}. \quad (11.15)$$

Mit der *Iterationsmatrix*

$$\boxed{\mathbf{T}_J := \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})} \quad (11.16)$$

und dem Konstantenvektor  $\mathbf{c}_J := \mathbf{D}^{-1}\mathbf{b}$  kann das J-Verfahren (11.9) formuliert werden als

$$\boxed{\mathbf{x}^{(k+1)} = \mathbf{T}_J\mathbf{x}^{(k)} + \mathbf{c}_J, \quad k = 0, 1, 2, \dots} \quad (11.17)$$

In Analogie ist die Rechenvorschrift (11.11) des Einzelschrittverfahrens äquivalent zu

$$\mathbf{D}\mathbf{x}^{(k+1)} = \mathbf{L}\mathbf{x}^{(k+1)} + \mathbf{U}\mathbf{x}^{(k)} + \mathbf{b},$$

beziehungsweise nach anderer Zusammenfassung gleichwertig zu

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

Jetzt stellt  $(\mathbf{D} - \mathbf{L})$  eine Linksdreiecksmatrix mit von null verschiedenen Diagonalelementen dar und ist deshalb regulär. Folglich erhalten wir für das Einzelschrittverfahren

$$\mathbf{x}^{(k+1)} = (\mathbf{D} - \mathbf{L})^{-1}\mathbf{U}\mathbf{x}^{(k)} + (\mathbf{D} - \mathbf{L})^{-1}\mathbf{b}. \quad (11.18)$$

Mit der nach (11.18) definierten Iterationsmatrix

$$\boxed{\mathbf{T}_{ES} := (\mathbf{D} - \mathbf{L})^{-1}\mathbf{U}} \quad (11.19)$$

und dem entsprechenden Konstantenvektor  $\mathbf{c}_{ES} := (\mathbf{D} - \mathbf{L})^{-1}\mathbf{b}$  erhält (11.18) dieselbe Form wie (11.17).

Aus dem *JOR*-Verfahren (11.12) resultiert auf ähnliche Weise die äquivalente Matrizenformulierung

$$\mathbf{D}\mathbf{x}^{(k+1)} = [(1 - \omega)\mathbf{D} + \omega(\mathbf{L} + \mathbf{U})]\mathbf{x}^{(k)} + \omega\mathbf{b}.$$

Deshalb ergeben sich wegen

$$\mathbf{x}^{(k+1)} = [(1 - \omega)\mathbf{I} + \omega\mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})]\mathbf{x}^{(k)} + \omega\mathbf{D}^{-1}\mathbf{b} \quad (11.20)$$

einerseits die vom Relaxationsparameter  $\omega$  abhängige Iterationsmatrix des *JOR*-Verfahrens

$\mathbf{T}_{\text{JOR}}(\omega) := (1 - \omega)\mathbf{I} + \omega\mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$

(11.21)

und andererseits der Konstantenvektor  $\mathbf{c}_{\text{JOR}}(\omega) := \omega\mathbf{D}^{-1}\mathbf{b}$ . Für  $\omega = 1$  gelten offensichtlich  $\mathbf{T}_{\text{JOR}}(1) = \mathbf{T}_J$  und  $\mathbf{c}_{\text{JOR}}(1) = \mathbf{c}_J$ .

Aus der zweiten Darstellung der Iterationsvorschrift (11.13) des *SOR*-Verfahrens erhalten wir nach Multiplikation mit  $a_{ii}$

$$\mathbf{D}\mathbf{x}^{(k+1)} = (1 - \omega)\mathbf{D}\mathbf{x}^{(k)} + \omega\mathbf{L}\mathbf{x}^{(k+1)} + \omega\mathbf{U}\mathbf{x}^{(k)} + \omega\mathbf{b},$$

oder nach entsprechender Zusammenfassung

$$(\mathbf{D} - \omega\mathbf{L})\mathbf{x}^{(k+1)} = [(1 - \omega)\mathbf{D} + \omega\mathbf{U}]\mathbf{x}^{(k)} + \omega\mathbf{b}.$$

Darin ist  $(\mathbf{D} - \omega\mathbf{L})$  unabhängig von  $\omega$  eine reguläre Linksdreiecksmatrix, da ihre Diagonalelemente wegen (11.7) von null verschieden sind, und sie ist somit invertierbar. Folglich kann die Iterationsvorsschrift des *SOR*-Verfahrens geschrieben werden als

$$\mathbf{x}^{(k+1)} = (\mathbf{D} - \omega\mathbf{L})^{-1}[(1 - \omega)\mathbf{D} + \omega\mathbf{U}]\mathbf{x}^{(k)} + \omega(\mathbf{D} - \omega\mathbf{L})^{-1}\mathbf{b}. \quad (11.22)$$

Die von  $\omega$  abhängige Iterationsmatrix des *SOR*-Verfahrens ist deshalb definiert durch

$\mathbf{T}_{\text{SOR}}(\omega) := (\mathbf{D} - \omega\mathbf{L})^{-1}[(1 - \omega)\mathbf{D} + \omega\mathbf{U}],$

(11.23)

und der Konstantenvektor ist  $\mathbf{c}_{\text{SOR}}(\omega) := \omega(\mathbf{D} - \omega\mathbf{L})^{-1}\mathbf{b}$ , mit denen das *SOR*-Verfahren auch die Gestalt (11.17) erhält. Für  $\omega = 1$  gelten selbstverständlich  $\mathbf{T}_{\text{SOR}}(1) = \mathbf{T}_{\text{ES}}$  und  $\mathbf{c}_{\text{SOR}}(1) = \mathbf{c}_{\text{ES}}$ .

Alle betrachteten Iterationsverfahren haben damit die Form einer Fixpunktiteration

$$\mathbf{x}^{(k+1)} = \mathbf{T}\mathbf{x}^{(k)} + \mathbf{c}, \quad k = 0, 1, 2, \dots, \quad (11.24)$$

mit der speziellen Eigenschaft, dass sie *linear* und *stationär* sind. Denn die Iterationsmatrix  $\mathbf{T}$  und der Konstantenvektor  $\mathbf{c}$  sind nicht vom iterierten Vektor  $\mathbf{x}^{(k)}$  und auch nicht von  $k$  abhängig. Sowohl  $\mathbf{T}$  als auch  $\mathbf{c}$  sind konstant, falls im *JOR*- und *SOR*-Verfahren der Relaxationsparameter  $\omega$  fest gewählt wird. Zudem handelt es sich um *einstufige* Iterationsverfahren, da zur Bestimmung von  $\mathbf{x}^{(k+1)}$  nur  $\mathbf{x}^{(k)}$  und keine zurückliegenden Iterationsvektoren verwendet werden.

Wenn die bisher betrachteten Fixpunktiterationen zur iterativen Lösung von linearen Gleichungssystemen konvergent sind, dann konvergieren sie gegen die Lösung des Gleichungssystems  $\mathbf{Ax} = \mathbf{b}$ . Diese Eigenschaft wird wie folgt definiert:

**Definition 11.1.** Ein Gleichungssystem  $\mathbf{A}\mathbf{x} = \mathbf{b}$  heißt *vollständig konsistent* mit einer Fixpunktgleichung  $\mathbf{x} = \mathbf{T}\mathbf{x} + \mathbf{c}$ , wenn jede Lösung der einen Gleichung auch Lösung der anderen ist.

### 11.2.2 Einige Konvergenzsätze

Zuerst wollen wir allgemein die notwendigen und hinreichenden Bedingungen dafür erkennen, dass die lineare stationäre Fixpunktgleichung (11.24) eine gegen den Fixpunkt konvergente Vektorfolge  $\mathbf{x}^{(k)}$  erzeugt. Auf Grund dieses Ergebnisses werden dann einige Konvergenzaussagen hergeleitet, die auf bestimmten Eigenschaften der Matrix  $\mathbf{A}$  beruhen. Für den ersten Punkt können wir den Banachschen Fixpunktsatz 4.4 heranziehen und brauchen dessen Aussagen nur auf den vorliegenden Spezialfall zu übertragen.

Die Abbildung  $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$  ist gemäß (11.24) definiert durch  $F(\mathbf{x}) := \mathbf{T}\mathbf{x} + \mathbf{c}$  mit  $\mathbf{T} \in \mathbb{R}^{n,n}$ ,  $\mathbf{x}, \mathbf{c} \in \mathbb{R}^n$ .  $\|\cdot\|$  bezeichne eine Matrix- und eine Vektornorm, die miteinander kompatibel sind. Dann gilt für beliebige Vektoren  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$

$$\|F(\mathbf{x}) - F(\mathbf{y})\| = \|\mathbf{T}\mathbf{x} - \mathbf{T}\mathbf{y}\| = \|\mathbf{T}(\mathbf{x} - \mathbf{y})\| \leq \|\mathbf{T}\| \cdot \|\mathbf{x} - \mathbf{y}\|. \quad (11.25)$$

Die Matrixnorm  $\|\mathbf{T}\|$  übernimmt somit die Rolle der Lipschitz-Konstanten  $L$  in (4.8), und die Abbildung ist sicher dann kontrahierend, falls  $L := \|\mathbf{T}\| < 1$  gilt. Damit folgt aus dem Banachschen Fixpunktsatz:

**Satz 11.2.** Für eine Matrixnorm  $\|\mathbf{T}\|$  gelte  $\|\mathbf{T}\| < 1$ . Dann besitzt die Fixpunktgleichung  $\mathbf{x} = \mathbf{T}\mathbf{x} + \mathbf{c}$  genau einen Fixpunkt  $\mathbf{x} \in \mathbb{R}^n$ , gegen den die Iterationsfolge  $\mathbf{x}^{(k+1)} = \mathbf{T}\mathbf{x}^{(k)} + \mathbf{c}$ ,  $k = 0, 1, 2, \dots$ , für beliebige Startvektoren  $\mathbf{x}^{(0)} \in \mathbb{R}^n$  konvergiert. Außerdem gilt

$$\|\mathbf{x}^{(k+1)} - \mathbf{x}\| \leq \|\mathbf{T}\| \|\mathbf{x}^{(k)} - \mathbf{x}\|, \quad (11.26)$$

und

$$\|\mathbf{x}^{(k)} - \mathbf{x}\| \leq \frac{\|\mathbf{T}\|^k}{1 - \|\mathbf{T}\|} \|\mathbf{x}^{(1)} - \mathbf{x}^{(0)}\|. \quad (11.27)$$

An der hinreichenden Konvergenzaussage ist unbefriedigend, dass die Kontraktionsbedingung von der verwendeten Matrixnorm abhängig ist.

**Beispiel 11.2.** Für die symmetrische Matrix

$$\mathbf{T} = \begin{pmatrix} 0.1 & -0.4 \\ -0.4 & 0.8 \end{pmatrix}$$

ist die Zeilensummennorm (2.66)  $\|\mathbf{T}\|_z = \|\mathbf{T}\|_\infty = 1.2 > 1$ , während für die Spektralnorm (2.77)  $\|\mathbf{T}\|_e = \|\mathbf{T}\|_2 = \max_i |\lambda_i(\mathbf{T})| = 0.9815 < 1$  gilt. Die Kontraktionseigenschaft der durch  $\mathbf{T}$  definierten linearen Abbildung  $F : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  kann mit der Spektralnorm erkannt werden, mit der Zeilensummennorm hingegen nicht.  $\triangle$

Da der Spektralradius  $\sigma(\mathbf{T}) := \max_i |\lambda_i(\mathbf{T})|$  für jede Matrixnorm von  $\mathbf{T}$  eine untere Schranke bildet, ist er die entscheidende Größe für die Konvergenz der Fixpunktiteration. Mit ihm ist auch die notwendige Konvergenz-Bedingung zu formulieren.

**Satz 11.3.** Eine Fixpunktiteration  $\mathbf{x}^{(k+1)} = \mathbf{T}\mathbf{x}^{(k)} + \mathbf{c}$ ,  $k = 0, 1, 2, \dots$ , welche zum Gleichungssystem  $\mathbf{Ax} = \mathbf{b}$  vollständig konsistent ist, erzeugt genau dann für jeden beliebigen Startvektor  $\mathbf{x}^{(0)}$  eine gegen die Lösung  $\mathbf{x}$  konvergente Folge, falls  $\sigma(\mathbf{T}) < 1$  ist.

*Beweis.* Es soll hier genügen, den Satz für symmetrische Matrizen zu beweisen, um auf den Fall komplexer Eigenwerte und Eigenvektoren nicht eingehen zu müssen.

Nach Satz 5.3 kann eine symmetrische Matrix  $\mathbf{T}$  mit einer orthogonalen Matrix  $\mathbf{C}$  ähnlich auf eine Diagonalmatrix transformiert werden:

$$\mathbf{CTC}^T = \text{diag}(\lambda_i) =: \Lambda.$$

Wegen  $\mathbf{C}^T\mathbf{C} = \mathbf{I}$  ist (11.24) äquivalent zu der Iterationsvorschrift

$$\tilde{\mathbf{x}}^{(k+1)} = \Lambda \tilde{\mathbf{x}}^{(k)} + \tilde{\mathbf{c}} \quad (11.28)$$

mit  $\tilde{\mathbf{x}}^{(k)} = \mathbf{C}\mathbf{x}^{(k)}$  und  $\tilde{\mathbf{c}} = \mathbf{Cc}$ . Da  $\Lambda$  eine Diagonalmatrix ist, zerfällt (11.28) in  $n$  einzelne Iterationsvorschriften

$$\tilde{x}_i^{(k+1)} = \lambda_i \tilde{x}_i^{(k)} + \tilde{c}_i, \quad i = 1, \dots, n. \quad (11.29)$$

Da für den Spektralradius  $\sigma(\mathbf{T}) < 1$  gilt, gilt dies auch für jeden einzelnen Eigenwert. Daraus folgt die Konvergenz der Folgen (11.29), woraus die Konvergenz der Folge (11.28) folgt, und die ist äquivalent zur Konvergenz der Folge (11.24).

Aus der vorausgesetzten Regularität der Matrix  $\mathbf{A}$  und der vollständigen Konsistenz folgt damit die Existenz und Eindeutigkeit des Fixpunktes  $\mathbf{x}$  der Iteration. Die Voraussetzung  $\sigma(\mathbf{T}) < 1$  ist somit hinreichend für die Konvergenz der Iterationsfolge.

Die Notwendigkeit der Bedingung ergibt sich aus folgender Betrachtung. Aus den beiden Gleichungen  $\mathbf{x}^{(k+1)} = \mathbf{T}\mathbf{x}^{(k)} + \mathbf{c}$  und  $\mathbf{x} = \mathbf{T}\mathbf{x} + \mathbf{c}$  erhalten wir durch Subtraktion für den Fehler  $\mathbf{f}^{(k)} := \mathbf{x}^{(k)} - \mathbf{x}$  die Beziehung

$$\mathbf{f}^{(k+1)} = \mathbf{T}\mathbf{f}^{(k)}, \quad k = 0, 1, 2, \dots. \quad (11.30)$$

Aus der Annahme, es sei  $\sigma(\mathbf{T}) \geq 1$  folgt aber, dass ein Eigenwert  $\lambda_i$  von  $\mathbf{T}$  existiert mit  $|\lambda_i| \geq 1$ . Der zugehörige Eigenvektor sei  $\mathbf{y}_i$ . Für einen Startvektor  $\mathbf{x}^{(0)}$  mit  $\mathbf{f}^{(0)} = \mathbf{x}^{(0)} - \mathbf{x} = \alpha \mathbf{y}_i$  kann die Folge der Fehlervektoren  $\mathbf{f}^{(k)} = \mathbf{T}^k \mathbf{x}^{(0)} = \alpha \lambda_i^k \mathbf{y}_i$  nicht gegen den Nullvektor konvergieren.  $\square$

Die a priori und a posteriori Fehlerabschätzungen (4.11) und (4.12) des Banachschen Fixpunktsatzes sowie die dazu äquivalenten Abschätzungen (11.26) und (11.27) behalten ihre Gültigkeit, falls dort  $L$  bzw.  $\|\mathbf{T}\|$  durch  $\sigma(\mathbf{T})$  ersetzt und die einschlägige Vektornorm verwendet wird. Deshalb ist asymptotisch die Anzahl der Iterationsschritte, die notwendig sind, um die Norm des Fehlers auf den zehnten Teil zu reduzieren, gegeben durch

$$m \geq \frac{-1}{\log_{10} \|\mathbf{T}\|} \approx \frac{1}{-\log_{10} \sigma(\mathbf{T})}.$$

Man bezeichnet  $r(\mathbf{T}) := -\log_{10} \sigma(\mathbf{T})$  als *asymptotische Konvergenzrate* der Fixpunktiteration, weil ihr Wert den Bruchteil von Dezimalstellen angibt, welcher pro Schritt in der Näherung  $\mathbf{x}^{(k)}$  an Genauigkeit gewonnen wird. Die Konstruktion von linearen Iterationsverfahren muss zum Ziel haben, Iterationsmatrizen  $\mathbf{T}$  mit möglichst kleinem Spektralradius zu erzeugen, um damit eine rasche Konvergenz zu garantieren.

Die Iterationsmatrix  $\mathbf{T}$  wird in den anvisierten Anwendungen eine große, kompliziert aufgebaute Matrix sein. Deshalb ist es in der Regel gar nicht möglich, ihren Spektralradius zu berechnen. In einigen wichtigen Fällen kann aber aus bestimmten Eigenschaften der Matrix  $\mathbf{A}$  auf  $\sigma(\mathbf{T})$  geschlossen werden oder es kann  $\sigma(\mathbf{T})$  in Abhängigkeit von Eigenwerten anderer Matrizen dargestellt werden. Im Folgenden wird eine kleine Auswahl von solchen Aussagen zusammengestellt.

**Satz 11.4.** *Falls das Jacobi-Verfahren konvergent ist, dann trifft dies auch für das JOR-Verfahren für  $0 < \omega \leq 1$  zu.*

*Beweis.* Wir bezeichnen die Eigenwerte der Matrix  $\mathbf{T}_J = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$  des J-Verfahrens mit  $\mu_j$  und diejenigen von  $\mathbf{T}_{JOR}(\omega) = (1 - \omega)\mathbf{I} + \omega\mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$  mit  $\lambda_j$ . Da zwischen den Iterationsmatrizen der Zusammenhang  $\mathbf{T}_{JOR}(\omega) = (1 - \omega)\mathbf{I} + \omega\mathbf{T}_J$  besteht, so gilt für die Eigenwerte

$$\lambda_j = (1 - \omega) + \omega\mu_j, \quad j = 1, 2, \dots, n. \quad (11.31)$$

Wegen der Voraussetzung  $\sigma(\mathbf{T}_J) < 1$  liegen alle Eigenwerte  $\mu_j$  im Innern des Einheitskreises der komplexen Zahleebene. Die Eigenwerte  $\lambda_j$  stellen sich nach (11.31) für  $0 < \omega \leq 1$  als konvexe Linearkombination der Zahl 1 und des Wertes  $\mu_j$  dar, wobei das Gewicht von 1 echt kleiner als Eins ist. Jeder Eigenwert  $\lambda_j$  von  $\mathbf{T}_{JOR}(\omega)$  liegt somit auf der halboffenen Verbindungsgeraden von 1 nach  $\mu_j$  und somit ebenfalls im Inneren der Einheitskreises. Folglich ist  $\sigma(\mathbf{T}_{JOR}(\omega)) < 1$  für alle  $\omega \in (0, 1]$ .  $\square$

Differenzenverfahren führen in der Regel auf Gleichungssysteme mit *schwach diagonal dominanter* Matrix  $\mathbf{A}$ . Ein Konvergenz-Beweis für solche Matrizen und die betrachteten Verfahren gelingt nur unter einer Zusatzbedingung, von der bald zu sehen sein wird, dass sie für die betrachteten Anwendungen auch sinnvoll ist, siehe Beispiel 11.5.

**Definition 11.5.** Eine Matrix  $\mathbf{A} \in \mathbb{R}^{n,n}$  mit  $n > 1$  heißt *irreduzibel* oder *unzerlegbar*, falls für zwei beliebige, nichtleere und disjunkte Teilmengen  $S$  und  $T$  von  $W = \{1, 2, \dots, n\}$  mit  $S \cup T = W$  stets Indexwerte  $i \in S$  und  $j \in T$  existieren, so dass  $a_{ij} \neq 0$  ist.

Es ist leicht einzusehen, dass folgende Definition äquivalent ist.

**Definition 11.6.** Eine Matrix  $\mathbf{A} \in \mathbb{R}^{n,n}$  mit  $n > 1$  heißt *irreduzibel* oder *unzerlegbar*, falls es keine Permutationsmatrix  $\mathbf{P} \in \mathbb{R}^{n,n}$  gibt, so dass bei gleichzeitiger Zeilen- und Spaltenpermutation von  $\mathbf{A}$

$$\mathbf{P}^T \mathbf{A} \mathbf{P} = \begin{pmatrix} \mathbf{F} & \mathbf{0} \\ \mathbf{G} & \mathbf{H} \end{pmatrix} \quad (11.32)$$

wird, wo  $\mathbf{F}$  und  $\mathbf{H}$  quadratische Matrizen und  $\mathbf{0}$  eine Nullmatrix darstellen.

Diese Definition der Irreduzibilität einer Matrix bedeutet im Zusammenhang mit der Lösung eines Gleichungssystems  $\mathbf{Ax} = \mathbf{b}$ , dass sich die Gleichungen und gleichzeitig die Unbekannten nicht so umordnen lassen, dass das System derart zerfällt, dass zuerst ein Teilsystem mit der Matrix  $\mathbf{F}$  und anschließend ein zweites Teilsystem mit der Matrix  $\mathbf{H}$  gelöst werden kann.

Um die Unzerlegbarkeit einer gegebenen Matrix  $\mathbf{A}$  in einer konkreten Situation entscheiden zu können, ist die folgende äquivalente Definition nützlich [Hac 93, You 71].

**Definition 11.7.** Eine Matrix  $\mathbf{A} \in \mathbb{R}^{n,n}$  heißt *irreduzibel*, falls zu beliebigen Indexwerten  $i$  und  $j$  mit  $i, j \in W = \{1, 2, \dots, n\}$  entweder  $a_{ij} \neq 0$  ist oder eine Indexfolge  $i_1, i_2, \dots, i_s$  existiert, so dass

$$a_{ii_1} \cdot a_{i_1 i_2} \cdot a_{i_2 i_3} \cdots a_{i_s j} \neq 0$$

gilt.

Die in der Definition 11.7 gegebene Charakterisierung der Irreduzibilität besitzt eine anschauliche Interpretation mit Hilfe eines der Matrix  $\mathbf{A} \in \mathbb{R}^{n,n}$  zugeordneten *gerichteten Graphen*  $G(\mathbf{A})$ . Er besteht aus  $n$  verschiedenen *Knoten*, die von 1 bis  $n$  durchnummert seien. Zu jedem Indexpaar  $(i, j)$ , für welches  $a_{ij} \neq 0$  ist, existiert eine *gerichtete Kante* vom Knoten  $i$  zum Knoten  $j$ . Falls  $a_{ij} \neq 0$  und  $a_{ji} \neq 0$  sind, dann gibt es im Graphen  $G(\mathbf{A})$  je eine gerichtete Kante von  $i$  nach  $j$  und von  $j$  nach  $i$ . Für  $a_{ii} \neq 0$  enthält  $G(\mathbf{A})$  eine so genannte *Schleife*. Diese sind für die Irreduzibilität allerdings bedeutungslos. Eine Matrix  $\mathbf{A} \in \mathbb{R}^{n,n}$  ist genau dann irreduzibel, falls der Graph  $G(\mathbf{A})$  in dem Sinn zusammenhängend ist, dass von jedem Knoten  $i$  jeder (andere) Knoten  $j$  über mindestens einen *gerichteten Weg*, der sich aus gerichteten Kanten zusammensetzt, erreichbar ist.

**Beispiel 11.3.** Die Matrix  $\mathbf{A}$  der Abb. 11.1 ist irreduzibel, weil der zugeordnete Graph  $G(\mathbf{A})$  offensichtlich zusammenhängend ist.  $\triangle$

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

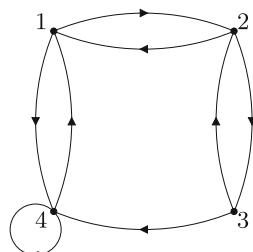


Abb. 11.1 Matrix  $\mathbf{A}$  und gerichteter Graph  $G(\mathbf{A})$ .

**Beispiel 11.4.** Die Matrix  $\mathbf{A}$  in Abb. 11.2 ist hingegen reduzibel, denn der Graph  $G(\mathbf{A})$  ist nicht zusammenhängend, da es keinen gerichteten Weg von 1 nach 4 gibt. In diesem Fall liefert eine gleichzeitige Vertauschung der zweiten und dritten Zeilen und Spalten eine Matrix der Gestalt (11.32). Mit  $S := \{1, 3\}$  und  $W := \{2, 4\}$  ist die Bedingung der Definition 11.5 nicht erfüllt.  $\triangle$

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{pmatrix}$$

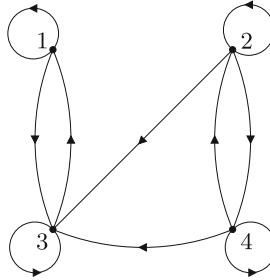


Abb. 11.2 Beispiel einer zerlegbaren Matrix.

**Beispiel 11.5.** Für die Diskretisierung einer partiellen Differenzialgleichung mit dem Differenzenverfahren und dem Differenzenstern aus Beispiel 11.1 ergibt sich eine vernünftige geometrische Bedingung für die Unzerlegbarkeit der entstehenden Matrix:

Jeder innere Punkt muss von jedem anderen inneren Punkt auf einem Weg über innere Nachbarpunkte erreichbar sein. Die Matrix zur Diskretisierung in Abb. 11.3 ist deshalb zerlegbar.  $\triangle$

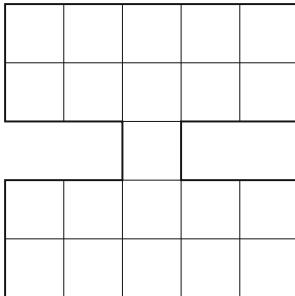


Abb. 11.3  
Diskretisierung eines Gebietes mit zerlegbarer Matrix.

**Lemma 11.8.** Eine irreduzible, schwach diagonal dominante Matrix  $\mathbf{A} \in \mathbb{C}^{n,n}$  hat nicht-verschwindende Diagonalelemente und ist regulär, d.h. es gilt  $|\mathbf{A}| \neq 0$ .

*Beweis.* Zuerst zeigen wir, dass  $a_{ii} \neq 0$  für alle  $i \in W = \{1, 2, \dots, n\}$  gilt. Angenommen, es sei für einen Index  $i$  das Diagonalelement  $a_{ii} = 0$ . Wegen der schwachen diagonalen Dominanz müsste dann  $a_{ij} = 0$  sein für alle  $j \neq i$ . Mit den Indexmengen  $S := \{i\}, T := W - \{i\}$  steht dies im Widerspruch zur vorausgesetzten Irreduzibilität nach Definition 11.5.

Die zweite Aussage wird ebenfalls indirekt gezeigt. Wir nehmen an, es sei  $|\mathbf{A}| = 0$ . Folglich besitzt das homogene Gleichungssystem  $\mathbf{A}\mathbf{z} = \mathbf{0}$  eine nichtriviale Lösung  $\mathbf{z} \neq \mathbf{0}$ . Wegen

$a_{ii} \neq 0$  können alle Gleichungen nach der in der Diagonale stehenden Unbekannten  $z_i$  aufgelöst werden, und wir erhalten

$$z_i = - \sum_{\substack{j=1 \\ j \neq i}}^n \frac{a_{ij}}{a_{ii}} z_j =: \sum_{j=1}^n b_{ij} z_j, \quad i = 1, 2, \dots, n, \quad (11.33)$$

mit  $b_{ii} := 0, b_{ij} := -a_{ij}/a_{ii}, (j \neq i)$ . Aus der schwachen diagonalen Dominanz von  $\mathbf{A}$  folgt aber für die Matrix  $\mathbf{B}$

$$\sum_{j=1}^n |b_{ij}| \leq 1 \quad \text{für } i = 1, 2, \dots, n, \quad (11.34)$$

wobei für mindestens einen Index  $i_0$  in (11.34) strikte Ungleichung gilt. Wir definieren  $M := \max_i |z_i| > 0$  und es sei  $k$  einer jener Indizes, für welchen  $|z_k| = M$  gilt. Aus (11.33) ergibt sich für die  $k$ -te Gleichung

$$M = |z_k| = \left| \sum_{j=1}^n b_{kj} z_j \right| \leq \sum_{j=1}^n |b_{kj}| \cdot |z_j|. \quad (11.35)$$

Wegen (11.34) gilt  $\sum_{j=1}^n |b_{kj}| \cdot M \leq M$  und zusammen mit (11.35) erhalten wir

$$\sum_{j=1}^n |b_{kj}|(|z_j| - M) \geq 0. \quad (11.36)$$

Da aber  $|z_j| \leq M$  ist für alle  $j$ , kann (11.36) nur dann erfüllt sein, falls für alle Matrixelemente  $b_{kj} \neq 0$  die Gleichheit  $|z_j| = M$  gilt. An dieser Stelle ist die Irreduzibilität von  $\mathbf{A}$  zu berücksichtigen. Nach Definition 11.7 existiert zu jedem Indexpaar  $(k, j)$  mit  $k \neq j$  entweder das Matrixelement  $a_{kj} \neq 0$  oder aber eine Indexfolge  $k_1, k_2, \dots, k_s$ , so dass  $a_{kk_1} \cdot a_{k_1 k_2} \cdot a_{k_2 k_3} \cdots a_{k_s j} \neq 0$  ist. Folglich ist entweder  $b_{kj} \neq 0$  oder  $b_{kk_1} \cdot b_{k_1 k_2} \cdot b_{k_2 k_3} \cdots b_{k_s j} \neq 0$ . Nach dem oben Gesagten muss somit entweder  $|z_j| = M$  oder  $|z_{k_1}| = M$  gelten. Im letzten Fall ist die Überlegung auch für die Gleichung mit dem Index  $k_1$  anwendbar, und wegen  $b_{k_1 k_2} \neq 0$  gilt dann auch  $|z_{k_2}| = M$ . Durch analoge Fortsetzung dieser Schlussweise ergibt sich somit, dass auch  $|z_j| = M$  für jedes beliebige  $j \neq k$  gelten muss. Für diejenige Gleichung mit dem Index  $i_0$ , für welche (11.34) als strikte Ungleichung gilt, folgt deshalb wegen  $|z_j| = M$

$$M \leq \sum_{j=1}^n |b_{i_0 j}| \cdot M < M$$

der gewünschte Widerspruch. Also muss die Matrix  $\mathbf{A}$  regulär sein.  $\square$

**Satz 11.9.** Für eine irreduzible, schwach diagonal dominante Matrix  $\mathbf{A}$  ist das J-Verfahren konvergent und somit auch das JOR-Verfahren für  $\omega \in (0, 1]$ .

*Beweis.* Wir zeigen die Aussage auf indirekte Art und treffen die Gegenannahme, es gelte  $\sigma(\mathbf{T}_J) \geq 1$ . Demnach existiert ein Eigenwert  $\mu$  von  $\mathbf{T}_J$  mit  $|\mu| \geq 1$ , und für ihn gelten

$$|\mathbf{T}_J - \mu\mathbf{I}| = 0 \quad \text{oder} \quad |\mathbf{I} - \mu^{-1}\mathbf{T}_J| = 0.$$

Aus der Voraussetzung, die Matrix  $\mathbf{A}$  sei irreduzibel, folgt, dass auch die Iterationsmatrix des J-Verfahrens  $\mathbf{T}_J = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$  irreduzibel ist, da diese Eigenschaft nur die Nicht-Diagonalelemente betrifft. Das gleiche gilt dann auch für die Matrix  $\mathbf{C} := \mathbf{I} - \mu^{-1}\mathbf{T}_J$ , welche zudem schwach diagonal dominant ist. Denn für die Matrixelemente  $t_{ij}$  von  $\mathbf{T}_J$  gelten  $t_{ii} = 0$ ,  $t_{ij} = -a_{ij}/a_{ii}$ , ( $j \neq i$ ), und infolge der schwachen diagonalen Dominanz von  $\mathbf{A}$  folgt somit  $\sum_{j \neq i} |t_{ij}| \leq 1$  für alle  $i$ , wobei für mindestens einen Index  $i_0$  strikte Ungleichheit

gilt. Weiter ist zu beachten, dass  $|\mu^{-1}| \leq 1$  ist, und zusammen mit der vorerwähnten Eigenschaft folgt die schwache diagonale Dominanz von  $\mathbf{C}$ . Nach Lemma 11.8 muss aber dann  $|\mathbf{C}| = |\mathbf{I} - \mu^{-1}\mathbf{T}_J| \neq 0$  sein, was den Widerspruch liefert. Unsere Gegenannahme ist falsch, und es gilt  $\sigma(\mathbf{T}_J) < 1$ .  $\square$

Im Folgenden betrachten wir den Spezialfall, dass die Matrix  $\mathbf{A}$  des linearen Gleichungssystems symmetrisch und positiv definit ist.

**Satz 11.10.** Es sei  $\mathbf{A} \in \mathbb{R}^{n,n}$  symmetrisch und positiv definit und überdies das J-Verfahren konvergent. Dann ist das JOR-Verfahren konvergent für alle  $\omega$  mit

$$0 < \omega < 2/(1 - \mu_{\min}) \leq 2, \tag{11.37}$$

wobei  $\mu_{\min}$  der kleinste, negative Eigenwert von  $\mathbf{T}_J$  ist.

*Beweis.* Aus der Symmetrie von  $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}$  folgt  $\mathbf{U} = \mathbf{L}^T$  und folglich ist  $(\mathbf{U} + \mathbf{L})$  symmetrisch. Wegen der positiven Definitheit von  $\mathbf{A}$  sind die Diagonalelemente  $a_{ii} > 0$ , und es kann die reelle, reguläre Diagonalmatrix  $\mathbf{D}^{1/2} := \text{diag}(\sqrt{a_{11}}, \sqrt{a_{22}}, \dots, \sqrt{a_{nn}})$  gebildet werden. Dann ist die Iterationsmatrix  $\mathbf{T}_J = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$  ähnlich zur symmetrischen Matrix

$$\mathbf{S} := \mathbf{D}^{1/2}\mathbf{T}_J\mathbf{D}^{-1/2} = \mathbf{D}^{-1/2}(\mathbf{L} + \mathbf{U})\mathbf{D}^{-1/2}.$$

Demzufolge sind die Eigenwerte  $\mu_j$  von  $\mathbf{T}_J$  reell. Unter ihnen muss mindestens einer negativ sein. Denn die Diagonalelemente von  $\mathbf{T}_J$  sind gleich null und somit ist auch die Spur von  $\mathbf{T}_J$  gleich null, die aber gleich der Summe der Eigenwerte ist. Somit gilt, falls  $\mathbf{T}_J \neq \mathbf{0}$  ist,  $\mu_{\min} = \min \mu_j < 0$ . Wegen der vorausgesetzten Konvergenz des J-Verfahrens ist  $\sigma(\mathbf{T}_J) < 1$  und deshalb  $\mu_{\min} > -1$ . Wegen der Relation (11.31) zwischen den Eigenwerten  $\lambda_j$  von  $\mathbf{T}_{\text{JOR}}(\omega)$  und den Eigenwerten  $\mu_j$  sind auch die  $\lambda_j$  reell. Die notwendige und hinreichende Bedingung für die Konvergenz des JOR-Verfahrens lautet deshalb

$$-1 < 1 - \omega + \omega\mu_j < 1, \quad j = 1, 2, \dots, n,$$

oder nach Subtraktion von 1 und anschließender Multiplikation mit  $-1$

$$0 < \omega(1 - \mu_j) < 2, \quad j = 1, 2, \dots, n.$$

Da  $1 - \mu_j > 0$  gilt, ist  $1 - \mu_j$  für  $\mu_j = \mu_{\min} < 0$  am größten, und es folgt daraus die Bedingung (11.37).  $\square$

Das Gesamtschrittverfahren braucht nicht für jede symmetrische und positiv definite Matrix  $\mathbf{A}$  zu konvergieren.

**Beispiel 11.6.** Die symmetrische Matrix

$$\mathbf{A} = \begin{pmatrix} 1 & a & a \\ a & 1 & a \\ a & a & 1 \end{pmatrix} = \mathbf{I} - \mathbf{L} - \mathbf{U}$$

ist positiv definit für  $a \in (-0.5, 1)$ , wie man mit Hilfe ihrer Cholesky-Zerlegung bestimmen kann. Die zugehörige Iterationsmatrix

$$\mathbf{T}_J = \mathbf{L} + \mathbf{U} = \begin{pmatrix} 0 & -a & -a \\ -a & 0 & -a \\ -a & -a & 0 \end{pmatrix}$$

hat die Eigenwerte  $\mu_{1,2} = a$  und  $\mu_3 = -2a$ , so dass  $\sigma(\mathbf{T}_J) = 2|a|$  ist. Das J-Verfahren ist dann und nur dann konvergent, falls  $a \in (-0.5, 0.5)$  gilt. Es ist nicht konvergent für  $a \in [0.5, 1)$ , für welche Werte  $\mathbf{A}$  positiv definit ist.  $\triangle$

Die bisherigen Sätze beinhalten nur die grundsätzliche Konvergenz des JOR-Verfahrens unter bestimmten Voraussetzungen an die Systemmatrix  $\mathbf{A}$ , enthalten aber keine Hinweise über die optimale Wahl von  $\omega$  für bestmögliche Konvergenz. Die diesbezügliche Optimierungsaufgabe lautet wegen (11.31):

$$\min_{\omega} \sigma(\mathbf{T}_{\text{JOR}}(\omega)) = \min_{\omega} \left\{ \max_j |\lambda_j| \right\} = \min_{\omega} \left\{ \max_j |1 - \omega + \omega \mu_j| \right\}$$

Diese Aufgabe kann dann gelöst werden, wenn über die Lage oder Verteilung der Eigenwerte  $\mu_j$  von  $\mathbf{T}_J$  konkrete Angaben vorliegen. Wir führen diese Diskussion im Spezialfall einer symmetrischen und positiv definiten Matrix  $\mathbf{A}$ , für die die Eigenwerte  $\mu_j$  von  $\mathbf{T}_J$  reell sind, und unter der Voraussetzung, dass das J-Verfahren konvergent sei, und somit  $-1 < \mu_{\min} \leq \mu_j \leq \mu_{\max} < 1$  gilt. Wegen (11.31) ist  $|\lambda_j| = |1 - \omega(1 - \mu_j)|$  eine stückweise lineare Funktion von  $\omega$ . In Abb. 11.4 sind die Geraden  $|\lambda_j|$  für  $\mu_{\min} = -0.6, \mu_{\max} = 0.85$  und einen weiteren Eigenwert  $\mu_j$  dargestellt. Der Wert von  $\omega_{\text{opt}}$  wird aus dem Schnittpunkt der beiden erstgenannten Geraden ermittelt, da sie den Spektralradius  $\sigma(\mathbf{T}_{\text{JOR}}(\omega))$  bestimmen. Aus der Gleichung

$$1 - \omega(1 - \mu_{\max}) = -1 + \omega(1 - \mu_{\min})$$

ergibt sich in diesem Fall

$$\omega_{\text{opt}} = 2/(2 - \mu_{\max} - \mu_{\min}). \tag{11.38}$$

Das führt zu einem Spektralradius

$$\sigma(\mathbf{T}_{\text{JOR}}(\omega_{\text{opt}})) = 0.8286 < 0.85 = \sigma(\mathbf{T}_J)$$

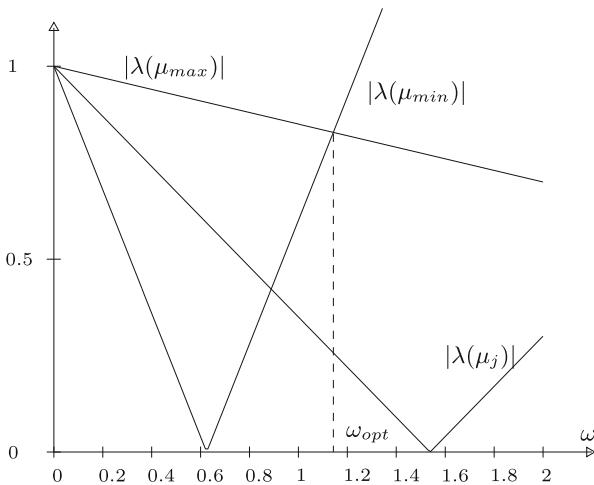


Abb. 11.4 Zur optimalen Wahl von  $\omega$ , JOR-Verfahren.

Diese Abnahme des Spektralradius  $\sigma(\mathbf{T}_J)$  zu  $\sigma(\mathbf{T}_{JOR}(\omega_{opt}))$  ist im betrachteten Fall minimal. Abb. 11.4 zeigt aber, dass mit einer Wahl von  $\omega > \omega_{opt}$  die Konvergenz stark verschlechtert wird. Für  $\mu_{\min} = -\mu_{\max}$  ist  $\omega_{opt} = 1$ , d.h. dann konvergiert das Gesamtschrittverfahren am schnellsten.

Bei anderen, speziellen Eigenwertverteilungen  $\mu_j$  kann durch geeignete Wahl von  $\omega$  eine beträchtliche Konvergenzverbesserung erzielt werden.

Im Folgenden untersuchen wir die Konvergenz des SOR-Verfahrens und behandeln das Einzelschrittverfahren als Spezialfall für  $\omega = 1$ .

**Satz 11.11.** Das SOR-Verfahren ist für  $0 < \omega \leq 1$  konvergent, falls die Matrix  $\mathbf{A} \in \mathbb{R}^{n,n}$  entweder strikt diagonal dominant oder irreduzibel und schwach diagonal dominant ist.

*Beweis.* Aus der Voraussetzung für  $\mathbf{A}$  folgt  $a_{ii} \neq 0$ , so dass die Matrix  $\mathbf{D}$  (11.14) regulär ist. Der Beweis des Satzes wird indirekt geführt. Wir nehmen an, es sei  $\sigma(\mathbf{T}_{SOR}(\omega)) \geq 1$  für  $0 < \omega \leq 1$ . Die Iterationsmatrix  $\mathbf{T}_{SOR}(\omega) = (\mathbf{D} - \omega\mathbf{L})^{-1}[(1 - \omega)\mathbf{D} + \omega\mathbf{U}]$  besitzt dann einen Eigenwert  $\lambda$  mit  $|\lambda| \geq 1$ . Für diesen Eigenwert gilt  $|\mathbf{T}_{SOR}(\omega) - \lambda\mathbf{I}| = 0$ . Für diese Determinante erhalten wir durch eine Reihe von Umformungen nacheinander

$$\begin{aligned} 0 &= |\mathbf{T}_{SOR}(\omega) - \lambda\mathbf{I}| = |(\mathbf{D} - \omega\mathbf{L})^{-1}[(1 - \omega)\mathbf{D} + \omega\mathbf{U}] - \lambda\mathbf{I}| \\ &= |(\mathbf{D} - \omega\mathbf{L})^{-1}\{(1 - \omega)\mathbf{D} + \omega\mathbf{U} - \lambda(\mathbf{D} - \omega\mathbf{L})\}| \\ &= |(\mathbf{D} - \omega\mathbf{L})^{-1}| \cdot |(1 - \omega - \lambda)\mathbf{D} + \omega\mathbf{U} + \lambda\omega\mathbf{L}| \\ &= |\mathbf{D} - \omega\mathbf{L}|^{-1}(1 - \omega - \lambda)^n \left| \mathbf{D} - \frac{\omega}{\lambda + \omega - 1}\mathbf{U} - \frac{\lambda\omega}{\lambda + \omega - 1}\mathbf{L} \right|. \end{aligned}$$

Bei der letzten Umformung wurde verwendet, dass der Faktor  $(1 - \omega - \lambda) \neq 0$  ist wegen

$0 < \omega \leq 1$  und  $|\lambda| \geq 1$ . Deswegen und weil  $|\mathbf{D} - \omega \mathbf{L}| \neq 0$  ist, gilt auf Grund unserer Annahme

$$\left| \mathbf{D} - \frac{\omega}{\lambda + \omega - 1} \mathbf{U} - \frac{\lambda \omega}{\lambda + \omega - 1} \mathbf{L} \right| = 0. \quad (11.39)$$

Der betrachtete Eigenwert  $\lambda$  kann komplex sein. Seinen Kehrwert setzen wir deshalb in der Form  $\lambda^{-1} = r \cdot e^{i\vartheta}$  an, und es gilt  $r \leq 1$ . Wir wollen nun zeigen, dass die Faktoren von  $\mathbf{U}$  und  $\mathbf{L}$  in (11.39) betragmäßig kleiner oder gleich Eins sind.

$$\begin{aligned} \left| \frac{\lambda \omega}{\lambda + \omega - 1} \right| &= \left| \frac{\omega}{1 + (\omega - 1)\lambda^{-1}} \right| = \left| \frac{\omega}{1 - (1 - \omega)r e^{i\vartheta}} \right| \\ &= \frac{\omega}{[\{1 - (1 - \omega)r \cos \vartheta\}^2 + (1 - \omega)^2 r^2 \sin^2 \vartheta]^{1/2}} \\ &= \frac{\omega}{[1 - 2(1 - \omega)r \cos \vartheta + (1 - \omega)^2 r^2]^{1/2}} \leq \frac{\omega}{1 - r(1 - \omega)} \end{aligned}$$

Der letzte Quotient ist aber durch Eins beschränkt für  $0 < \omega \leq 1$  und  $r \leq 1$ , denn es ist

$$1 - \frac{\omega}{1 - r(1 - \omega)} = \frac{(1 - r)(1 - \omega)}{1 - r(1 - \omega)} \geq 0.$$

Deshalb folgen in der Tat die Abschätzungen

$$\left| \frac{\omega}{\lambda + \omega - 1} \right| \leq \left| \frac{\omega \lambda}{\lambda + \omega - 1} \right| \leq 1.$$

Die Matrix  $\mathbf{A}$  ist nach Voraussetzung diagonal dominant oder irreduzibel und schwach diagonal dominant. Dasselbe gilt auch für die im Allgemeinen komplexwertige Matrix der Determinante (11.39). Nach dem Lemma 11.8 ist die Determinante einer solchen Matrix aber von null verschieden, und dies liefert den Widerspruch. Die Annahme  $\sigma(\mathbf{T}_{\text{SOR}}(\omega)) \geq 1$  für  $0 < \omega \leq 1$  ist falsch, und damit ist die Aussage des Satzes bewiesen.  $\square$

**Satz 11.12.** *Das SOR-Verfahren ist höchstens für  $0 < \omega < 2$  konvergent.*

*Beweis.* Zum Beweis verwenden wir die Tatsache, dass das Produkt der  $n$  Eigenwerte einer  $(n \times n)$ -Matrix gleich der Determinante der Matrix ist. Für die Iterationsmatrix  $\mathbf{T}_{\text{SOR}}(\omega)$  gilt aber unter Beachtung der Dreiecksgestalt von Matrizen

$$\begin{aligned} |\mathbf{T}_{\text{SOR}}(\omega)| &= |(\mathbf{D} - \omega \mathbf{L})^{-1}[(1 - \omega)\mathbf{D} + \omega \mathbf{U}]| \\ &= |\mathbf{D} - \omega \mathbf{L}|^{-1} \cdot |(1 - \omega)\mathbf{D} + \omega \mathbf{U}| \\ &= |\mathbf{D}|^{-1} (1 - \omega)^n |\mathbf{D}| = (1 - \omega)^n. \end{aligned}$$

Daraus folgt die Ungleichung

$$\sigma(\mathbf{T}_{\text{SOR}}(\omega))^n \geq \prod_{i=1}^n |\lambda_i| = |1 - \omega|^n,$$

und somit  $\sigma(\mathbf{T}_{\text{SOR}}(\omega)) \geq |1 - \omega|$ . Damit kann  $\sigma(\mathbf{T}_{\text{SOR}}(\omega)) < 1$  höchstens dann gelten, falls  $\omega \in (0, 2)$ .  $\square$

Es gibt Fälle von Matrizen  $\mathbf{A}$ , bei denen für gewisse  $\omega$ -Werte  $\sigma(\mathbf{T}_{\text{SOR}}(\omega)) = |1 - \omega|$  ist. Dass andererseits das mögliche Intervall für  $\omega$  ausgeschöpft werden kann, zeigt der folgende

**Satz 11.13.** *Für eine symmetrische und positiv definite Matrix  $\mathbf{A} \in \mathbb{R}^{n,n}$  gilt*

$$\sigma(\mathbf{T}_{\text{SOR}}(\omega)) < 1 \quad \text{für } \omega \in (0, 2). \quad (11.40)$$

*Beweis.* Wir wollen zeigen, dass jeder Eigenwert  $\lambda \in \mathbb{C}$  von  $\mathbf{T}_{\text{SOR}}(\omega)$  für  $\omega \in (0, 2)$  betragsmäßig kleiner als Eins ist. Sei also  $\mathbf{z} \in \mathbb{C}^n$  ein zu  $\lambda$  gehöriger Eigenvektor, so dass gilt

$$\mathbf{T}_{\text{SOR}}(\omega)\mathbf{z} = \lambda\mathbf{z}.$$

Dann gelten auch die beiden folgenden äquivalenten Gleichungen

$$\begin{aligned} (\mathbf{D} - \omega\mathbf{L})^{-1}[(1 - \omega)\mathbf{D} + \omega\mathbf{U}]\mathbf{z} &= \lambda\mathbf{z}, \\ 2[(1 - \omega)\mathbf{D} + \omega\mathbf{U}]\mathbf{z} &= \lambda 2(\mathbf{D} - \omega\mathbf{L})\mathbf{z}. \end{aligned} \quad (11.41)$$

Für die beiden Matrizen in (11.41) sind Darstellungen zu verwenden, in denen insbesondere  $\mathbf{A}$  und  $\mathbf{D}$  auftreten, die symmetrisch und positiv definit sind. Wegen  $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}$  gelten

$$\begin{aligned} 2[(1 - \omega)\mathbf{D} + \omega\mathbf{U}] &= (2 - \omega)\mathbf{D} - \omega\mathbf{D} + 2\omega\mathbf{U} \\ &= (2 - \omega)\mathbf{D} - \omega\mathbf{A} - \omega\mathbf{L} - \omega\mathbf{U} + 2\omega\mathbf{U} = (2 - \omega)\mathbf{D} - \omega\mathbf{A} + \omega(\mathbf{U} - \mathbf{L}), \\ 2(\mathbf{D} - \omega\mathbf{L}) &= (2 - \omega)\mathbf{D} + \omega\mathbf{D} - 2\omega\mathbf{L} \\ &= (2 - \omega)\mathbf{D} + \omega\mathbf{A} + \omega\mathbf{L} + \omega\mathbf{U} - 2\omega\mathbf{L} = (2 - \omega)\mathbf{D} + \omega\mathbf{A} + \omega(\mathbf{U} - \mathbf{L}). \end{aligned}$$

Setzen wir die beiden Ausdrücke, die sich nur im Vorzeichen des Summanden  $\omega\mathbf{A}$  unterscheiden, in (11.41) ein und multiplizieren die Vektorgleichung von links mit  $\mathbf{z}^H = \bar{\mathbf{z}}^T$ , erhalten wir unter Beachtung der Distributivität des Skalarproduktes für komplexe Vektoren und der Tatsache, dass  $\omega$  und  $(2 - \omega)$  reell sind

$$\begin{aligned} (2 - \omega)\mathbf{z}^H\mathbf{D}\mathbf{z} - \omega\mathbf{z}^H\mathbf{A}\mathbf{z} + \omega\mathbf{z}^H(\mathbf{U} - \mathbf{U}^T)\mathbf{z} \\ = \lambda[(2 - \omega)\mathbf{z}^H\mathbf{D}\mathbf{z} + \omega\mathbf{z}^H\mathbf{A}\mathbf{z} + \omega\mathbf{z}^H(\mathbf{U} - \mathbf{U}^T)\mathbf{z}]. \end{aligned} \quad (11.42)$$

Da  $\mathbf{A}$  symmetrisch und positiv definit ist, ist für jeden komplexen Vektor  $\mathbf{z} \neq \mathbf{0}$  der Wert  $\mathbf{z}^H\mathbf{A}\mathbf{z} = a$  eine reelle positive Zahl. Dasselbe gilt auch für  $\mathbf{z}^H\mathbf{D}\mathbf{z} = d$ . Die Matrix  $(\mathbf{U} - \mathbf{U}^T)$  ist hingegen schiefsymmetrisch, so dass die quadratische Form  $\mathbf{z}^H(\mathbf{U} - \mathbf{U}^T)\mathbf{z} = ib$ ,  $b \in \mathbb{R}$ , einen rein imaginären Wert annimmt. Aus der skalaren Gleichung (11.42) folgt damit

$$\lambda = \frac{(2 - \omega)d - \omega a + i\omega b}{(2 - \omega)d + \omega a + i\omega b}.$$

Für  $\omega \in (0, 2)$  sind  $(2 - \omega)d > 0$  und  $\omega a > 0$  und somit  $|(2 - \omega)d - \omega a| < (2 - \omega)d + \omega a$ . Jeder Eigenwert  $\lambda$  von  $\mathbf{T}_{\text{SOR}}(\omega)$  ist darstellbar als Quotient von zwei komplexen Zahlen mit gleichem Imaginärteil, wobei der Zähler einen betragskleineren Realteil als der Nenner aufweist. Daraus folgt die Behauptung (11.40).  $\square$

Für die zahlreichen weiteren Konvergenzsätze, Varianten und Verallgemeinerungen der hier behandelten Iterationsverfahren sei auf die weiterführende Spezialliteratur [Hac 93, You 71] verwiesen.

### 11.2.3 Optimaler Relaxationsparameter und Konvergenzgeschwindigkeit

Die Sätze 11.11 und 11.13 garantieren die Konvergenz der Überrelaxationsmethode für bestimmte  $\omega$ -Intervalle, lassen aber die Frage einer optimalen Wahl des Relaxationsparameters  $\omega$  zur Minimierung des Spektralradius  $\sigma(\mathbf{T}_{\text{SOR}}(\omega))$  offen. Für eine Klasse von Matrizen  $\mathbf{A}$  mit spezieller Struktur, wie sie bei Differenzenverfahren für elliptische Randwertaufgaben auftreten, existiert eine entsprechende Aussage, siehe [Sch 08]. Da die Konvergenz der Relaxationsverfahren selbst für optimale Werte von  $\omega$  unbefriedigend ist, wollen wir in diesem Abschnitt die Abhängigkeit der Konvergenz von  $\omega$  nur am Beispiel betrachten und dabei auch die Abhängigkeit der Konvergenz vom Diskretisierungs-Parameter  $h$  untersuchen.

**Beispiel 11.7.** Um die Konvergenzverbesserung des SOR-Verfahrens gegenüber dem Gesamtschritt- und Einzelschrittverfahren zu illustrieren, und um die Abhängigkeit der Relaxationsverfahren vom Diskretisierungsparameter  $h$  zu bestimmen, betrachten wir das Modellproblem der elliptischen Randwertaufgabe im Einheitsquadrat entsprechend zum Beispiel 11.1 mit einer rechten Seite, die die Angabe der exakten Lösung erlaubt:

$$\begin{aligned} -\Delta u &= f \quad \text{in } G = (0, 1)^2 \\ u &= 0 \quad \text{auf } \Gamma \end{aligned} \tag{11.43}$$

mit der rechten Seite

$$f = 2[(1 - 6x^2)y^2(1 - y^2) + (1 - 6y^2)x^2(1 - x^2)]. \tag{11.44}$$

Das Problem hat dann die exakte Lösung

$$u(x, y) = (x^2 - x^4)(y^4 - y^2). \tag{11.45}$$

Für dieses Modellproblem kann der Spektralradius  $\mu_1 = \sigma(\mathbf{T}_J)$  des J-Verfahrens angegeben werden. Ist er kleiner als eins, und werden außerdem die inneren Gitterpunkte schachbrettartig eingefärbt und anschließend so nummeriert, dass zuerst die Gitterpunkte der einen Farbe und dann diejenigen der anderen Farbe erfasst werden (siehe Abb. 11.5), dann lässt sich der optimale Relaxationsparameter des SOR-Verfahrens berechnen als

$$\omega_{\text{opt}} = 2/(1 + \sqrt{1 - \mu_1^2}). \tag{11.46}$$

Sei  $N$  die Anzahl der inneren Gitterpunkte pro Zeile und Spalte bei homogener Diskretisierung mit der Gitterweite  $h = 1/(N + 1)$ . Dann besitzt die Matrix  $\mathbf{A}$  der Ordnung  $n = N^2$  für die Fünfpunkte-Formel (11.1) offenbar die spezielle Blockstruktur

$$\mathbf{A} = \begin{pmatrix} \mathbf{D}_1 & \mathbf{H} \\ \mathbf{K} & \mathbf{D}_2 \end{pmatrix} \quad \text{mit Diagonalmatrizen } \mathbf{D}_1 \text{ und } \mathbf{D}_2,$$

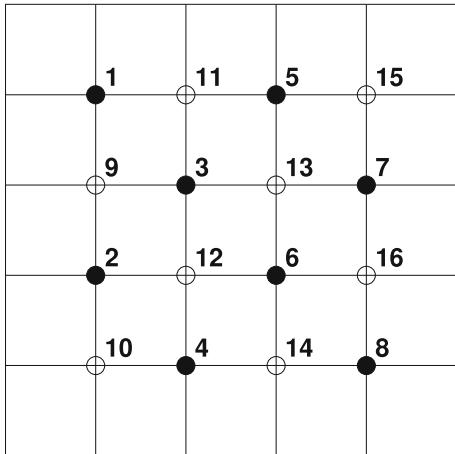


Abb. 11.5  
Schachbrett-Nummerierung der Gitterpunkte.

weil in der Differenzengleichung für einen schwarzen Gitterpunkt neben der Unbekannten des betreffenden Punktes nur Unbekannte von weiß markierten Gitterpunkten auftreten und umgekehrt. Dies kann man am ( $N = 4$ )-Beispiel gut erkennen:

$$A = \frac{1}{h^2} \begin{pmatrix} + & \cdot & \times & \cdot & \times & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \times & \times & \cdot & \times & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \times & \cdot & \times & \times & \times & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \times & \cdot & \times & \cdot & \times & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \times & \cdot & \times & \cdot & \times & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \times & \times & \times & \cdot & \times \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \times & \cdot & \times & \times \\ \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \times & \cdot & \times \\ \times & \times & \times & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot \\ \cdot & \times & \cdot & \times & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \times & \cdot & \times & \cdot & \times & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \times & \times & \times & \cdot & \times & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \times & \cdot & \times & \times & \times & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \times & \cdot & \times & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \times & \times & \times & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \times & \times & \times & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot \end{pmatrix}$$

Dabei sind  $+ \equiv 4$ ,  $\times \equiv -1$  und  $\cdot \equiv 0$ .

Die Matrix  $A$  geht aus (11.3) durch eine gleichzeitige Zeilen- und Spaltenpermutation hervor. Die Matrix  $A$  ist irreduzibel, schwach diagonal dominant und symmetrisch wegen  $H^T = K$ , die Iterationsmatrix  $T_J$  hat reelle Eigenwerte  $\mu_j$ , und wegen Satz 11.9 gilt für sie  $\sigma(T_J) < 1$ . Deshalb ist der optimale Wert des Relaxationsparameters der SOR-Methode nach (11.46) berechenbar. Für das Modellproblem (11.43) sind  $D_1$  und  $D_2$  je gleich den Vierfachen entsprechender Einheitsmatrizen. Aus diesem Grund gilt für die Iterationsmatrix

$$T_J = \frac{1}{4} \begin{pmatrix} 0 & -H \\ -K & 0 \end{pmatrix} = I - \frac{1}{4} A.$$

Tab. 11.1 Konvergenzverhalten für das Modellproblem.

$N$	$n$	$\mu_{11}$	$m_J$	$\sigma(\mathbf{T}_{\text{ES}})$	$m_{\text{ES}}$	$m_{\text{ESP}}$	Zeit	$\omega_{\text{opt}}$	$\sigma(\mathbf{T}_{\text{SOR}})$	$m_{\text{SOR}}$	$q$
4	16	0.8090	11	0.6545	5	7		1.2596	0.2596	1.7	6
8	64	0.9397	37	0.8830	19	19		1.4903	0.4903	3.2	11
16	256	0.9830	134	0.9662	67	62		1.6895	0.6895	6.2	22
32	1024	0.9955	507	0.9910	254	232	1	1.8264	0.8264	12	42
64	4096	0.9988	1971	0.9977	985	895	47	1.9078	0.9078	24	83
128	16384	0.9997	7764	0.9994	3882	3520	3422	1.9525	0.9525	47	164

Die Eigenwerte der Matrix  $\mathbf{A}$  lassen sich formelmäßig angeben:

$$\lambda_{jk} = 4 - 2 \left\{ \cos \left( \frac{j\pi}{N+1} \right) + \cos \left( \frac{k\pi}{N+1} \right) \right\}, \quad j, k = 1, 2, \dots, N. \quad (11.47)$$

Aus den Eigenwerten (11.47) der Matrix  $\mathbf{A}$  ergeben sich die Eigenwerte von  $\mathbf{T}_J$  als

$$\mu_{jk} = \frac{1}{2} \left\{ \cos \left( \frac{j\pi}{N+1} \right) + \cos \left( \frac{k\pi}{N+1} \right) \right\}, \quad j, k = 1, 2, \dots, N.$$

Daraus resultiert der Spektralradius für  $j = k = 1$

$$\sigma(\mathbf{T}_J) = \mu_{11} = \cos \left( \frac{\pi}{N+1} \right) = \cos(\pi h).$$

Hieraus folgt, dass das Gesamtschrittverfahren linear konvergiert mit einer asymptotischen Konvergenzrate der Größenordnung  $O(1-h^2)$ ; das ist für kleine  $h$  sehr langsam. Besonders katastrophal ist aber der daraus folgende Umstand, dass mit größer werdender Matrixordnung die Anzahl der Iterationsschritte, die zum Erreichen einer gewissen Genauigkeit notwendig sind, linear mit der Ordnung  $n$  der Matrix wächst. Dies ist in Tabelle 11.1 in den Spalten für  $n$  und  $m_J$  gut abzulesen. Dort sind für einige Werte  $N$  die Ordnungen  $n = N^2$ , die Spektralradien  $\mu_{11} = \sigma(\mathbf{T}_J)$  des J-Verfahrens,  $\sigma(\mathbf{T}_{\text{ES}})$  des Einzelschrittverfahrens, die optimalen Werte  $\omega_{\text{opt}}$  des SOR-Verfahrens und die zugehörigen Spektralradien  $\sigma(\mathbf{T}_{\text{SOR}}(\omega))$  angegeben. Zu Vergleichszwecken sind die ganzzahligen Werte  $m$  von Iterationsschritten aufgeführt, welche zur Reduktion des Fehlers auf den zehnten Teil nötig sind sowie das Verhältnis  $q = m_J/m_{\text{SOR}}$ , welche die wesentliche Konvergenzsteigerung zeigen. Die Werte sind theoretische Werte, die aus den Spektralradien und ihren Beziehungen untereinander und zum optimalen Relaxationsparameter des SOR-Verfahrens berechnet wurden. Bei der praktischen Rechnung hängen sie noch von anderen Parametern wie der Struktur der Startnäherung und der rechten Seite ab, wenn auch nur geringfügig. Deshalb wurden in Tabelle 11.1 neben den theoretischen Werten  $m_{\text{ES}}$  die entsprechenden Werte  $m_{\text{ESP}}$  aufgeführt, die sich bei der Lösung des konkreten Beispiels (11.43) ergaben, und für  $N \geq 32$  die relativen Rechenzeiten.

Die Tabelle zeigt, dass das SOR-Verfahren etwa  $N$  mal schneller konvergiert als das Gesamtschrittverfahren, falls  $\omega$  optimal gewählt wird. Diese Tatsache kann für das Modellproblem auch auf analytischem Weg nachgewiesen werden. Sie besagt, dass dann die lineare Konvergenz die bessere Rate  $O(1-h)$  besitzt, die allerdings immer noch eine schlechte und bei größer werdender Koeffizientenmatrix mit  $N = \sqrt{n}$  schlechter werdende Konvergenz widerspiegelt. Außerdem ist der optimale Relaxationsparameter in den meisten Anwendungen nicht berechenbar.  $\triangle$

### 11.3 Mehrgittermethoden

Die Effizienz und asymptotische Optimalität (siehe Seite 489) der Mehrgittermethoden soll hauptsächlich an zwei Modellproblemen demonstriert werden. Das eine ist die Poisson-Gleichung im Einheitsquadrat, siehe Beispiel 11.7, für das andere gehen wir auf eindimensionale Randwertprobleme zurück, an denen sich die Eigenschaften noch besser verstehen lassen; außerdem sind die meisten Beobachtungen auf zwei- und dreidimensionale Probleme übertragbar.

#### 11.3.1 Ein eindimensionales Modellproblem

Die Randwertaufgabe (9.64) soll in vereinfachter und leicht abgeänderter Form

$$\begin{aligned} -u''(x) + qu(x) &= g(x), \quad 0 < x < 1, \\ u(0) = u(1) &= 0. \end{aligned} \tag{11.48}$$

mit einem Differenzenverfahren diskretisiert werden. Dazu wird das Intervall  $[0, 1]$  in  $n$  gleich lange Intervalle  $[x_i, x_{i+1}]$  aufgeteilt mit

$$x_i := i h, \quad i = 0, 1, \dots, n, \quad h := \frac{1}{n}. \tag{11.49}$$

Jetzt ersetzt man für jeden inneren Punkt  $x_i$  dieses Gitters die Differenzialgleichung durch eine algebraische Gleichung mit Näherungen  $u_i$  der Funktionswerte  $u(x_i)$  als Unbekannte, indem man alle Funktionen in  $x_i$  auswertet und die Ableitungswerte durch dividierte Differenzen approximiert, siehe Abschnitt 9.4.2. Das entstehende Gleichungssystem sei  $\mathbf{A}\mathbf{u} = \mathbf{f}$  und ist wie (9.66) tridiagonal und symmetrisch.

**Beispiel 11.8.** In einem ersten numerischen Experiment wollen wir das eindimensionale Problem (11.48) lösen mit  $q = 0$  und der rechten Seite  $g = 0$ .

$$\begin{aligned} -u''(x) &= 0, \quad 0 < x < 1, \\ u(0) = u(1) &= 0. \end{aligned} \tag{11.50}$$

Dann ist natürlich  $u = 0$  die exakte Lösung.

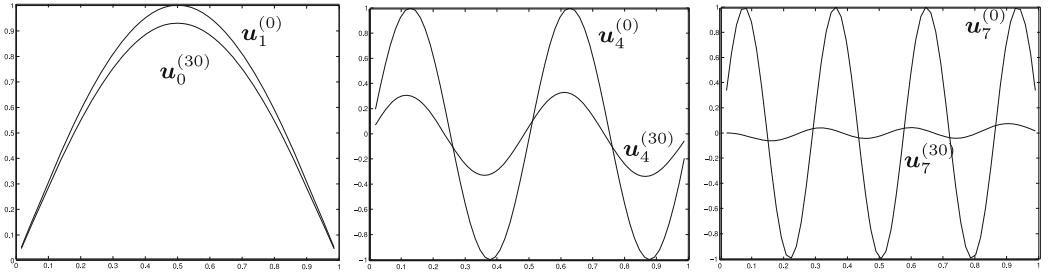
Wir diskretisieren dieses Problem mit  $h = 2^{-6}$ , also  $n = 64$ . Als Startvektor für die Iterationsverfahren nehmen wir jetzt Vektoren unterschiedlicher Frequenz

$$\mathbf{u}_k^{(0)} = (u_k^{(0)})_i = \sin\left(\frac{ik\pi}{n}\right), \quad k = 1, 4, 7.$$

Jetzt rechnen wir dreißig Iterationsschritte mit dem Einzelschrittverfahren (11.11). Das lautet algorithmisch so:

$$\begin{aligned} u_1 &:= u_2/2, \\ u_i &:= (u_{i-1} + u_{i+1})/2, \quad i = 2, \dots, n-2, \\ u_{n-1} &:= u_{n-2}/2. \end{aligned}$$

An der Abb. 11.6 sehen wir, dass der Startfehler, der hier mit dem Startvektor identisch ist, für verschiedene Frequenzen stark unterschiedlich gedämpft wird. Diese Beobachtung führte auf die Idee der Mehrgittermethoden.  $\triangle$

Abb. 11.6 Unterschiedliche Dämpfung der Fehlerfrequenzen  $k = 1, 4, 7$ .

### 11.3.2 Eigenschaften der gedämpften Jacobi-Iteration

Jetzt soll die gedämpfte Jacobi-Iteration (11.12) auf das triviale Problem (11.50) angewendet werden, weil dabei die Wirkung des Relaxationsparameters gut studiert werden kann. Wegen der einfachen Struktur des Problems und des Verfahrens geht das auch analytisch.

Für alle bisher betrachteten linearen und stationären Fixpunktiterationen gilt

$$\mathbf{u}^{(k+1)} = \mathbf{T}\mathbf{u}^{(k)} + \mathbf{c} \quad \text{und} \quad (11.51)$$

$$\mathbf{u} = \mathbf{T}\mathbf{u} + \mathbf{c}, \quad (11.52)$$

wenn  $\mathbf{u}$  die exakte Lösung des entsprechenden Gleichungssystems ist. Daraus folgt für den Fehler

$$\mathbf{e}^{(k+1)} = \mathbf{T}\mathbf{e}^{(k)} = \mathbf{T}^{k+1}\mathbf{e}^{(0)}. \quad (11.53)$$

Also stellt der Spektralradius von  $\mathbf{T}$  wieder die asymptotische Konvergenzrate dar. Nun gilt für die gedämpfte Jacobi-Iteration wegen (11.16) und (11.21) für (11.50)

$$\mathbf{T}_{\text{JOR}}(\omega) = (1 - \omega)\mathbf{I} + \omega\mathbf{T}_J = \mathbf{I} - \frac{\omega}{2} \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \\ 0 & \ddots & \ddots & \ddots & \ddots \\ \vdots & \ddots & \ddots & \ddots & -1 \\ 0 & \cdots & 0 & -1 & 2 \end{pmatrix} = \mathbf{I} - \frac{\omega}{2}\mathbf{A}.$$

Deshalb gilt für die Eigenwerte von  $\mathbf{T}_{\text{JOR}}(\omega)$  und  $\mathbf{A}$

$$\lambda(\mathbf{T}_{\text{JOR}}(\omega)) = 1 - \frac{\omega}{2}\lambda(\mathbf{A}). \quad (11.54)$$

Die Eigenwerte und Eigenvektoren dieser Standardmatrix sind bekannt als

$$\lambda_k(\mathbf{A}) = 4 \sin^2\left(\frac{k\pi}{2n}\right), \quad w_{k,j} = \sin\left(\frac{jk\pi}{n}\right), \quad k = 1, \dots, n-1. \quad (11.55)$$

Die Eigenvektoren von  $\mathbf{T}_{\text{JOR}}(\omega)$  stimmen mit denen von  $\mathbf{A}$  überein, während für die Eigenwerte gilt

$$\lambda(\mathbf{T}_{\text{JOR}}(\omega)) = 1 - 2\omega \sin^2\left(\frac{k\pi}{2n}\right), \quad k = 1, \dots, n-1. \quad (11.56)$$

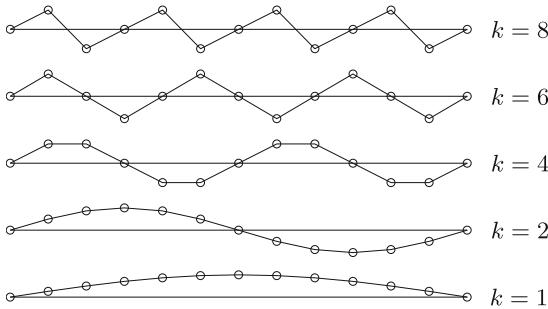


Abb. 11.7 Die Oszillationseigenschaft der diskreten Eigenvektoren.

Die Eigenvektoren sind für die folgenden Beobachtungen von großer Bedeutung. Zunächst halten wir (ohne Beweis) fest, dass sie eine wichtige Eigenschaft der Eigenfunktionen des kontinuierlichen Problems ins Diskrete übertragen: die Oszillationseigenschaft, siehe Abschnitt 9.1. In Abb. 11.7 sehen wir die  $k$ -ten Eigenvektoren für  $k = 1, 2, 4, 6, 8$ . Die Oszillationseigenschaft gilt im Diskreten allerdings nur für Frequenzen mit einer Wellenzahl kleiner als  $n$ . Wellen mit höherer Wellenzahl können auf dem groben Gitter nicht dargestellt werden. Der so genannte *Aliasing-Effekt* führt dazu, dass eine Welle mit einer Wellenlänge kleiner als  $2h$  auf dem Gitter als Welle mit einer Wellenlänge größer als  $2h$  erscheint.

Der Anfangsfehler  $\mathbf{e}^{(0)}$  wird als Fourier-Reihe der Eigenvektoren dargestellt<sup>1</sup>:

$$\mathbf{e}^{(0)} = \sum_{j=1}^{n-1} c_j \mathbf{w}_j. \quad (11.57)$$

Wegen (11.53) gilt dann

$$\mathbf{e}^{(k)} = (\mathbf{T}_{\text{JOR}}(\omega))^k \mathbf{e}^{(0)} = \sum_{j=1}^{n-1} c_j (\mathbf{T}_{\text{JOR}}(\omega))^k \mathbf{w}_j = \sum_{j=1}^{n-1} c_j \lambda_j^k (\mathbf{T}_{\text{JOR}}(\omega)) \mathbf{w}_j.$$

Das bedeutet, dass die  $j$ -te Frequenz des Fehlers nach  $k$  Iterationsschritten um den Faktor  $\lambda_j^k (\mathbf{T}_{\text{JOR}}(\omega))$  reduziert wird. Wir sehen auch, dass die gedämpfte Jacobi-Iteration die Fehlerfrequenzen nicht mischt; wenn wir sie auf einen Fehler mit nur einer Frequenz (also  $c_l \neq 0$ , aber  $c_j = 0$  für alle  $j \neq l$  in (11.57)) anwenden, dann ändert die Iteration die Amplitude dieser Frequenz, aber es entsteht keine andere Frequenz neu. Das liegt daran, dass die  $\mathbf{w}_j$  Eigenvektoren sowohl der Problematrix  $\mathbf{A}$  als auch der Iterationsmatrix  $\mathbf{T}_{\text{JOR}}(\omega)$  sind. Dies gilt nicht für jede stationäre Iteration.

Wir wenden uns jetzt der Frequenz-abhängigen Konvergenz zu, die wir schon im Beispiel 11.8 beobachtet haben. Dazu führen wir folgende nahe liegenden Bezeichnungen ein:

- Terme der Fourier-Reihe des Fehlers oder der Lösung mit einer Wellenzahl  $1 \leq k < n/2$  nennen wir *niederfrequent* oder *glatt*.
- Terme der Fourier-Reihe des Fehlers oder der Lösung mit einer Wellenzahl  $n/2 \leq k \leq n-1$  nennen wir *hochfrequent* oder *oszillatorisch*.

<sup>1</sup> Wegen ihrer linearen Unabhängigkeit bilden die Eigenvektoren eine Basis des Vektorraumes.

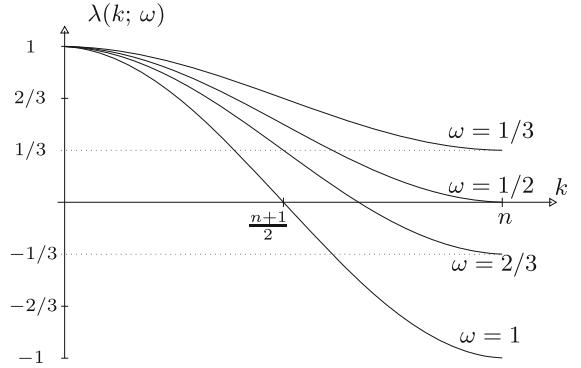


Abb. 11.8 Die Glättungseigenschaft der gedämpften Jacobi-Iteration: Die Eigenwerte  $\lambda_k(\mathbf{T}_{\text{JOR}}(\omega))$  abhängig von Wellenzahl  $k$  und Dämpfung  $\omega$ .

Wir suchen jetzt den optimalen Wert für  $\omega$  in (11.56). Das ist der Wert, der  $\lambda_k(\mathbf{T}_{\text{JOR}}(\omega))$  am kleinsten macht für alle  $k$  zwischen 1 und  $n - 1$ . Nun ist aber

$$\lambda_1 = 1 - 2\omega \sin^2\left(\frac{\pi}{2n}\right) = 1 - 2\omega \sin^2\left(\frac{h\pi}{2}\right) \approx 1 - \frac{\omega h^2 \pi^2}{2}. \quad (11.58)$$

Das bedeutet, dass der Eigenwert zum glattesten Eigenvektor immer nahe bei 1 liegt. Es gibt deshalb keinen Wert für  $\omega$ , der die glatten Komponenten des Fehlers schnell reduziert. Wenn wir mit der gedämpften Jacobi-Iteration die glatten Komponenten des Fehlers nicht effizient reduzieren können, wollen wir dies wenigstens für die oszillatorischen Komponenten versuchen, also für  $n/2 \leq k \leq n - 1$ . Diese Forderung können wir dadurch erfüllen, dass wir  $\lambda_{n/2} = -\lambda_n$  fordern. Die Lösung dieser Gleichung führt auf den Wert  $\omega = \frac{2}{3}$ , siehe Abb. 11.8. Es ist leicht zu zeigen, dass mit  $\omega = \frac{2}{3}$  für alle hochfrequenten Eigenwerte  $|\lambda_k| < 1/3$  gilt,  $n/2 \leq k \leq n - 1$ . Diesen Wert nennt man *Glättungsfaktor (smoothing factor)*. Ein kleiner Glättungsfaktor, der wie hier auch noch unabhängig von der Gitterweite  $h$  ist, stellt eine wichtige Grundlage zur Konstruktion von Mehrgittermethoden dar.

**Beispiel 11.9.** Abschließend wollen wir die gedämpfte Jacobi-Iteration mit  $\omega = 2/3$  auf das triviale Problem (11.50) anwenden, wie im Beispiel 11.8 schon das Gauß-Seidel-Verfahren. Da der Effekt ganz ähnlich ist, wollen wir nur die Glättung einer überlagerten Schwingung zeigen, an der die Glättungseigenschaft besonders deutlich wird, siehe Abb. 11.9.  $\triangle$

### 11.3.3 Ideen für ein Zweigitterverfahren

Aufbauend auf den vorangegangenen Beobachtungen beim eindimensionalen Modellproblem soll jetzt ein Verfahren auf zwei Gittern entwickelt werden. Eine Möglichkeit, die langsame Konvergenz eines Relaxationsverfahrens zu verbessern, liegt in der Wahl einer guten Startnäherung. Diese kann man z.B. durch einige Schritte eines Iterationsverfahrens auf einem groben Gitter erhalten. Auf ihm ist einerseits die Konvergenzrate  $O(1 - h^2)$  geringfügig besser als auf einem feinen Gitter, andererseits ist der Aufwand für einen Iterationsschritt geringer.

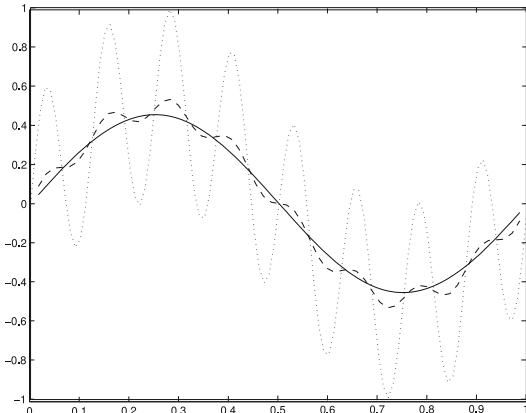


Abb. 11.9 Dämpfung des Fehlers  $(w_2 + w_{16})/2$  ( $\cdots$ ) nach 10 (- -) und nach 30 (—) Iterationen.

Zur Vorbereitung betrachten wir eine niederfrequente Funktion auf dem feinen Gitter, die wir dann auf das grobe Gitter projizieren. In Abb. 11.10 sehen wir eine Welle mit fünf Nullstellen ( $k = 4$ ) auf einem Gitter  $G_h$  mit 11 inneren Punkten ( $n = 12$ ,  $k < n/2$ ). Die Projektion dieser Welle auf ein Gitter  $G_{2h}$  mit 5 inneren Punkten ( $n = 6$ ) stellt auf diesem eine hochfrequente Funktion ( $k \geq n/2$ ) dar. Als Formel lässt sich diese Tatsache so ausdrücken:

$$w_{k,2j}^h = \sin\left(\frac{2jk\pi}{n}\right) = \sin\left(\frac{jk\pi}{n/2}\right) = w_{k,j}^{2h}, \quad 1 \leq k < \frac{n}{2}. \quad (11.59)$$

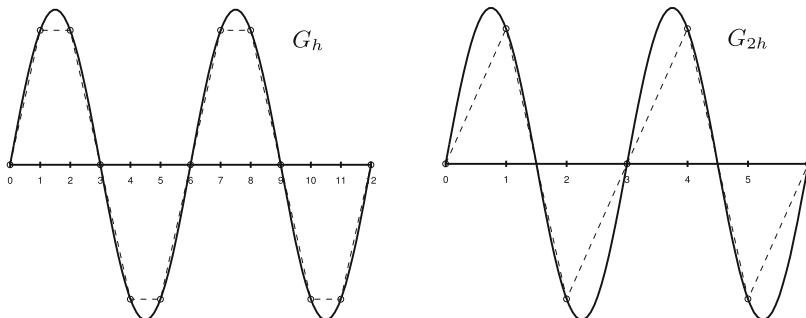


Abb. 11.10 Die Welle “ $k = 4$ ” auf feinem und grobem Gitter

Wir halten fest, dass Wellen einer gewissen Frequenz auf einem groben Gitter oszillierischer sind als auf einem feinen Gitter. Wechseln wir also von einem feinen auf ein grobes Gitter, dann ist die Relaxation effizienter, die Fehlerkomponenten einer gewissen Frequenz werden schneller kleiner. Wir müssen deshalb den Vorgang des Wechsels zwischen zwei Gittern formalisieren.

Es sei an die Nachiteration aus Kapitel 2 erinnert. Wenn  $v$  eine Näherungslösung von

$\mathbf{A}\mathbf{u} = \mathbf{f}$  ist, und wenn  $\mathbf{r} = \mathbf{f} - \mathbf{A}\mathbf{v}$  das zugehörige Residuum und  $\mathbf{e} = \mathbf{u} - \mathbf{v}$  der Fehler sind, dann gilt

$$\mathbf{A}\mathbf{e} = \mathbf{r}. \quad (11.60)$$

Das heißt, dass wir statt der Lösung auch den Fehler zu einer Näherungslösung mit einem Relaxationsverfahren behandeln können. Nehmen wir noch  $\mathbf{v} = \mathbf{0}$  als Näherungslösung, so kommen wir zu der Aussage

*Relaxation der Originalgleichung  $\mathbf{A}\mathbf{u} = \mathbf{f}$  mit einer beliebigen Startnäherung  $\mathbf{v}$  ist äquivalent zur Relaxation der Residuumsgleichung  $\mathbf{A}\mathbf{e} = \mathbf{r}$  mit einer Startnäherung  $\mathbf{e} = \mathbf{0}$ .*

Eine Mehrgittermethode sollte also feine Gitter zur Glättung und grobe Gitter zur Reduktion der niederfrequenten Fehleranteile benutzen. Das führt zu einer Strategie, die die Idee der Nachiteration aufgreift und hier als Zweigittermethode formuliert werden soll:

- Relaxiere  $\mathbf{A}\mathbf{u} = \mathbf{f}$  auf  $G^h$ , um eine Näherung  $\mathbf{v}^h$  zu bekommen.
- Berechne  $\mathbf{r} = \mathbf{f} - \mathbf{A}\mathbf{v}^h$ .
- Übertrage  $\mathbf{r}$  auf das grobe Gitter  $G^{2h}$ .
- Löse die Gleichung  $\mathbf{A}\mathbf{e} = \mathbf{r}$  auf  $G^{2h}$ .
- Übertrage  $\mathbf{e}$  auf das feine Gitter  $G^h$ .
- Korrigiere die Näherung  $\mathbf{v}^h := \mathbf{v}^h + \mathbf{e}$ .

Diese Strategie heißt *Korrektur-Schema (correction scheme)*. Für die Realisierung benötigen wir noch folgende Definitionen:

- Das Gleichungssystem  $\mathbf{A}\mathbf{u} = \mathbf{f}$  muss auf Gittern verschiedener Feinheit definiert werden.
- Es müssen Operatoren definiert werden, die Vektoren vom feinen Gitter  $G^h$  auf das grobe Gitter  $G^{2h}$  transformieren und umgekehrt.

Durch die Bezeichnungsweise wird nahe gelegt, dass die Gitterweite sich immer um den Faktor 2 ändert. Dies ist die übliche Methode, und es gibt kaum einen Grund einen anderen Faktor zu nehmen. Der Faktor 2 erleichtert nämlich die Konstruktion der Abbildungen zwischen den verschiedenen Vektorräumen.

Um einen Vektor von einem groben auf ein feines Gitter zu übertragen, wird man Interpolationsmethoden wählen. So macht man aus wenigen viele Komponenten. Dadurch wird der Vektor ‘verlängert’. Man nennt deshalb diese Abbildung *Interpolation* oder *Prolongation*.

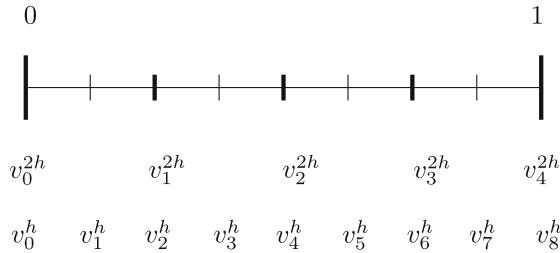
Für den umgekehrten Weg muss man aus vielen Werten wenige machen, diesen Vorgang nennt man deshalb *Restriktion*.

### 11.3.4 Eine eindimensionale Zweigittermethode

Für das eindimensionale Modellproblem (11.48) werden jetzt die Elemente eines Mehrgitterverfahrens eingeführt, mit denen dann zunächst eine Zweigittermethode konstruiert wird.

Das grobe Gitter  $G^{2h}$  und das feine Gitter  $G^h$  seien gegeben durch

$$\begin{aligned} G^{2h} &:= \{x \in \mathbb{R} \mid x = x_j = 2jh, j = 0, 1, \dots, n/2\}, & n \text{ gerade,} \\ G^h &:= \{x \in \mathbb{R} \mid x = x_j = jh, j = 0, 1, \dots, n\}. \end{aligned} \quad (11.61)$$



Die linearen Gleichungssysteme werden einfach durch die übliche Diskretisierung mit dem Differenzenverfahren auf den beiden Gittern erzeugt und mit dem entsprechenden Index gekennzeichnet als  $\mathbf{A}^h \mathbf{u}^h = \mathbf{f}^h$  bzw.  $\mathbf{A}^{2h} \mathbf{u}^{2h} = \mathbf{f}^{2h}$ .

### Interpolation (Prolongation)

Die Interpolation ist eine Abbildung von Vektoren  $\mathbf{v}^{2h}$ , die auf dem groben Gitter definiert sind, auf Vektoren  $\mathbf{v}^h$  des feinen Gitters; das soll hier vereinfacht dargestellt werden als

$$\mathbf{I}_{2h}^h : G^{2h} \rightarrow G^h.$$

Seien  $\mathbf{v}^h$  bzw.  $\mathbf{v}^{2h}$  Vektoren auf  $G^h$  bzw.  $G^{2h}$ . Dann wird die lineare Interpolation  $\mathbf{I}_{2h}^h$  definiert als

$$\mathbf{I}_{2h}^h \mathbf{v}^{2h} = \mathbf{v}^h \text{ mit } \begin{cases} v_{2i}^h = v_i^{2h} & \text{für } 1 \leq i \leq \frac{n}{2} - 1, \\ v_{2i+1}^h = \frac{1}{2}(v_i^{2h} + v_{i+1}^{2h}) & \text{für } 0 \leq i \leq \frac{n}{2} - 1. \end{cases} \quad (11.62)$$

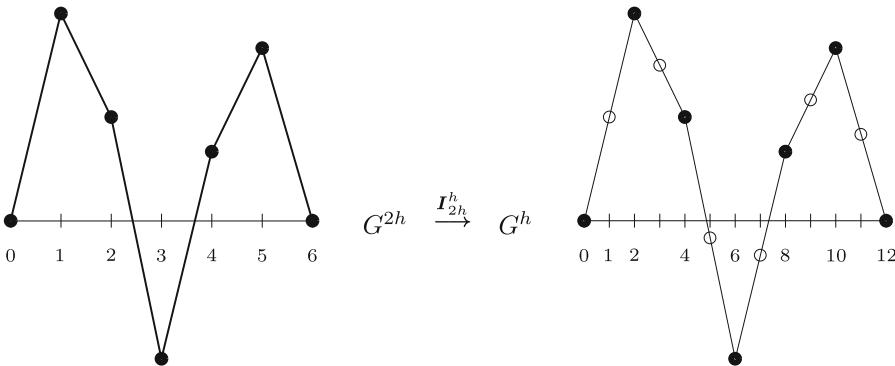


Abb. 11.11 Lineare Interpolation eines Vektors vom groben auf das feine Gitter.

Da die Randwerte nicht als Unbekannte in das lineare Gleichungssystem eingehen, ist die Ordnung der Matrizen  $n - 1$  bzw.  $n/2 - 1$  auf feinem bzw. grobem Gitter.  $\mathbf{I}_{2h}^h$  ist deshalb eine lineare Abbildung vom  $\mathbb{R}^{n/2-1}$  in den  $\mathbb{R}^{n-1}$ .

Für  $n = 8$  wird sie geschrieben als

$$\begin{pmatrix} v_1^h \\ v_2^h \\ v_3^h \\ v_4^h \\ v_5^h \\ v_6^h \\ v_7^h \end{pmatrix}_{7 \times 1} := \begin{pmatrix} 1/2 & & & & & & \\ & 1 & & & & & \\ & & 1/2 & 1/2 & & & \\ & & & 1 & & & \\ & & & & 1/2 & 1/2 & \\ & & & & & 1 & \\ & & & & & & 1/2 \end{pmatrix}_{7 \times 3} \begin{pmatrix} v_1^{2h} \\ v_2^{2h} \\ v_3^{2h} \end{pmatrix}_{3 \times 1} \quad (11.63)$$

$\mathbf{I}_{2h}^h$  hat vollen Rang, der Nullraum besteht nur aus dem Nullelement.

Bei der Mehrgittermethode wird die Abbildung auf Fehler-Vektoren angewendet. Wenn wir davon ausgehen, dass der Fehler auf dem feinen Gitter glatt ist, weil die hochfrequenten Fehleranteile durch Relaxation stark gedämpft wurden, dann stellt die lineare Interpolation eine gute Approximationsmethode dar.

### Restriktion

Die Restriktion ist jetzt die Abbildung in der Gegenrichtung, in vereinfachter Schreibweise

$$\mathbf{I}_h^{2h} : G^h \rightarrow G^{2h}.$$

Es soll hier nur die Restriktion durch einen so genannten *full-weighting*- oder FW-Operator betrachtet werden. Er berücksichtigt den Grobgitter-Punkt und zwei Nachbarwerte auf dem feinen Gitter. Aus den zusammen drei Werten wird ein gewichtetes Mittel gebildet

$$\mathbf{I}_h^{2h} \mathbf{v}^h = \mathbf{v}^{2h} \quad \text{mit} \quad v_i^{2h} = \frac{1}{4}(v_{2i-1}^h + 2v_{2i}^h + v_{2i+1}^h) \quad (11.64)$$

$\mathbf{I}_h^{2h}$  ist ein linearer Operator von  $\mathbb{R}^{n-1}$  nach  $\mathbb{R}^{n/2-1}$ . Für  $n = 8$  bekommen wir

$$\begin{pmatrix} v_1^{2h} \\ v_2^{2h} \\ v_3^{2h} \end{pmatrix} := \begin{pmatrix} 1/4 & 1/2 & 1/4 & & & & \\ & 1/4 & 1/2 & 1/4 & & & \\ & & 1/4 & 1/2 & 1/4 & & \\ & & & 1/4 & 1/2 & 1/4 & \end{pmatrix} \begin{pmatrix} v_1^h \\ v_2^h \\ v_3^h \\ v_4^h \\ v_5^h \\ v_6^h \\ v_7^h \end{pmatrix} \quad (11.65)$$

Es ist  $\text{Rang}(\mathbf{I}_h^{2h}) = \frac{n}{2} - 1$ , also ist die Dimension des Nullraums  $\dim(N(\mathbf{I}_h^{2h})) = \frac{n}{2}$ .

### Beziehungen zwischen Prolongation und Restriktion

Die Interpolationsmatrix in (11.63) ist bis auf einen konstanten Faktor gleich der Transponierten der Restriktionsmatrix in (11.65). Die Operatoren stehen damit in einer Beziehung, die auch als allgemeine Forderung sehr sinnvoll ist, sie sind (quasi) *adjungiert*:

$$\mathbf{I}_{2h}^h = c (\mathbf{I}_h^{2h})^T \quad \text{für ein } c \in \mathbb{R}. \quad (11.66)$$

Diese Tatsache bezeichnet man auch als *Variationseigenschaft*.

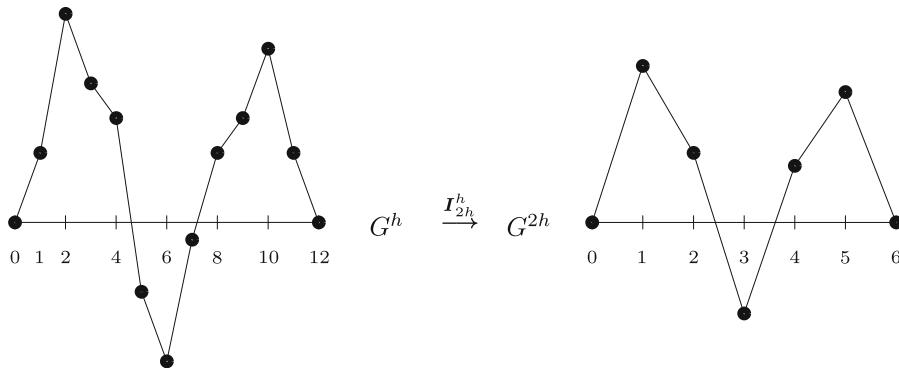


Abb. 11.12 Restriktion eines Vektors vom feinen auf das grobe Gitter.

**Zweigitter-Korrektur-Schema:**  $\mathbf{v}^h \leftarrow \text{MG}(\mathbf{v}^h, \mathbf{f})$

Relaxation:	Relaxiere $\nu_1$ mal $\mathbf{A}^h \mathbf{u}^h = \mathbf{f}^h$ auf $G^h$ mit der Start-Näherung $\mathbf{v}^h$ und dem Ergebnis $\mathbf{v}^h$ .
Residuum:	Berechne $\mathbf{r}^h := \mathbf{f}^h - \mathbf{A}^h \mathbf{v}^h$ .
Restriktion:	Berechne $\mathbf{r}^{2h} := I_{2h}^h \mathbf{r}^h$ .
Lösung:	Löse $\mathbf{A}^{2h} \mathbf{e}^{2h} = \mathbf{r}^{2h}$ auf $G^{2h}$ mit einem direkten Verfahren (Cholesky, Gauß).
Interpolation:	Berechne $\mathbf{e}^h := I_{2h}^h \mathbf{e}^{2h}$ .
Korrektur:	Korrigiere die Näherung $\mathbf{v}^h := \mathbf{v}^h + \mathbf{e}^h$ .
Relaxation:	Relaxiere $\nu_2$ mal $\mathbf{A}^h \mathbf{u}^h = \mathbf{f}^h$ auf $G^h$ mit der Start-Näherung $\mathbf{v}^h$ und dem Ergebnis $\mathbf{v}^h$ .

(11.67)

Dies ist das Korrektur-Schema aus Abschnitt 11.3.3, jetzt nur mit konkreten Operatoren. Auf dem feinen Gitter wird am Anfang und Ende des Zweigitter-Schrittes  $\nu_1$  mal bzw.  $\nu_2$  mal relaxiert, wobei  $\nu_1$  und  $\nu_2 = 1, 2$  oder  $3$  sind, auf dem groben Gitter wird ‘exakt’ gelöst.

Dieser Algorithmus kann folgendermaßen kommentiert und erweitert werden:

- Der Algorithmus wird zum Iterationsverfahren, indem  $\mathbf{v}^h \leftarrow \text{MG}(\mathbf{v}^h, \mathbf{f})$  solange aufgerufen wird, bis eine Fehlerabfrage erfüllt ist wie z.B.  $\|\mathbf{v}_{\text{alt}}^h - \mathbf{v}_{\text{neu}}^h\| < \varepsilon$ .
- $\nu_1$  und  $\nu_2$  sind Parameter des Algorithmus, sie werden normalerweise auf zwei konstante Werte festgelegt, können aber auch von Schritt zu Schritt variieren.
- Die exakte Lösung auf dem groben Gitter kann auch durch eine Näherungslösung ersetzt werden, z.B. wenn die exakte Lösung zu aufwändig erscheint.
- Das Zweigitterverfahren wird zum Mehrgitterverfahren, wenn man die exakte Lösung auf dem groben Gitter rekursiv durch ein Zweigitterverfahren ersetzt.
- Bei einem Mehrgitterverfahren mit mehr als zwei Gittern wird unterschiedlich geglättet. Wir haben ja gesehen, dass ‘hochfrequent’ auf verschiedenen feinen Gittern unterschiedliche Frequenzbereiche bezeichnet.

**Beispiel 11.10.** Das Balkenproblem (Beispiel 9.3) soll mit der Zweigittermethode behandelt werden. Es ist durch eine lineare Transformation leicht auf die Form des Problems (11.48) zu bringen, und es spielt auch hier keine Rolle, dass die Koeffizienten-Funktion  $q$  negative Werte annimmt. Es wird mit folgenden Parametern gerechnet:

JOR-Methode mit  $\omega = 2/3$  als Relaxation,  $\nu_1 = \nu_2 = 3$  Vor- und Nach-Relaxationen,  $n = 63$  innere Gitterpunkte.

Da die exakte Lösung auf dem groben Gitter schon eine sehr gute Näherung darstellt und das gedämpfte Jacobi-Verfahren gut glättet, soll die Lösung durch den besonders ungünstigen Startvektor

$$u = \frac{x+1}{3} + 0.2(\sin(\frac{n}{4}\pi x) + 1)$$

erschwert werden. Diese Startnäherung ist die Summe einer glatten und einer stark oszillierenden Funktion; ihre Werte liegen im Bereich der Lösungswerte, aber mit einem großen Fehler direkt neben den Randpunkten.

In Bild 11.13 sehen wir oben links neben der Lösung (- -) die Startnäherung. Daneben sehen wir, dass drei Schritte des gedämpften Jacobi-Verfahrens die Startnäherung schon gut glätten. Die Unsymmetrie und Randabweichung der Startnäherung ist nach einem V-Zyklus unten links nur noch schwach in der Nähe des rechten Randes zu sehen, nach fünf V-Zyklen ist die Näherungslösung von der korrekten Lösung graphisch nicht mehr zu unterscheiden. Der Fehler in der euklidischen und der Maximumsnorm liegen dann bei  $10^{-6}$ , die Konvergenzrate bei 0.04. Dieses außerordentlich gute Verhalten des Verfahrens liegt daran, dass die exakte Lösung auf dem groben Gitter schon sehr gut die Lösung auf dem feinen Gitter approximiert. Um mit dem klassischen Jacobi-Verfahren (11.9) mit  $\omega = 1$  dieselbe Genauigkeit zu erzielen, müssten fast 1000 Iterationsschritte ausgeführt werden; das würde etwa die zehnfache Rechenzeit benötigen.

Vergleicht man die Konvergenzraten für verschiedene Werte von  $n$ , so ist das Ziel einer von  $h$  unabhängigen Konvergenzrate nicht vollständig erreicht, die Schwankungen sind aber sehr gering. Bei Mehrgitterverfahren mit mehr als zwei Gittern erwarten wir eine von  $h$  vollkommen unabhängige Konvergenz.

△

### 11.3.5 Eine erste Mehrgittermethode

Eine Mehrgittermethode mit mehr als zwei Stufen kann jetzt leicht aus dem Zweigitter-Algorithmus (11.67) konstruiert werden. Der Schritt *Lösung* auf dem groben Gitter wird sukzessive durch eine Zweigittermethode ersetzt. Wird das z.B. zweimal gemacht, liegt eine Viergittermethode vor. Das ist eine rekursive Vorgehensweise, deshalb liegt es auch nahe den zugehörigen Algorithmus entsprechend zu formulieren. Da dabei  $h$  und  $2h$  durch den rekursiven Aufruf ihre Werte während der Rechnung ändern, muss noch die größte Gitterweite  $h_g$  definiert sein. Außerdem wird eine allgemeine  $\nu$ -fache Glättungsiteration

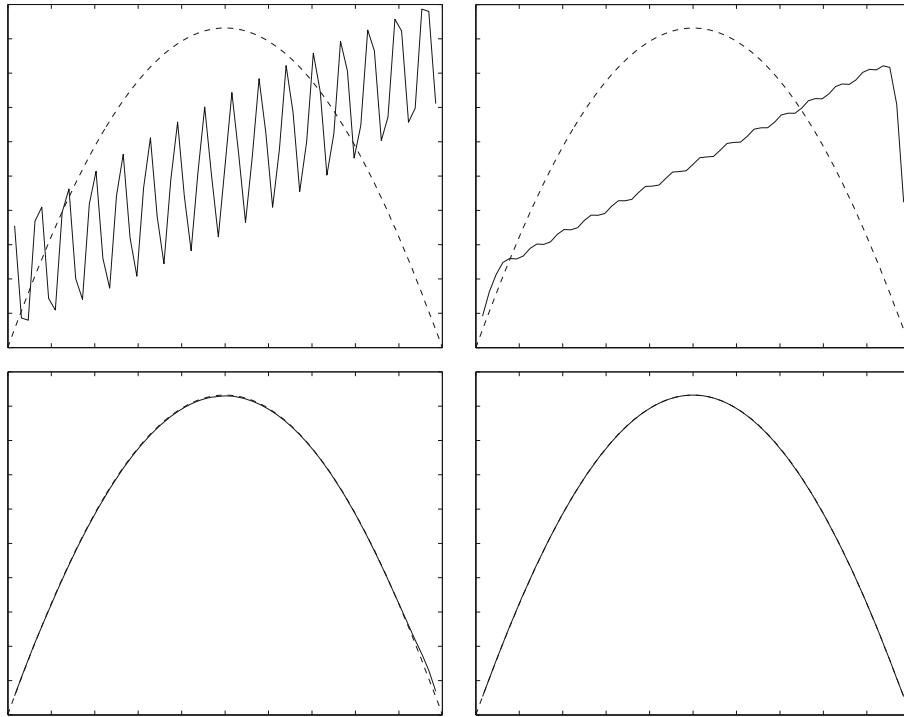


Abb. 11.13 Zweigittermethode mit oszillierendem Anfangsvektor (oben links), Näherungslösung nach drei gedämpften Jacobi-Relaxationen (oben rechts) und nach einem und fünf Gesamtschritten (unten).

mit dem Operator  $\left(S^h(\mathbf{v}^h, \mathbf{f}^h)\right)^\nu$  bezeichnet.

$\mathbf{v}^h := \text{V}(\mathbf{v}^h, \mathbf{f}^h, \nu_1, \nu_2)$ <hr/> if $h = h_g$ then $\mathbf{v}^h = (\mathbf{A}^h)^{-1} \mathbf{f}^h$ RETURN else $\mathbf{v}^h := \left(S^h(\mathbf{v}^h, \mathbf{f}^h)\right)^{\nu_1}$ $\mathbf{f}^{2h} := \mathbf{I}_{2h}^{2h} (\mathbf{f}^h - \mathbf{A}^h \mathbf{v}^h)$ $\mathbf{v}^{2h} := \mathbf{0}$ $\mathbf{v}^{2h} := \text{V}(\mathbf{v}^{2h}, \mathbf{f}^{2h}, \nu_1, \nu_2)$ $\mathbf{v}^h = \mathbf{v}^h + \mathbf{I}_{2h}^h \mathbf{v}^{2h}$ $\mathbf{v}^h := \left(S^h(\mathbf{v}^h, \mathbf{f}^h)\right)^{\nu_2}$ RETURN end	(11.68)
---	---------

Um die Rekursion in diesem Algorithmen richtig zu verstehen, muss man bedenken, dass jeder Aufruf der Routine V mit einem RETURN endet, von dem ein Rücksprung in den voran gegangenen Aufruf von V solange erfolgt, bis der Rücksprung in den ersten Aufruf erreicht ist. Auf diesen so genannten V-Zyklus kommen wir bei den zweidimensionalen Mehrgittermethoden zurück.

Wenn im eindimensionalen Fall ein feinstes Gitter mit  $n - 1$  inneren Punkten vorliegt und  $n$  eine Zweierpotenz ist, dann kann solange vergröbert werden, bis das Gitter nur noch aus einem inneren Punkt besteht. Diese Vorgehensweise soll beim nächsten Beispiel angewendet werden.

**Beispiel 11.11.** Wir kehren zu unserem Balkenbeispiel (9.3) zurück, das wir schon mit dem Zweigitterverfahren (11.67) in Beispiel 11.10 behandelt haben. Es soll jetzt für verschiedene Werte von  $n$  und von den anderen Parametern mit V-Zyklen behandelt werden. Es sollen aber auch einige neue Aspekte einbezogen werden. Dazu gehören der Einsatz des Gauß-Seidelschen Einzelschrittverfahrens als Glätter, die Abhängigkeit der Konvergenzrate von  $h$  und der Vergleich des Aufwands der verschiedenen Methoden.

Ein Ergebnis nehmen wir vorweg: Die Abhängigkeit der Konvergenzrate von der Startnäherung ist sehr gering, es gibt also keinen Grund, eine andere als die Startnäherung  $v^h = \mathbf{0}$  zu wählen. Das haben wir deshalb grundsätzlich getan.

Da die Mehrgittermethode jetzt immer bis zum gröbsten Gitter mit nur einem inneren Punkt herab steigt, fällt der Vorteil einer schon sehr genauen exakten Lösung dort weg; dadurch geben die erzielten Konvergenzraten ein realistischeres Bild über die Eigenschaften der Verfahren, als dies in Beispiel 11.10 bei der Zweigittermethode der Fall war.

Wir fassen unsere Ergebnisse in Tabellen zusammen, die wir dann im Einzelnen erläutern.

Tabelle 11.2 zeigt, dass das Einzelschrittverfahren (GS) dem Gesamtschrittverfahren (JOR) weit überlegen ist, und, dass die  $h$ -unabhängige Konvergenz so gut wie erreicht ist.

Um die Zahlen in Tabelle 11.3 zu interpretieren, muss der Aufwand für einen Zyklus abhängig von  $\nu$  untersucht werden, siehe Abschnitt 11.3.8. Wenn man wie dort davon ausgeht, dass die Relaxationsschritte den Hauptaufwand darstellen, also die restlichen Anteile des Algorithmus bei der Aufwandsabschätzung vernachlässigt werden können, dann zeigen diese Zahlen, dass  $\nu = 1$  schon optimal ist, denn es ist sowohl  $0.51^2 < 0.37$  als auch  $0.26^2 < 0.1$ , also sind unter dieser Annahme zwei Zyklen mit  $\nu = 1$  für die Fehlerreduktion günstiger als ein Zyklus mit  $\nu = 2$ . Für größere Werte von  $\nu$  ist das noch ausgeprägter. Diese Aussage bleibt korrekt, wenn man 30 % des Aufwand eines Relaxationsschrittes als Aufwand für die anderen Verfahrensanteile mit berücksichtigt.

Tab. 11.2 Konvergenzraten für das Balkenbeispiel mit V-Zyklen und Gauß-Seidel bzw. JOR(2/3) als Glätter für unterschiedliche feinste Gitter, immer mit  $\nu_1 = \nu_2 = 2$ .

$m$	$h$	GS	JOR
3	0.25	0.11	0.385
4	0.125	0.10	0.375
5	0.0625	0.10	0.372
6	0.03125	0.10	0.371

Tab. 11.3 Konvergenzraten für das Balkenbeispiel mit V-Zyklen und Gauß-Seidel bzw. JOR(2/3) als Glätter mit  $n = 63$  inneren Punkten für verschiedene Relaxationsparameterwerte  $\nu := \nu_1 = \nu_2$ .

$m$	$\nu$	GS	JOR
6	1	0.26	0.51
6	2	0.10	0.37
6	3	0.04	0.27
6	4	0.015	0.20
6	5	0.0075	0.14
6	6	0.0045	0.11

Die bestmögliche Genauigkeit, die angestrebt werden kann, liegt bei  $O(h^2)$ , da das die Größenordnung des Diskretisierungsfehlers ist, dessen Unterschreitung bei der Lösung des diskretisierten Gleichungssystems keinen Sinn macht. Dieses Ziel erreicht das Gauß-Seidel-Verfahren beim Balkenbeispiel für alle untersuchten Werte von  $n = 2^m$  mit vier V-Zyklen. Das Jacobi-Verfahren mit  $\omega = 2/3$  benötigt mindestens zwei Zyklen mehr.  $\triangle$

### 11.3.6 Die Mehrgitter-Operatoren für das zweidimensionale Modellproblem

Jetzt soll das zweidimensionale Modellproblem (11.43) mit einer Mehrgittermethode behandelt werden. Um das Zweigitter-Korrektur-Schema (11.67) anwenden zu können, müssen die Operatoren (Matrizen)  $\mathbf{A}^h$ ,  $\mathbf{A}^{2h}$ ,  $\mathbf{I}_h^{2h}$  und  $\mathbf{I}_{2h}^h$  definiert werden.  $\mathbf{A}^h$  und  $\mathbf{A}^{2h}$  seien die in Beispiel 11.1 hergeleiteten Matrizen für  $h = 1/(N+1)$  bzw.  $2h = 2/(N+1)$ . Die Operatoren  $\mathbf{I}_h^{2h}$ ,  $\mathbf{I}_{2h}^h$  werden am einfachsten durch ihre ‘‘Sterne’’, d.h. ihre Wirkung auf einen inneren Punkt in  $G_h$  bzw.  $G_{2h}$  repräsentiert.

#### Restriktionsoperator

Es sollen wieder die Abbildungen zwischen den Vektorräumen durch die zwischen den zugehörigen Gittern symbolisiert werden. Dann ist die Restriktion eine Abbildung

$$\mathbf{I}_h^{2h} : G_h \rightarrow G_{2h}.$$

Auch hier soll nur der full-weighting oder FW-Operator definiert werden:

$$\begin{array}{c}
 \frac{1}{16} * \begin{array}{|c|c|} \hline
 \begin{array}{|c|c|} \hline
 & 1 \\ \hline
 1 & 2 & 1 \\ \hline
 & 2 \\ \hline
 \end{array} &
 \begin{array}{|c|c|} \hline
 2 & 1 \\ \hline
 4 & 2 \\ \hline
 2 & \\ \hline
 \end{array} \\
 \hline
 \end{array} \\
 \end{array}$$

$$\mathbf{I}_h^{2h} \hat{=} \frac{1}{16} \left[ \begin{array}{ccc} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{array} \right]_h^{2h} \quad (11.69)$$

$$G_{2h} : —, \quad G_h : — \text{ und } -\cdots-$$

Es wird ein gewichtetes Mittel unter allen Nachbarn des Grobgitterpunktes auf dem feinen Gitter gebildet. Bei zweidimensionaler Indizierung ergibt sich die Gleichung

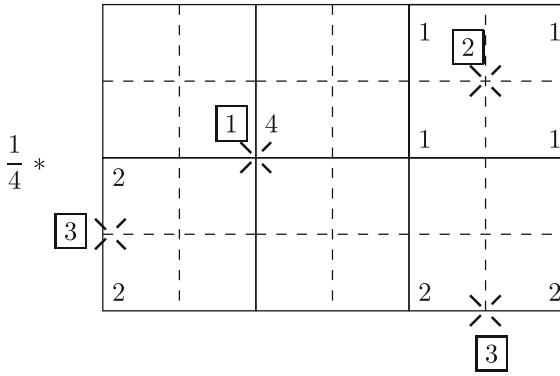
$$\begin{aligned} v_{i,j}^{2h} = & \frac{1}{16} [v_{2i-1,2j-1}^h + v_{2i-1,2j+1}^h + v_{2i+1,2j-1}^h + v_{2i+1,2j+1}^h \\ & + 2(v_{2i,2j-1}^h + v_{2i,2j+1}^h + v_{2i-1,2j}^h + v_{2i+1,2j}^h) \\ & + 4v_{2i,2j}^h], \quad 1 \leq i, j \leq \frac{n}{2} - 1. \end{aligned} \quad (11.70)$$

### Interpolationsoperator

Die in Rückrichtung zu definierende Prolongation

$$I_{2h}^h : G_{2h} \rightarrow G_h$$

soll eine zur FW-Restriktion (quasi) adjungierte Abbildung sein. Das gelingt mit



$$I_{2h}^h \hat{=} \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_{2h}^h \quad (11.71)$$

$G_{2h}$  : —,  $G_h$  : — und - - -

Hier müssen drei Fälle unterschieden werden:

- Fall **[1]**:

$x \in G_h$  und  $x \in G_{2h}$ , d.h. kein Nachbarpunkt von  $x$  im feinen Gitter liegt auf dem groben Gitter. Dann wird der Wert in  $x$  übernommen.

- Fall **[2]**:

$x \notin G_{2h}$ , 4 Nachbarn  $\in G_{2h}$ .

Dann bekommen diese den Faktor 1/4.

- Fall **[3]**:

$x \notin G_{2h}$ , 2 Nachbarn  $\in G_{2h}$ .

Dann bekommen diese den Faktor 1/2.

Diese Fall-unterscheidende Vorschrift lautet bei zweidimensionaler Indizierung so:

$$\begin{aligned} v_{2i,2j}^h &= v_{ij}^{2h} \\ v_{2i+1,2j}^h &= \frac{1}{2}(v_{ij}^{2h} + v_{i+1,j}^{2h}) \\ v_{2i,2j+1}^h &= \frac{1}{2}(v_{ij}^{2h} + v_{i,j+1}^{2h}) \\ v_{2i+1,2j+1}^h &= \frac{1}{4}(v_{ij}^{2h} + v_{i+1,j}^{2h} + v_{i,j+1}^{2h} + v_{i+1,j+1}^{2h}) \end{aligned} \quad (11.72)$$

Damit sind die Operatoren definiert, die als Module einer Mehrgittermethode benötigt werden.

Bevor ein Beispiel präsentiert wird, sollen verschiedene Formen von Mehrgittermethoden mit mehr als zwei Gittern eingeführt werden.

### 11.3.7 Vollständige Mehrgitterzyklen

Um verschiedene Formen von Mehrgittermethoden formulieren zu können, soll die Bezeichnungsweise durch Stufen-Bezeichnungen (engl. *level*) ergänzt werden:

Die Anzahl der verwendeten Gitter sei  $l + 1$ :

Das feinste Gitter  $G_0$  bekommt die Stufe  $m = 0$  entsprechend der Gitterweite  $h$ .

Das nächst gröbere Gitter  $G_1$  bekommt die Stufe  $m = 1$  entsprechend der Gitterweite  $2h$ .

Ein allgemeines Gitter  $G_m$  hat die Stufe  $m$  mit der Gitterweite  $h_m = 2^m h$ .

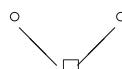
Das größte Gitter  $G_l$  bekommt die Stufe  $m = l$  mit der Gitterweite  $h_l = 2^l h$ .

Der Wechsel zwischen Gittern verschiedener Stufen und die angewendeten Methoden (Relaxation, Interpolation, Restriktion) werden durch Symbole gekennzeichnet, damit ein Mehrgitterzyklus graphisch dargestellt werden kann:

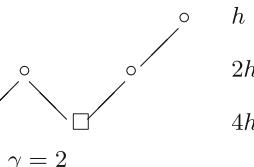
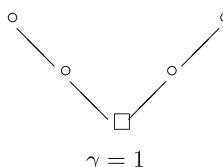
- Relaxation
- Exakte Lösung der Defektgleichung
- \ Restriktion
- / Interpolation
- $\gamma$  Wiederholungszahl der Lösungs/Relaxations-Stufen
- $\gamma = 1$  V-Zyklus
- $\gamma = 2$  W-Zyklus

#### Beispiel 11.12.

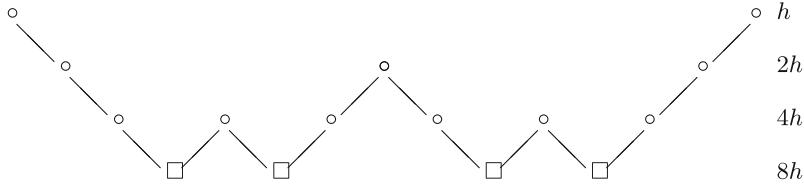
1. Zweigittermethode ( $\gamma$  ohne Bedeutung):



2. Dreigittermethoden:

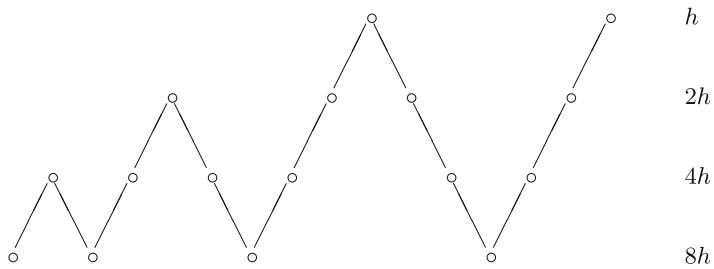


3. Viergittermethode mit  $\gamma = 2$ :



#### 4. FMG-Zyklus:

Neben den oben dargestellten  $V$ - und  $W$ -Zyklen gibt es noch eine andere Form, die sich bei vielen Beispielen als besonders effizient erweist, der *Full Multigrid*- oder FMG-Zyklus. Er startet im Gegensatz zu den anderen Zyklen auf dem größten Gitter und kann in seiner einfachsten Form für vier Stufen graphisch wie folgt dargestellt werden:



△

Die zugehörigen Algorithmen sollen nicht dargestellt werden. In ihrer elegantesten Form sind sie rekursiv definiert; das ist wegen der Stufen-Struktur nahe liegend. Stattdessen soll für den einfachsten Fall eines  $V$ -Zyklus kurz auf die Komplexität der Algorithmen eingegangen werden; ein wenig Theorie soll zeigen, warum die Mehrgittermethoden asymptotisch optimal sein können; abschließend sollen Beispiele die Effizienz der Methoden demonstrieren.

### 11.3.8 Komplexität

Was kostet ein Mehrgitter-Zyklus an Speicherplatz und Rechenzeit?

#### Speicherplatzbedarf

Dies ist einfach zu beantworten. Wir betrachten ein  $d$ -dimensionales Gitter mit  $n^d$  Punkten. Der Einfachheit halber sei  $n$  eine Potenz von 2. Auf jeder Stufe müssen zwei Felder gespeichert werden. Das feinste Gitter  $G^h$  benötigt also  $2n^d$  Speicherplätze. Das nächst gröbere Gitter benötigt um den Faktor  $2^d$  weniger Speicherplatz. Das führt insgesamt zu einem Speicherplatzbedarf von

$$S = 2n^d \left( 1 + 2^{-d} + 2^{-2d} + \dots + 2^{-nd} \right) < \frac{2n^d}{1 - 2^{-d}}.$$

Das bedeutet, dass für das eindimensionale Problem für alle Stufen zusammen weniger als das Doppelte des Speicherplatzes für die feinste Stufe benötigt wird. In zwei oder mehr Dimensionen ist es sogar weniger als das  $4/3$ -fache.

### Rechenaufwand

Wir wollen den Aufwand in Arbeitseinheiten messen. Eine Arbeitseinheit (AE) sei der Aufwand für einen Relaxationsschritt auf dem feinsten Gitter. Der Aufwand für die Interpolationen und Restriktionen wird vernachlässigt; er beträgt etwa 10 bis 20% des Aufwands für den gesamten Zyklus.

Wir betrachten einen  $V$ -Zyklus mit einem Relaxationsschritt auf jeder Stufe ( $\nu_1 = \nu_2 = 1$ ). Jede Stufe wird zweimal aufgesucht und das Gitter  $G^{ph}$  benötigt  $p^{-d}$  AE. Damit bekommen wir entsprechend zur Speicherplatz-Rechnung

$$Z_{V\text{-Zyklus}} = 2(1 + 2^{-d} + 2^{-2d} + \dots + 2^{-nd}) \text{ AE} < \frac{2}{1 - 2^{-d}} \text{ AE.}$$

Ein einziger  $V$ -Zyklus kostet also ungefähr 4 AE bei einem eindimensionalen Problem, ungefähr  $8/3$  AE für  $d = 2$  und  $16/7$  AE für  $d = 3$ . Die entscheidende Frage ist natürlich, wie gut die unterschiedlichen Mehrgitter-Schemata konvergieren, also wie viele Zyklen für eine bestimmte Genauigkeit durchlaufen werden müssen.

#### 11.3.9 Ein Hauch Theorie

Mit Hilfe von Frequenz-Analysen können Konvergenzraten und Glättungsfaktoren für die Modellprobleme bestimmt werden. Es soll eine lokale oder Fourier-Analyse in ihrer einfachsten Form präsentiert werden, um die wichtigen Glättungsfaktoren zu berechnen. In ihrer vollen Allgemeinheit kann die Fourier-Analyse auf allgemeine Operatoren und auf unterschiedliche Mehrgitter-Zyklen angewendet werden. Dann ist sie ein mächtiges Werkzeug, das die Leistung der Mehrgitter-Algorithmen mit den theoretischen Erwartungen vergleichen kann.

Die Fourier-Analyse setzt vereinfachend voraus, dass Konvergenz und Glättung lokale Effekte sind, d.h. dass ihre Wirkung auf einzelne Unbekannte beschränkt betrachtet werden kann. Sie hängt nur von den direkten Nachbarn ab, der Einfluss von Rändern und Randwerten kann vernachlässigt werden. So kann diese Wirkung auch analysiert werden, indem man das endliche Gebiet durch ein unendliches ersetzt.

#### Eindimensionale Fourier-Analyse

Wir betrachten ein lineares Relaxationsverfahren mit der Iterationsmatrix  $T$ . Für den Fehlervektor  $e^{(m)}$  gilt nach dem  $m$ -ten Schritt

$$e^{(m+1)} = Te^{(m)}.$$

Für die Fourier-Analyse wird vorausgesetzt, dass der Fehler sich als eine Fourier-Reihe darstellen lässt. Wir haben schon am Anfang dieses Abschnitts die Wirkung einer Relaxation auf einen einzelnen Fourier-Term  $w_j = \sin(jk\pi/n)$ ,  $1 \leq k \leq n$ , untersucht. Mit der neuen

Voraussetzung eines unendlichen Gebiets müssen die Fourier-Terme nicht mehr auf diskrete Frequenzzahlen beschränkt werden, die Werte  $jk\pi/n$  können durch eine kontinuierliche Variable  $\theta \in (-\pi, \pi]$  ersetzt werden. Deshalb untersuchen wir jetzt Fourier-Terme der Form  $w_j = e^{\iota j\theta}$  mit  $\iota := \sqrt{-1}$ . Zu einem festen Wert  $\theta$  hat der zugehörige Fourier-Term die Wellenlänge  $2\pi h/|\theta|$ . Werte von  $|\theta|$  nahe bei null entsprechen niederfrequenten Wellen, solche nahe bei  $\pi$  hochfrequenten Wellen. Die Benutzung einer komplexen Exponentialfunktion macht die Rechnung einfacher und berücksichtigt sowohl sin- als auch cos-Terme.

Es ist wichtig anzumerken, dass die Fourier-Analyse nur dann vollständig korrekt ist, wenn die Fourier-Terme Eigenvektoren der Relaxationsmatrix sind. Das ist i. A. nicht der Fall. Für höhere Frequenzen approximieren die Fourier-Terme aber die Eigenvektoren recht gut. Deshalb eignet sich die lokale Fourier-Analyse besonders gut zur Untersuchung der Glättungseigenschaft der Relaxation.

Wir setzen voraus, dass der Fehler nach dem  $m$ -ten Schritt der Relaxation am  $j$ -ten Gitterpunkt aus einem einzelnen Fourier-Term besteht, d. h. er hat die Form

$$e_j^m = A(m) e^{\iota j\theta}, \quad -\pi < \theta \leq \pi. \quad (11.73)$$

Ziel der lokalen Analyse ist es nun zu untersuchen, wie sich die Amplitude  $A(m)$  des Fourier-Terms mit jedem Relaxationsschritt ändert. Diese Änderung setzen wir an als

$$A(m+1) = G(\theta)A(m).$$

Die Funktion  $G$ , die die Entwicklung der Fehleramplitude beschreibt, heißt *Verstärkungsfaktor*. Es muss  $|G(\theta)| < 1$  für alle  $\theta$  sein, damit die Methode konvergiert. Unser Interesse gilt aber der Glättungseigenschaft, d. h. der Untersuchung des Verstärkungsfaktors für die hochfrequenten Fourier-Terme  $\pi/2 \leq |\theta| \leq \pi$ . Deshalb definieren wir den Glättungsfaktor als

$$\mu = \max_{\frac{\pi}{2} \leq |\theta| \leq \pi} |G(\theta)|.$$

Die oszillatorischen Fourier-Terme werden in jedem Relaxationsschritt um mindestens diesen Faktor verkleinert. Wir wollen dies an einem Beispiel durchrechnen.

**Beispiel 11.13.** Wir betrachten das eindimensionale Modellproblem

$$-u''(x) + q(x)u(x) = f(x).$$

$v_j$  seien die Approximationen von  $u(x_j)$  auf einem äquidistanten Gitter mit der Gitterweite  $h$ . Wir wenden wieder die gedämpfte Jacobi-Relaxation auf das mit den üblichen zentralen zweiten dividierten Differenzen diskretisierte Problem an:

$$v_j^{m+1} = \frac{\omega}{2 + h^2 q_j} (v_{j-1}^m + v_{j+1}^m + h^2 f_j) + (1 - \omega) v_j^m, \quad (11.74)$$

mit  $q_j = q(x_j)$ . Da der Fehler  $e_j = u(x_j) - v_j$  für jeden Relaxationsschritt mit der Iterationsmatrix multipliziert wird, gilt für ihn am  $j$ -ten Gitterpunkt die entsprechende Gleichung

$$e_j^{m+1} = \frac{\omega}{2 + h^2 q_j} (e_{j+1}^m + e_{j-1}^m) + (1 - \omega) e_j^m. \quad (11.75)$$

Jetzt wird vorausgesetzt, dass der Fehler aus einem einzelnen Fourier-Term der Form (11.73) besteht. Die Fehlerdarstellung (11.75) zeigt, dass der Fehler für kleine  $h$  nur schwach von der Koeffi-

zientenfunktion  $q(x)$  abhängt. Deshalb wird vereinfachend  $q_j = 0$  gesetzt. Dann gilt

$$A(m+1) e^{\iota j\theta} = \frac{\omega}{2} \left( A(m) \underbrace{(e^{\iota(j+1)\theta} + e^{\iota(j-1)\theta})}_{2e^{\iota j\theta} \cos \theta} \right) + (1-\omega)A(m)e^{\iota j\theta}.$$

Aus der Eulerformel  $e^{\iota\theta} = \cos(\theta) + \iota \sin(\theta)$  folgt

$$A(m+1) e^{\iota j\theta} = A(m) \underbrace{(1 - \omega(1 - \cos \theta))}_{2 \sin^2(\theta/2)} e^{\iota j\theta}.$$

Kürzen und Ausnutzen der trigonometrischen Identitäten ergibt:

$$A(m+1) = \left(1 - 2\omega \sin^2\left(\frac{\theta}{2}\right)\right) A(m) \equiv G(\theta) A(m), \quad -\pi < \theta \leq \pi.$$

Den Verstärkungsfaktor, der hier auftritt, kennen wir schon aus Abschnitt 11.3.2. Wir müssen nur  $\theta = \theta_k := \frac{k\pi}{n}$  setzen und  $\omega = 2/3$ . Dann ergibt sich wie in Abschnitt 11.3.2

$$\mu = G\left(\frac{\pi}{2}\right) = |G(\pm\pi)| = \frac{1}{3}.$$

Eine entsprechende Rechnung kann für das Gauß-Seidel-Verfahren durchgeführt werden. Die recht komplizierte Rechnung soll hier weggelassen werden, siehe etwa [Wes 04, Bri 00]. Die Verteilung der Verstärkungsfaktoren  $G(\theta)$  ist in Abb. 11.14 zu sehen. Es ergibt sich als Glättungsfaktor

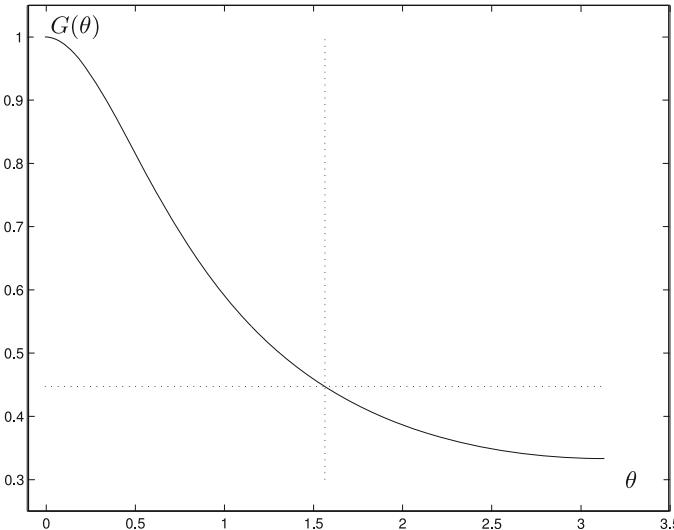


Abb. 11.14 Der Verstärkungsfaktor für das Gauß-Seidelsche Einzelschrittverfahren angewendet auf  $-u'' = f$ .  $G(\theta)$  ist symmetrisch um  $\theta = 0$ . Der Glättungsfaktor ist  $\mu = G(\pi/2) \approx 0.45$ .

$$\mu = \left|G\left(\frac{\pi}{2}\right)\right| = \frac{1}{\sqrt{5}} \approx 0.45.$$

△

### Mehrdimensionale Fourier-Analyse

Die lokale Fourier-Analyse lässt sich leicht auf zwei oder mehr Dimensionen erweitern. In zwei Dimensionen haben die Fourier-Terme die Form

$$e_{jk}^{(m)} = A(m) e^{\iota(\theta_1 + \theta_2)}, \quad (11.76)$$

wo  $-\pi < \theta_1, \theta_2 \leq \pi$  die Wellenzahlen in  $x$ - bzw.  $y$ -Richtung sind. Setzen wir das in die Fehlerfortpflanzung ein, so bekommen wir die allgemeine Gleichung

$$A(m+1) = G(\theta_1, \theta_2) A(m).$$

Der Verstärkungsfaktor  $G$  hängt jetzt von zwei Wellenzahlen ab. Der Glättungsfaktor entspricht wieder dem Maximum über die hochfrequenten Terme:

$$\mu = \max_{\pi/2 \leq |\theta_i| \leq \pi} |G(\theta_1, \theta_2)|.$$

Sie gehören zu Wellenzahlen  $\pi/2 \leq |\theta_i| \leq \pi$  mit  $i = 1$  oder  $i = 2$ .

**Beispiel 11.14.** Wir betrachten das zweidimensionale Modellproblem

$$-u_{xx} - u_{yy} = f(x, y)$$

auf einem homogenen Gitter mit der Gitterweite  $h$ . Die gedämpfte Jacobi-Iteration ergibt für die Fehlerfortpflanzung

$$e_{jk}^{(m+1)} = \frac{\omega}{4} \left( e_{j-1,k}^{(m)} + e_{j+1,k}^{(m)} + e_{j,k-1}^{(m)} + e_{j,k+1}^{(m)} \right) + (1 - \omega) e_{jk}^{(m)}. \quad (11.77)$$

Hier setzen wir die Fourier-Terme (11.76) ein und erhalten

$$A(m+1) = \left[ 1 - \omega \left( \sin^2 \left( \frac{\theta_1}{2} \right) + \sin^2 \left( \frac{\theta_2}{2} \right) \right) \right] A(m) \equiv G(\theta_1, \theta_2) A(m).$$

In Abb. 11.15 sehen wir zwei Ansichten des Verstärkungsfaktors für den Fall  $\omega = 4/5$ . Die Funktion ist symmetrisch bez. beider Achsen. Eine einfache Rechnung zeigt, dass  $\omega = 4/5$  der für den Glättungsfaktor optimale Wert ist und dass für diesen gilt

$$\mu(\omega = \frac{4}{5}) = 0.6.$$

Dieser Wert ist auch wieder unabhängig von der Gitterweite  $h$ .

Eine entsprechende Rechnung kann für das Gauß-Seidel-Verfahren durchgeführt werden, der wir nur in groben Schritten folgen wollen. Die Fehlerentwicklung gehorcht der Formel

$$e_{jk}^{(m+1)} = \frac{e_{j-1,k}^{(m+1)} + e_{j+1,k}^{(m)} + e_{j,k-1}^{(m+1)} + e_{j,k+1}^{(m)}}{4}. \quad (11.78)$$

Einsetzen der lokalen Fourier-Terme ergibt den Verstärkungsfaktor in komplexer Darstellung

$$G(\theta_1, \theta_2) = \frac{e^{\iota\theta_1} + e^{\iota\theta_2}}{4 - e^{-\iota\theta_1} + e^{-\iota\theta_2}}. \quad (11.79)$$

$G$  ist in Abb. 11.16 wiedergegeben. Die Untersuchung dieser Funktion ist aufwändig; nach einiger Rechnung findet man

$$|G(\theta_1, \theta_2)|^2 = \frac{1 + \cos \beta}{9 - 8 \cos \left( \frac{\alpha}{2} \right) \cos \left( \frac{\beta}{2} \right) + \cos \beta}, \quad (11.80)$$

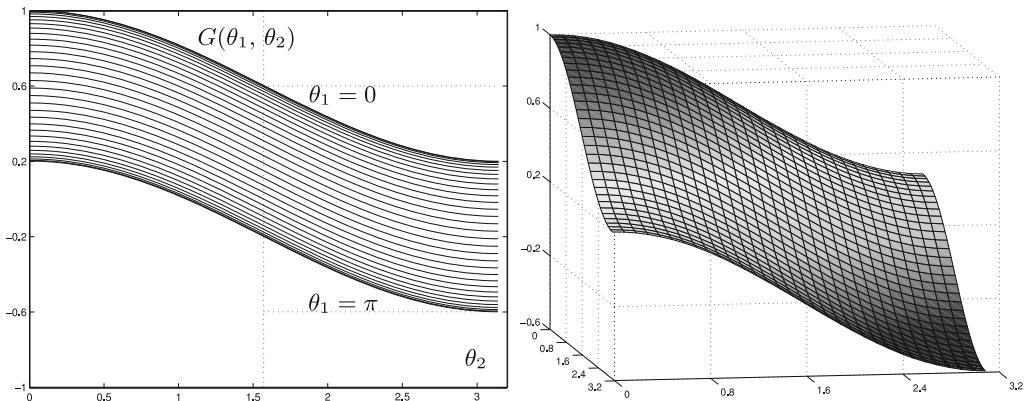


Abb. 11.15 Der Verstärkungsfaktor  $G(\theta_1, \theta_2)$  für  $\omega = 4/5$  links als Funktionenschar  $G(\theta_2)$  mit  $\theta_1 = 0$  oben und  $\theta_1 = \pi$  unten, rechts als Fläche über dem Quadrat  $[-\pi, \pi]^2$ .

mit  $\alpha = \theta_1 + \theta_2$  und  $\beta = \theta_1 - \theta_2$ . Beschränkt man diesen Ausdruck auf die oszillatorischen Frequenzen, so ergibt eine komplizierte Rechnung

$$\mu = G\left(\frac{\pi}{2}, \cos^{-1}\left(\frac{4}{5}\right)\right) = \frac{1}{2}.$$

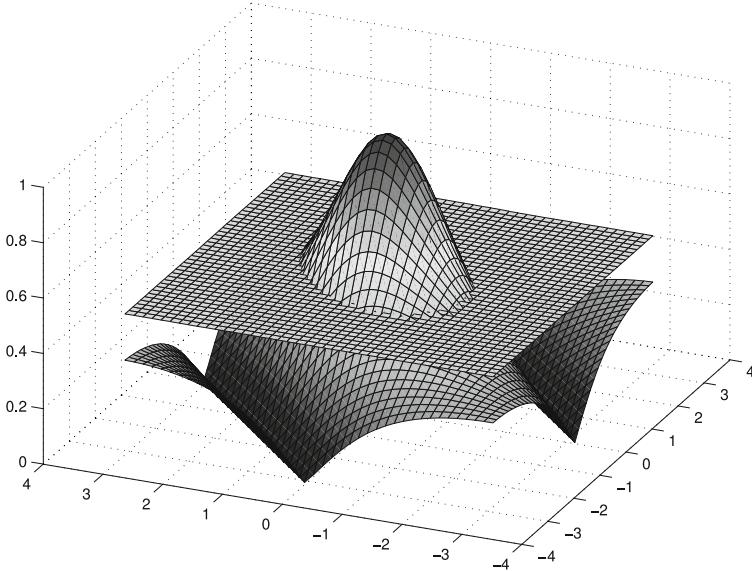


Abb. 11.16 Der Verstärkungsfaktor  $G(\theta_1, \theta_2)$  für das Gauß-Seidel-Verfahren als Fläche über dem Quadrat  $[-\pi, \pi]^2$ , die eingezeichnete Ebene  $G \equiv 1/2$  zeigt anschaulich, dass dies etwa der Glättungsfaktor ist.



**Beispiel 11.15.** Dieser Abschnitt soll abgeschlossen werden mit der Anwendung einer Mehrgittermethode auf das Beispiel 11.7, das schon zum Vergleich der Konvergenzraten verschiedener Relaxationsverfahren diente, siehe Tabelle 11.1. Das Beispiel hat den Vorteil, dass man die exakte Lösung (11.45) kennt und deshalb den Fehler berechnen kann.

In den Ergebnistabellen wird dieser Fehler in der diskreten  $L^2$ -Norm

$$\|e^h\|_h = \left( h^2 \sum_i (e_i^h)^2 \right)^{1/2}$$

aufgeführt. Das hat den Vorteil, dass Fehler- und Residuums-Normen für verschiedene Werte von  $h$  vergleichbar werden. Da der exakte Fehler in praktischen Anwendungen normalerweise nicht zur Verfügung steht, wird daneben auch die diskrete  $L^2$ -Norm des Residuums verzeichnet. Die Zahlen sind dem Tutorium von Briggs, Henson und McCormick [Bri 00] entnommen, dem wir auch andere Ideen und Anregungen zu diesem Abschnitt verdanken.

Gerechnet wird auf einem Schachbrett-Gitter (engl. *Red-Black-Gauß-Seidel*) mit  $N^2$  Punkten für  $N = 15, 31, 63$  und  $127$ ; das sind vier verschiedene Rechnungen. Als Mehrgittermethode wird ein  $V$ -Zyklus benutzt mit der Gauß-Seidel-Relaxation, der FW-Restriktion und linearer Interpolation, dessen größtes Gitter nur einen inneren Punkt hat. Tab. 11.4 zeigt Residuum und Fehler nach jedem  $V$ -Zyklus, außerdem in den Ratio-Spalten das Verhältnis der Normen von Residuum und Fehler zwischen aufeinander folgenden  $V$ -Zyklen.

Tab. 11.4  $V$ -Zyklus mit  $\nu_1 = 2$  und  $\nu_2 = 1$  auf einer Schachbrett-Diskretisierung. Gauß-Seidel-Relaxation, FW-Restriktion und lineare Interpolation.

V-Zyklus	$N = 15$				$N = 31$			
	$\ r^h\ _h$	Ratio	$\ e\ _h$	Ratio	$\ r^h\ _h$	Ratio	$\ e\ _h$	Ratio
0	$6.75 \cdot 10^2$		$5.45 \cdot 10^{-1}$		$2.60 \cdot 10^3$		$5.61 \cdot 10^{-1}$	
1	$4.01 \cdot 10^0$	0.01	$1.05 \cdot 10^{-2}$	0.02	$1.97 \cdot 10^1$	0.01	$1.38 \cdot 10^{-2}$	0.02
2	$1.11 \cdot 10^{-1}$	0.03	$4.10 \cdot 10^{-4}$	0.04	$5.32 \cdot 10^{-1}$	0.03	$6.32 \cdot 10^{-4}$	0.05
3	$3.96 \cdot 10^{-3}$	0.04	$1.05 \cdot 10^{-4}$	0.26	$2.06 \cdot 10^{-2}$	0.04	$4.41 \cdot 10^{-5}$	0.07
4	$1.63 \cdot 10^{-4}$	0.04	$1.03 \cdot 10^{-4}$	0.98*	$9.79 \cdot 10^{-4}$	0.05	$2.59 \cdot 10^{-5}$	0.59
5	$7.45 \cdot 10^{-6}$	0.05	$1.03 \cdot 10^{-4}$	1.00*	$5.20 \cdot 10^{-5}$	0.05	$2.58 \cdot 10^{-5}$	1.00*
6	$3.75 \cdot 10^{-7}$	0.05	$1.03 \cdot 10^{-4}$	1.00*	$2.96 \cdot 10^{-6}$	0.06	$2.58 \cdot 10^{-5}$	1.00*
7	$2.08 \cdot 10^{-8}$	0.06	$1.03 \cdot 10^{-4}$	1.00*	$1.77 \cdot 10^{-7}$	0.06	$2.58 \cdot 10^{-5}$	1.00*

V-Zyklus	$N = 63$				$N = 127$			
	$\ r^h\ _h$	Ratio	$\ e\ _h$	Ratio	$\ r^h\ _h$	Ratio	$\ e\ _h$	Ratio
0	$1.06 \cdot 10^4$		$5.72 \cdot 10^{-1}$		$4.16 \cdot 10^4$		$5.74 \cdot 10^{-1}$	
1	$7.56 \cdot 10^1$	0.01	$1.39 \cdot 10^{-2}$	0.02	$2.97 \cdot 10^2$	0.01	$1.39 \cdot 10^{-2}$	0.02
2	$2.07 \cdot 10^0$	0.03	$6.87 \cdot 10^{-4}$	0.05	$8.25 \cdot 10^0$	0.03	$6.92 \cdot 10^{-4}$	0.05
3	$8.30 \cdot 10^{-2}$	0.04	$4.21 \cdot 10^{-5}$	0.06	$3.37 \cdot 10^{-1}$	0.04	$4.22 \cdot 10^{-5}$	0.06
4	$4.10 \cdot 10^{-3}$	0.05	$7.05 \cdot 10^{-6}$	0.17	$1.65 \cdot 10^{-2}$	0.05	$3.28 \cdot 10^{-6}$	0.08
5	$2.29 \cdot 10^{-4}$	0.06	$6.45 \cdot 10^{-6}$	0.91*	$8.99 \cdot 10^{-4}$	0.05	$1.63 \cdot 10^{-6}$	0.50
6	$1.39 \cdot 10^{-5}$	0.06	$6.44 \cdot 10^{-6}$	1.00*	$5.29 \cdot 10^{-5}$	0.06	$1.61 \cdot 10^{-6}$	0.99*
7	$8.92 \cdot 10^{-7}$	0.06	$6.44 \cdot 10^{-6}$	1.00*	$3.29 \cdot 10^{-6}$	0.06	$1.61 \cdot 10^{-6}$	1.00*

Für jedes der vier Gitter fällt die Norm des Fehlers von Zyklus zu Zyklus rasch, bevor Sie bei einer gewissen Größenordnung stehen bleibt. Dies ist die Größenordnung des Diskretisierungsfehlers  $O(h^2)$ ; er vermindert sich erwartungsgemäß etwa um den Faktor vier, wenn  $h$  halbiert wird. Danach machen die Verhältniszahlen keinen Sinn mehr und werden deshalb mit einem \* versehen.

Die Normen des Residuums fallen auch recht schnell. Dabei erreicht ihr Verhältnis einen nahezu konstanten Wert. Dieser Wert (ungefähr 0.07) ist eine gute Schätzung für den asymptotischen Konvergenzfaktor. Nach 12 bis 14 V-Zyklen erreicht die Näherungslösung des linearen Gleichungssystems eine Genauigkeit nahe der Maschinengenauigkeit.  $\triangle$

## 11.4 Methode der konjugierten Gradienten (CG-Verfahren)

Im Folgenden befassen wir uns mit der iterativen Lösung von linearen Gleichungssystemen  $\mathbf{A}\mathbf{x} = \mathbf{b}$  mit symmetrischer und positiv definiter Matrix  $\mathbf{A} \in \mathbb{R}^{n,n}$ . Solche Gleichungssysteme treten auf im Zusammenhang mit Differenzenverfahren und mit der Methode der finiten Elemente zur Behandlung von elliptischen Randwertaufgaben.

### 11.4.1 Herleitung des Algorithmus

Als Grundlage zur Begründung des iterativen Verfahrens zur Lösung von symmetrischen positiv definiten Gleichungssystemen dient der

**Satz 11.14.** *Die Lösung  $\mathbf{x}$  von  $\mathbf{A}\mathbf{x} = \mathbf{b}$  mit symmetrischer und positiv definiter Matrix  $\mathbf{A} \in \mathbb{R}^{n,n}$  ist das Minimum der quadratischen Funktion*

$$F(\mathbf{v}) := \frac{1}{2} \sum_{i=1}^n \sum_{k=1}^n a_{ik} v_k v_i - \sum_{i=1}^n b_i v_i = \frac{1}{2} (\mathbf{v}, \mathbf{Av}) - (\mathbf{b}, \mathbf{v}). \quad (11.81)$$

*Beweis.* Die  $i$ -te Komponente des Gradienten von  $F(\mathbf{v})$  ist

$$\frac{\partial F}{\partial v_i} = \sum_{k=1}^n a_{ik} v_k - b_i, \quad i = 1, 2, \dots, n, \quad (11.82)$$

und deshalb ist der Gradient

$$\text{grad } F(\mathbf{v}) = \mathbf{Av} - \mathbf{b} = \mathbf{r} \quad (11.83)$$

gleich dem Residuenvektor  $\mathbf{r}$  zum Vektor  $\mathbf{v}$ .

Für die Lösung  $\mathbf{x}$  ist mit  $\text{grad } F(\mathbf{x}) = \mathbf{0}$  die notwendige Bedingung für ein Extremum erfüllt. Überdies ist die Hessesche Matrix  $\mathbf{H}$  von  $F(\mathbf{v})$  gleich der Matrix  $\mathbf{A}$  und somit positiv definit. Das Extremum ist in der Tat ein Minimum.

Umgekehrt ist jedes Minimum von  $F(\mathbf{v})$  Lösung des Gleichungssystems, denn wegen der stetigen Differenzierbarkeit der Funktion  $F(\mathbf{v})$  muss  $\text{grad } F(\mathbf{v}) = \mathbf{A}\mathbf{v} - \mathbf{b} = \mathbf{0}$  gelten, d.h.  $\mathbf{v}$  muss gleich der eindeutigen Lösung  $\mathbf{x}$  sein.  $\square$

Wegen Satz 11.14 wird die Aufgabe,  $\mathbf{Ax} = \mathbf{b}$  zu lösen, durch die äquivalente Aufgabe ersetzt, das Minimum von  $F(\mathbf{v})$  auf iterativem Weg zu bestimmen. Das Grundprinzip der Methode besteht darin, zu einem gegebenen Näherungsvektor  $\mathbf{v}$  und einem gegebenen, geeignet festzulegenden *Richtungsvektor*  $\mathbf{p} \neq \mathbf{0}$  die Funktion  $F(\mathbf{v})$  in dieser Richtung zu minimieren. Gesucht wird somit ein  $t \in \mathbb{R}$  in

$$\mathbf{v}' := \mathbf{v} + t\mathbf{p} \quad \text{so, dass} \quad F(\mathbf{v}') = F(\mathbf{v} + t\mathbf{p}) = \min! \quad (11.84)$$

Bei festem  $\mathbf{v}$  und  $\mathbf{p}$  stellt dies eine Bedingung an  $t$  dar. Aus

$$\begin{aligned} F^*(t) &:= F(\mathbf{v} + t\mathbf{p}) = \frac{1}{2}(\mathbf{v} + t\mathbf{p}, \mathbf{A}(\mathbf{v} + t\mathbf{p})) - (\mathbf{b}, \mathbf{v} + t\mathbf{p}) \\ &= \frac{1}{2}(\mathbf{v}, \mathbf{A}\mathbf{v}) + t(\mathbf{p}, \mathbf{A}\mathbf{v}) + \frac{1}{2}t^2(\mathbf{p}, \mathbf{A}\mathbf{p}) - (\mathbf{b}, \mathbf{v}) - t(\mathbf{b}, \mathbf{p}) \\ &= \frac{1}{2}t^2(\mathbf{p}, \mathbf{A}\mathbf{p}) + t(\mathbf{p}, \mathbf{r}) + F(\mathbf{v}) \end{aligned}$$

ergibt sich durch Nullsetzen der Ableitung nach  $t$

$$t_{\min} = -\frac{(\mathbf{p}, \mathbf{r})}{(\mathbf{p}, \mathbf{A}\mathbf{p})}, \quad \mathbf{r} = \mathbf{A}\mathbf{v} - \mathbf{b}. \quad (11.85)$$

Da  $\mathbf{A}$  positiv definit ist, ist

$$(\mathbf{p}, \mathbf{A}\mathbf{p})^{1/2} =: \|\mathbf{p}\|_A \quad (11.86)$$

eine Norm, die so genannte *Energienorm*. Deshalb ist mit  $\mathbf{p} \neq \mathbf{0}$  der Nenner in (11.85) eine positive Zahl. Der Parameter  $t_{\min}$  liefert tatsächlich ein Minimum von  $F(\mathbf{v})$  in Richtung  $\mathbf{p}$ , weil die zweite Ableitung von  $F^*(t)$  nach  $t$  positiv ist. Die Abnahme des Funktionswertes von  $F(\mathbf{v})$  zu  $F(\mathbf{v}')$  im *Minimalpunkt*  $\mathbf{v}'$  ist maximal, weil der Graph von  $F^*(t)$  eine sich nach oben öffnende Parabel ist. Der Richtungsvektor  $\mathbf{p}$  darf aber nicht orthogonal zum Residuenvektor  $\mathbf{r}$  sein, da andernfalls wegen  $t_{\min} = 0$  dann  $\mathbf{v}' = \mathbf{v}$  gilt.

**Satz 11.15.** *Im Minimalpunkt  $\mathbf{v}'$  ist der Residuenvektor  $\mathbf{r}' = \mathbf{A}\mathbf{v}' - \mathbf{b}$  orthogonal zum Richtungsvektor  $\mathbf{p}$ .*

*Beweis.* Wegen(11.85) gilt

$$\begin{aligned} (\mathbf{p}, \mathbf{r}') &= (\mathbf{p}, \mathbf{A}\mathbf{v}' - \mathbf{b}) = (\mathbf{p}, \mathbf{A}(\mathbf{v} + t_{\min}\mathbf{p}) - \mathbf{b}) \\ &= (\mathbf{p}, \mathbf{r} + t_{\min}\mathbf{A}\mathbf{p}) = (\mathbf{p}, \mathbf{r}) + t_{\min}(\mathbf{p}, \mathbf{A}\mathbf{p}) = 0. \end{aligned}$$

$\square$

Ein Iterationsschritt zur Verkleinerung des Wertes  $F(\mathbf{v})$  besitzt für  $n = 2$  folgende geometrische Interpretation, welche später die Motivation für den Algorithmus liefert. Die Niveaulinien  $F(\mathbf{v}) = \text{const}$  sind konzentrische Ellipsen, deren gemeinsamer Mittelpunkt gleich dem Minimumspunkt  $\mathbf{x}$  ist. Im gegebenen Punkt  $\mathbf{v}$  steht der Residuenvektor  $\mathbf{r}$  senkrecht zur

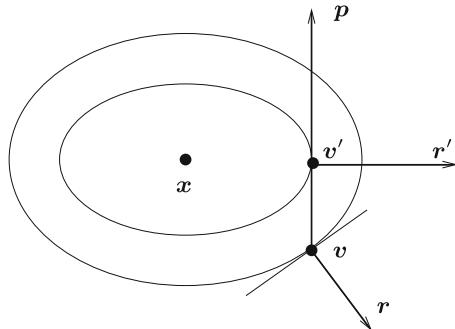


Abb. 11.17

Geometrische Interpretation eines Iterations schrittes.

Niveaulinie durch den Punkt  $v$ . Mit dem Richtungsvektor  $p$  wird derjenige Punkt  $v'$  ermittelt, für den  $F(v')$  minimal ist. Da dort nach Satz 11.15 der Residuenvektor  $r'$  orthogonal zu  $p$  ist, ist  $p$  Tangente an die Niveaulinie durch  $v'$  (vgl. Abb. 11.17).

Es ist nahe liegend als Richtungsvektors  $p$  den negativen Gradienten  $p = -\text{grad } F(v) = -(Av - b) = -r$  zu wählen. Dies führt auf die *Methode des steilsten Abstiegs*, auf die wir aber nicht weiter eingehen werden. Denn dieses Vorgehen erweist sich oft als nicht besonders vorteilhaft, obwohl in jedem Iterationsschritt diejenige Richtung gewählt wird, welche *lokal* die stärkste Abnahme der Funktion  $F(v)$  garantiert. Sind nämlich im Fall  $n = 2$  die Ellipsen sehr langgestreckt, entsprechend einer großen Konditionszahl von  $A$ , werden viele Schritte benötigt, um in die Nähe des Minimumspunktes  $x$  zu gelangen.

In der *Methode der konjugierten Gradienten* von Hestenes und Stiefel [Hes 52] wird von der geometrischen Tatsache Gebrauch gemacht, dass diejenige Richtung  $p$ , welche vom Punkt  $v$  den Mittelpunkt  $x$  der Ellipsen trifft, mit der Tangentenrichtung im Punkt  $v$  im Sinn der Kegelschnittgleichungen konjugiert ist. Mit dieser Wahl würde man im Fall  $n = 2$  die Lösung  $x$  unmittelbar finden.

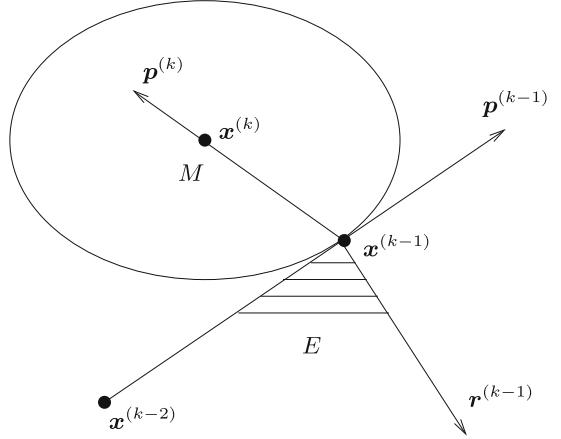
**Definition 11.16.** Zwei Vektoren  $p, q \in \mathbb{R}^n$  heißen *konjugiert* oder *A-orthogonal*, falls für die positiv definite Matrix  $A$  gilt

$$(p, Aq) = 0. \quad (11.87)$$

Ausgehend von einem Startvektor  $x^{(0)}$  wird im ersten Schritt der Richtungsvektor  $p^{(1)}$  durch den negativen Residuenvektor festgelegt und der Minimalpunkt  $x^{(1)}$  bestimmt. Mit (11.85) lautet dieser Schritt

$$\begin{aligned} p^{(1)} &= -r^{(0)} = -(Ax^{(0)} - b), \\ q_1 &:= \frac{(r^{(0)}, r^{(0)})}{(p^{(1)}, Ap^{(1)})}, \quad x^{(1)} = x^{(0)} + q_1 p^{(1)}. \end{aligned} \quad (11.88)$$

Im allgemeinen  $k$ -ten Schritt betrachtet man die zweidimensionale Ebene  $E$  des  $\mathbb{R}^n$  durch den Iterationspunkt  $x^{(k-1)}$ , welche aufgespannt wird vom vorhergehenden Richtungsvektor  $p^{(k-1)}$  und dem nach Satz 11.15 dazu orthogonalen Residuenvektor  $r^{(k-1)}$ . Der Schnitt der Ebene  $E$  mit der Niveaumenge  $F(v) = F(x^{(k-1)})$  ist eine Ellipse (vgl. Abb. 11.18). Der Richtungsvektor  $p^{(k-1)}$  von  $x^{(k-2)}$  durch  $x^{(k-1)}$  ist Tangente an diese Ellipse, weil

Abb. 11.18 Wahl des Richtungsvektors  $p^{(k)}$ .

$x^{(k-1)}$  Minimalpunkt ist. Das Ziel des  $k$ -ten Iterationsschrittes besteht darin, das Minimum von  $F(\mathbf{v})$  bezüglich der Ebene  $E$  zu ermitteln, welches im Mittelpunkt der Schnittellipse angenommen wird. Der Richtungsvektor  $\mathbf{p}^{(k)}$  muss somit konjugiert sein zu  $\mathbf{p}^{(k-1)}$  bezüglich der Schnittellipse und damit auch bezüglich des Ellipsoids  $F(\mathbf{v}) = F(\mathbf{x}^{(k-1)})$ .

Im zweckmäßigen Ansatz für den Richtungsvektor

$$\mathbf{p}^{(k)} = -\mathbf{r}^{(k-1)} + e_{k-1} \mathbf{p}^{(k-1)} \quad (11.89)$$

bestimmt sich  $e_{k-1}$  aus der Bedingung  $(\mathbf{p}^{(k)}, \mathbf{A}\mathbf{p}^{(k-1)}) = 0$  zu

$$e_{k-1} = \frac{(\mathbf{r}^{(k-1)}, \mathbf{A}\mathbf{p}^{(k-1)})}{(\mathbf{p}^{(k-1)}, \mathbf{A}\mathbf{p}^{(k-1)})}. \quad (11.90)$$

Mit dem so festgelegten Richtungsvektor  $\mathbf{p}^{(k)}$  erhält man den iterierten Vektor  $\mathbf{x}^{(k)}$  als Minimalpunkt gemäß

$$\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} + q_k \mathbf{p}^{(k)} \quad \text{mit } q_k = -\frac{(\mathbf{r}^{(k-1)}, \mathbf{p}^{(k)})}{(\mathbf{p}^{(k)}, \mathbf{A}\mathbf{p}^{(k)})}. \quad (11.91)$$

Die Nenner von  $e_{k-1}$  und von  $q_k$  sind positiv, falls  $\mathbf{p}^{(k-1)}$  bzw.  $\mathbf{p}^{(k)}$  von null verschieden sind. Dies trifft dann zu, falls  $\mathbf{r}^{(k-2)}$  und  $\mathbf{r}^{(k-1)}$  ungleich null sind, weil dann  $\mathbf{p}^{(k-1)} \neq \mathbf{0}$  und  $\mathbf{p}^{(k)} \neq \mathbf{0}$  wegen (11.89) gelten, d.h. solange  $\mathbf{x}^{(k-2)} \neq \mathbf{x}$  und  $\mathbf{x}^{(k-1)} \neq \mathbf{x}$  sind. Der Residuenvektor  $\mathbf{r}^{(k)}$  zu  $\mathbf{x}^{(k)}$  ist rekursiv berechenbar gemäß

$$\mathbf{r}^{(k)} = \mathbf{A}\mathbf{x}^{(k)} - \mathbf{b} = \mathbf{A}(\mathbf{x}^{(k-1)} + q_k \mathbf{p}^{(k)}) - \mathbf{b} = \mathbf{r}^{(k-1)} + q_k \mathbf{A}\mathbf{p}^{(k)}. \quad (11.92)$$

Die Methode der konjugierten Gradienten ist damit in ihren Grundzügen bereits vollständig beschrieben. Die Darstellung der beiden Skalare  $e_{k-1}$  (11.90) und  $q_k$  (11.91) kann noch vereinfacht werden. Dazu ist zu berücksichtigen, dass nach Satz 11.15 der Residuenvektor  $\mathbf{r}^{(k)}$  orthogonal zu  $\mathbf{p}^{(k)}$  ist, aber auch orthogonal zur Ebene  $E$  ist, weil  $\mathbf{x}^{(k)}$  darin Minimalpunkt ist, und folglich gelten die Orthogonalitätsrelationen

$$(\mathbf{r}^{(k)}, \mathbf{p}^{(k)}) = 0, \quad (\mathbf{r}^{(k)}, \mathbf{r}^{(k-1)}) = 0, \quad (\mathbf{r}^{(k)}, \mathbf{p}^{(k-1)}) = 0. \quad (11.93)$$

Für den Zähler von  $q_k$  ergibt sich deshalb

$$(\mathbf{r}^{(k-1)}, \mathbf{p}^{(k)}) = (\mathbf{r}^{(k-1)}, -\mathbf{r}^{(k-1)} + e_{k-1} \mathbf{p}^{(k-1)}) = -(\mathbf{r}^{(k-1)}, \mathbf{r}^{(k-1)}),$$

und somit

$$q_k = \frac{(\mathbf{r}^{(k-1)}, \mathbf{r}^{(k-1)})}{(\mathbf{p}^{(k)}, \mathbf{A}\mathbf{p}^{(k)})}. \quad (11.94)$$

Wegen (11.94) ist sichergestellt, dass  $q_k > 0$  gilt, falls  $\mathbf{r}^{(k-1)} \neq \mathbf{0}$ , d.h.  $\mathbf{x}^{(k-1)} \neq \mathbf{x}$  ist.

Für den Zähler von  $e_{k-1}$  erhalten wir wegen (11.92) für  $k-1$  anstelle von  $k$

$$\begin{aligned} \mathbf{A}\mathbf{p}^{(k-1)} &= (\mathbf{r}^{(k-1)} - \mathbf{r}^{(k-2)})/q_{k-1}, \\ (\mathbf{r}^{(k-1)}, \mathbf{A}\mathbf{p}^{(k-1)}) &= (\mathbf{r}^{(k-1)}, \mathbf{r}^{(k-1)})/q_{k-1}. \end{aligned}$$

Verwendet man für  $q_{k-1}$  den entsprechenden Ausdruck (11.94), so ergibt sich aus (11.90)

$$e_{k-1} = \frac{(\mathbf{r}^{(k-1)}, \mathbf{r}^{(k-1)})}{(\mathbf{r}^{(k-2)}, \mathbf{r}^{(k-2)})}. \quad (11.95)$$

Mit den neuen Darstellungen wird eine Reduktion des Rechenaufwandes erzielt. Der CG-Algorithmus lautet damit:

$$\begin{aligned} &\text{Start: Wahl von } \mathbf{x}^{(0)}; \mathbf{r}^{(0)} = \mathbf{A}\mathbf{x}^{(0)} - \mathbf{b}; \mathbf{p}^{(1)} = -\mathbf{r}^{(0)}; \\ &\text{Iteration } k = 1, 2, 3, \dots : \\ &\quad \text{Falls } k > 1 : \begin{cases} e_{k-1} &= (\mathbf{r}^{(k-1)}, \mathbf{r}^{(k-1)})/(\mathbf{r}^{(k-2)}, \mathbf{r}^{(k-2)}) \\ \mathbf{p}^{(k)} &= -\mathbf{r}^{(k-1)} + e_{k-1} \mathbf{p}^{(k-1)} \end{cases} \\ &\quad \mathbf{z} = \mathbf{A}\mathbf{p}^{(k)} \\ &\quad q_k = (\mathbf{r}^{(k-1)}, \mathbf{r}^{(k-1)})/(\mathbf{p}^{(k)}, \mathbf{z}) \\ &\quad \mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} + q_k \mathbf{p}^{(k)}; \quad \mathbf{r}^{(k)} = \mathbf{r}^{(k-1)} + q_k \mathbf{z} \\ &\quad \text{Test auf Konvergenz} \end{aligned} \quad (11.96)$$

Der Rechenaufwand für einen typischen Iterationsschritt setzt sich zusammen aus einer Matrix-Vektor-Multiplikation  $\mathbf{z} = \mathbf{A}\mathbf{p}$ , bei der die schwache Besetzung von  $\mathbf{A}$  ausgenützt werden kann, aus zwei Skalarprodukten und drei skalaren Multiplikationen von Vektoren. Sind  $\gamma n$  Matrixelemente von  $\mathbf{A}$  ungleich null, wobei  $\gamma \ll n$  gilt, beträgt der Rechenaufwand pro CG-Schritt etwa

$$Z_{CGS} = (\gamma + 5)n \quad (11.97)$$

multiplikative Operationen. Der Speicherbedarf beträgt neben der Matrix  $\mathbf{A}$  nur rund  $4n$  Plätze, da für  $\mathbf{p}^{(k)}, \mathbf{r}^{(k)}$  und  $\mathbf{x}^{(k)}$  offensichtlich nur je ein Vektor benötigt wird, und dann noch der Hilfsvektor  $\mathbf{z}$  auftritt.

### 11.4.2 Eigenschaften der Methode der konjugierten Gradienten

Wir stellen die wichtigsten Eigenschaften des CG-Algorithmus (11.96) zusammen, welche anschließend die Grundlage dazu bilden werden, über das Konvergenzverhalten Aussagen zu machen.

**Satz 11.17.** Die Residuenvektoren  $\mathbf{r}^{(k)}$  bilden ein Orthogonalsystem, und die Richtungsvektoren  $\mathbf{p}^{(k)}$  sind paarweise konjugiert. Für  $k \geq 2$  gelten

$$(\mathbf{r}^{(k-1)}, \mathbf{r}^{(j)}) = 0, \quad j = 0, 1, \dots, k-2; \quad (11.98)$$

$$(\mathbf{r}^{(k-1)}, \mathbf{p}^{(j)}) = 0, \quad j = 1, 2, \dots, k-1; \quad (11.99)$$

$$(\mathbf{p}^{(k)}, \mathbf{A}\mathbf{p}^{(j)}) = 0, \quad j = 1, 2, \dots, k-1. \quad (11.100)$$

*Beweis.* Die Aussagen werden durch vollständige Induktion nach  $k$  gezeigt.

*Induktionsbeginn.* Für  $k = 2$  ist  $(\mathbf{r}^{(1)}, \mathbf{p}^{(1)}) = 0$  wegen  $\mathbf{r}^{(0)} = -\mathbf{p}^{(1)}$  und Satz 11.15. Damit sind (11.98) und (11.99) richtig. (11.100) gilt für  $k = 2$  nach Konstruktion von  $\mathbf{p}^{(2)}$ .

*Induktionsvoraussetzung.* (11.98) bis (11.100) sind für ein  $k \geq 2$  richtig.

*Induktionbehauptung.* Die Relationen (11.98) bis (11.100) sind auch für  $k + 1$  richtig.

*Induktionsbeweis.* Um  $(\mathbf{r}^{(k)}, \mathbf{r}^{(j)}) = 0$  für  $j = 0, 1, 2, \dots, k-1$  zu zeigen, wird zuerst  $\mathbf{r}^{(k)}$  auf Grund der Rekursionsformel (11.92) ersetzt und dann wird im zweiten Skalarprodukt  $\mathbf{r}^{(j)} = e_j \mathbf{p}^{(j)} - \mathbf{p}^{(j+1)}$  gesetzt, was mit  $e_0 := 0$  und  $\mathbf{p}^{(0)} := \mathbf{0}$  auch für  $j = 0$  richtig bleibt:

$$\begin{aligned} (\mathbf{r}^{(k)}, \mathbf{r}^{(j)}) &= (\mathbf{r}^{(k-1)}, \mathbf{r}^{(j)}) + q_k (\mathbf{A}\mathbf{p}^{(k)}, \mathbf{r}^{(j)}) \\ &= (\mathbf{r}^{(k-1)}, \mathbf{r}^{(j)}) + q_k e_j (\mathbf{A}\mathbf{p}^{(k)}, \mathbf{p}^{(j)}) - q_k (\mathbf{A}\mathbf{p}^{(k)}, \mathbf{p}^{(j+1)}) \end{aligned}$$

Wegen  $(\mathbf{A}\mathbf{p}^{(k)}, \mathbf{p}^{(j)}) = (\mathbf{p}^{(k)}, \mathbf{A}\mathbf{p}^{(j)})$  sind nach Induktionsvoraussetzung alle drei Skalarprodukte für  $j = 0, 1, \dots, k-2$  gleich null. Für  $j = k-1$  ist das mittlere Skalarprodukt gleich null, und die verbleibenden ergeben null wegen (11.94).

Analog folgt

$$(\mathbf{r}^{(k)}, \mathbf{p}^{(j)}) = (\mathbf{r}^{(k-1)}, \mathbf{p}^{(j)}) + q_k (\mathbf{p}^{(k)}, \mathbf{A}\mathbf{p}^{(j)}) = 0,$$

denn nach Induktionsvoraussetzung sind beide Skalarprodukte für  $j = 1, \dots, k-1$  gleich null. Für  $j = k$  ist  $(\mathbf{r}^{(k)}, \mathbf{p}^{(k)}) = 0$  wegen (11.93).

Für den Nachweis von  $(\mathbf{p}^{(k+1)}, \mathbf{A}\mathbf{p}^{(j)}) = 0$  können wir uns auf  $j = 1, 2, \dots, k-1$  beschränken, weil  $\mathbf{p}^{(k+1)}$  und  $\mathbf{p}^{(k)}$  nach Konstruktion konjugiert sind. Wegen (11.89) und dann (11.92) gilt

$$\begin{aligned} (\mathbf{p}^{(k+1)}, \mathbf{A}\mathbf{p}^{(j)}) &= -(\mathbf{r}^{(k)}, \mathbf{A}\mathbf{p}^{(j)}) + e_k (\mathbf{p}^{(k)}, \mathbf{A}\mathbf{p}^{(j)}) \\ &= -[(\mathbf{r}^{(k)}, \mathbf{r}^{(j)}) - (\mathbf{r}^{(k)}, \mathbf{r}^{(j-1)})]/q_j + e_k (\mathbf{p}^{(k)}, \mathbf{A}\mathbf{p}^{(j)}). \end{aligned}$$

Die Skalarprodukte sind auf Grund des ersten Teils des Induktionsbeweises oder der Induktionsvoraussetzung gleich null.  $\square$

Eine unmittelbare Folge von Satz 11.17 ist der

**Satz 11.18.** Die Methode der konjugierten Gradienten liefert die Lösung eines Gleichungssystems in  $n$  Unbekannten nach höchstens  $n$  Schritten.

*Beweis.* Da die Residuenvektoren  $\mathbf{r}^{(0)}, \mathbf{r}^{(1)}, \dots, \mathbf{r}^{(k)}$  im  $\mathbb{R}^n$  ein Orthogonalsystem bilden, kann es höchstens  $n$  von null verschiedene Vektoren enthalten, und es muss spätestens  $\mathbf{r}^{(n)} = \mathbf{0}$  und deshalb  $\mathbf{x}^{(n)} = \mathbf{x}$  sein.  $\square$

Theoretisch ist der iterativ konzipierte CG-Algorithmus ein endlicher Prozess. Numerisch werden die Orthogonalitätsrelationen (11.98) nicht exakt erfüllt sein. Deshalb ist eine Fortsetzung des Verfahrens über die  $n$  Schritte hinaus sinnvoll, weil die Funktion  $F(\mathbf{v})$  stets verkleinert wird. Andererseits ist aber zu hoffen, dass insbesondere bei sehr großen Gleichungssystemen bedeutend weniger als  $n$  Schritte nötig sein werden, um eine Näherung der Lösung  $\mathbf{x}$  mit genügend kleinem Fehler zu produzieren.

Als nächstes zeigen wir eine Optimalitätseigenschaft der  $k$ -ten Iterierten  $\mathbf{x}^{(k)}$  des CG-Verfahrens. Auf Grund der Rekursionsformel (11.91) besitzt sie die Darstellung

$$\mathbf{x}^{(k)} = \mathbf{x}^{(0)} + \sum_{i=1}^k q_i \mathbf{p}^{(i)}, \quad k = 1, 2, 3, \dots \quad (11.101)$$

In jedem einzelnen CG-Schritt wird die Funktion  $F(\mathbf{v})$ , ausgehend von  $\mathbf{x}^{(k-1)}$ , nur in Richtung von  $\mathbf{p}^{(k)}$  lokal minimiert. Wir wollen zeigen, dass der erreichte Wert  $F(\mathbf{x}^{(k)})$  gleich dem globalen Minimum von  $F(\mathbf{v})$  bezüglich des Unterraums ist, der von den  $k$  Richtungsvektoren aufgespannt ist.

**Satz 11.19.** Die  $k$ -te Iterierte  $\mathbf{x}^{(k)}$  der Methode der konjugierten Gradienten (11.96) minimiert die Funktion  $F(\mathbf{v})$  in Bezug auf den Unterraum  $S_k := \text{span}\{\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \dots, \mathbf{p}^{(k)}\}$ , denn es gilt

$$F(\mathbf{x}^{(k)}) = \min_{c_1, \dots, c_k} F\left(\mathbf{x}^{(0)} + \sum_{i=1}^k c_i \mathbf{p}^{(i)}\right). \quad (11.102)$$

*Beweis.* Es ist zu zeigen, dass die Koeffizienten  $c_i$  in (11.102), welche die Funktion minimieren, mit den Werten der  $q_i$  in (11.94) identisch sind. Wegen (11.100) gilt

$$\begin{aligned} F\left(\mathbf{x}^{(0)} + \sum_{i=1}^k c_i \mathbf{p}^{(i)}\right) &= \frac{1}{2} \left( \mathbf{x}^{(0)} + \sum_{i=1}^k c_i \mathbf{p}^{(i)}, \mathbf{A} \left( \mathbf{x}^{(0)} + \sum_{j=1}^k c_j \mathbf{p}^{(j)} \right) \right) \\ &\quad - \left( \mathbf{b}, \mathbf{x}^{(0)} + \sum_{i=1}^k c_i \mathbf{p}^{(i)} \right) \\ &= \frac{1}{2} \sum_{i=1}^k \sum_{j=1}^k c_i c_j (\mathbf{p}^{(i)}, \mathbf{A} \mathbf{p}^{(j)}) + \sum_{i=1}^k c_i (\mathbf{p}^{(i)}, \mathbf{A} \mathbf{x}^{(0)}) \\ &\quad + \frac{1}{2} (\mathbf{x}^{(0)}, \mathbf{A} \mathbf{x}^{(0)}) - \sum_{i=1}^k c_i (\mathbf{p}^{(i)}, \mathbf{b}) - (\mathbf{b}, \mathbf{x}^{(0)}) \\ &= \frac{1}{2} \sum_{i=1}^k c_i^2 (\mathbf{p}^{(i)}, \mathbf{A} \mathbf{p}^{(i)}) + \sum_{i=1}^k c_i (\mathbf{p}^{(i)}, \mathbf{r}^{(0)}) + F(\mathbf{x}^{(0)}). \end{aligned}$$

Die notwendigen Bedingungen für ein Extremum von  $F$  sind deshalb

$$\frac{\partial F}{\partial c_i} = c_i(\mathbf{p}^{(i)}, \mathbf{A}\mathbf{p}^{(i)}) + (\mathbf{p}^{(i)}, \mathbf{r}^{(0)}) = 0, \quad i = 1, 2, \dots, k,$$

also

$$c_i = -(\mathbf{p}^{(i)}, \mathbf{r}^{(0)})/(\mathbf{p}^{(i)}, \mathbf{A}\mathbf{p}^{(i)}), \quad i = 1, 2, \dots, k.$$

Für  $i = 1$  ist  $\mathbf{p}^{(1)} = -\mathbf{r}^{(0)}$ , und somit gilt  $c_1 = q_1$ . Für  $i > 1$  besitzt  $\mathbf{p}^{(i)}$  nach wiederholter Anwendung von (11.89) die Darstellung

$$\mathbf{p}^{(i)} = -\mathbf{r}^{(i-1)} - e_{i-1}\mathbf{r}^{(i-2)} - e_{i-1}e_{i-2}\mathbf{r}^{(i-3)} - \dots - \left( \prod_{j=1}^{i-1} e_j \right) \mathbf{r}^{(0)}. \quad (11.103)$$

Wegen (11.98) und (11.95) folgt daraus

$$-(\mathbf{p}^{(i)}, \mathbf{r}^{(0)}) = e_{i-1}e_{i-2} \dots e_1(\mathbf{r}^{(0)}, \mathbf{r}^{(0)}) = (\mathbf{r}^{(i-1)}, \mathbf{r}^{(i-1)}),$$

und somit  $c_i = q_i$  für  $i = 1, 2, \dots, k$ .  $\square$

Die Unterräume  $S_k = \text{span}\{\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \dots, \mathbf{p}^{(k)}\}$ ,  $k = 1, 2, 3, \dots$ , sind aber identisch mit denjenigen, welche durch die  $k$  ersten Residuenvektoren  $\mathbf{r}^{(0)}, \mathbf{r}^{(1)}, \dots, \mathbf{r}^{(k-1)}$  aufgespannt werden. Die Identität der Unterräume sieht man mit Hilfe einer induktiven Schlussweise wie folgt:

Für  $k = 1$  ist wegen  $\mathbf{p}^{(1)} = -\mathbf{r}^{(0)}$  offensichtlich  $S_1 = \text{span}\{\mathbf{p}^{(1)}\} = \text{span}\{\mathbf{r}^{(0)}\}$ . Nun nehmen wir an, dass  $S_{k-1} = \text{span}\{\mathbf{p}^{(1)}, \dots, \mathbf{p}^{(k-1)}\} = \text{span}\{\mathbf{r}^{(0)}, \dots, \mathbf{r}^{(k-2)}\}$  für ein  $k > 1$ . Wegen (11.103) (mit  $k$  statt  $i$ ) gilt dann aber, weil der Koeffizient von  $\mathbf{r}^{(k-1)}$  gleich  $-1$  ist, dass  $\mathbf{p}^{(k)} \in \text{span}\{\mathbf{r}^{(0)}, \mathbf{r}^{(1)}, \dots, \mathbf{r}^{(k-1)}\} = S_k$ , aber  $\mathbf{p}^{(k)} \notin S_{k-1}$ . Daraus ergibt sich in der Tat für  $k = 1, 2, 3, \dots$

$$S_k = \text{span}\{\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \dots, \mathbf{p}^{(k)}\} = \text{span}\{\mathbf{r}^{(0)}, \mathbf{r}^{(1)}, \dots, \mathbf{r}^{(k-1)}\}, \quad (11.104)$$

und wegen der Orthogonalität (11.98) der Residuenvektoren gilt natürlich für die Dimension der Unterräume  $S_k$

$$\dim(S_k) = k,$$

solange  $\mathbf{x}^{(k-1)} \neq \mathbf{x}$  und somit  $\mathbf{r}^{(k-1)} \neq \mathbf{0}$  ist.

Die Folge der Unterräume  $S_k$  (11.104) ist identisch mit der Folge von *Krylov-Unterräumen*, welche von  $\mathbf{r}^{(0)}$  und der Matrix  $\mathbf{A}$  erzeugt werden gemäß

$$\mathcal{K}^{(k)}(\mathbf{r}^{(0)}, \mathbf{A}) := \text{span}\{\mathbf{r}^{(0)}, \mathbf{A}\mathbf{r}^{(0)}, \mathbf{A}^2\mathbf{r}^{(0)}, \dots, \mathbf{A}^{k-1}\mathbf{r}^{(0)}\}. \quad (11.105)$$

Für  $k = 1$  ist die Aussage trivialerweise richtig. Für  $k = 2$  ist wegen  $\mathbf{r}^{(1)} = \mathbf{r}^{(0)} + q_1 \mathbf{A}\mathbf{p}^{(1)} = \mathbf{r}^{(0)} - q_1 \mathbf{A}\mathbf{r}^{(0)} \in \text{span}\{\mathbf{r}^{(0)}, \mathbf{A}\mathbf{r}^{(0)}\}$ , aber  $\mathbf{r}^{(1)} \notin \text{span}\{\mathbf{r}^{(0)}\}$ , falls  $\mathbf{r}^{(1)} \neq \mathbf{0}$  ist, und folglich

ist  $S_2 = \mathcal{K}^{(2)}(\mathbf{r}^{(0)}, \mathbf{A})$ . Weiter ist offensichtlich  $\mathbf{A}\mathbf{r}^{(1)} = \mathbf{A}\mathbf{r}^{(0)} - q_1\mathbf{A}^2\mathbf{r}^{(0)} \in \mathcal{K}^{(3)}(\mathbf{r}^{(0)}, \mathbf{A})$ . Durch vollständige Induktion nach  $i$  folgt wegen (11.103)

$$\begin{aligned}\mathbf{r}^{(i)} &= \mathbf{r}^{(i-1)} + q_i(\mathbf{A}\mathbf{p}^{(i)}) \\ &= \mathbf{r}^{(i-1)} - q_i \left[ \mathbf{A}\mathbf{r}^{(i-1)} + e_{i-1}\mathbf{A}\mathbf{r}^{(i-2)} + \dots + \left( \prod_{j=1}^{i-1} e_j \right) (\mathbf{A}\mathbf{r}^{(0)}) \right] \\ &\in \mathcal{K}^{(i+1)}(\mathbf{r}^{(0)}, \mathbf{A})\end{aligned}$$

und für  $\mathbf{r}^{(i)} \neq \mathbf{0}$  wegen der Orthogonalität von  $\mathbf{r}^{(i)}$  zu  $S_{i-1}$  und damit auch zu  $\mathcal{K}^{(i)}(\mathbf{r}^{(0)}, \mathbf{A})$  weiter  $\mathbf{r}^{(i)} \notin \mathcal{K}^{(i)}(\mathbf{r}^{(0)}, \mathbf{A})$ . Deshalb gilt allgemein

$$S_k = \text{span}\{\mathbf{r}^{(0)}, \mathbf{r}^{(1)}, \dots, \mathbf{r}^{(k-1)}\} = \mathcal{K}^{(k)}(\mathbf{r}^{(0)}, \mathbf{A}). \quad (11.106)$$

### 11.4.3 Konvergenzabschätzung

Auf Grund der Optimalitätseigenschaft der CG-Methode von Satz 11.19 und der Charakterisierung der Unterräume  $S_k$  als Krylov-Unterräume (11.106) kann der Fehler  $\mathbf{f}^{(k)} := \mathbf{x}^{(k)} - \mathbf{x}$  in einer geeignet gewählten Vektornorm abgeschätzt werden. Dazu verwenden wir die *Energienorm* (11.86). Für einen beliebigen Vektor  $\mathbf{z} \in \mathbb{R}^n$  und die Lösung  $\mathbf{x}$  von  $\mathbf{Ax} = \mathbf{b}$  gilt nun

$$\begin{aligned}\|\mathbf{z} - \mathbf{x}\|_A^2 &= (\mathbf{z} - \mathbf{x}, \mathbf{A}(\mathbf{z} - \mathbf{x})) \\ &= (\mathbf{z}, \mathbf{Az}) - 2(\mathbf{z}, \mathbf{Ax}) + (\mathbf{x}, \mathbf{Ax}) \\ &= (\mathbf{z}, \mathbf{Az}) - 2(\mathbf{z}, \mathbf{b}) + (\mathbf{A}^{-1}\mathbf{b}, \mathbf{b}) \\ &= 2F(\mathbf{z}) + (\mathbf{A}^{-1}\mathbf{b}, \mathbf{b}).\end{aligned} \quad (11.107)$$

Nach Satz 11.19 minimiert die  $k$ -te Iterierte  $\mathbf{x}^{(k)}$  des CG-Verfahrens die Funktion  $F(\mathbf{x}^{(0)} + \mathbf{v})$  für  $\mathbf{v} \in S_k$ . Da sich das Quadrat der Energienorm für  $\mathbf{z} - \mathbf{x}$  nur um eine additive Konstante und um den Faktor 2 von  $F(\mathbf{z})$  unterscheidet, so folgt mit  $\mathbf{z} = \mathbf{x}^{(k)}$ , dass die Iterierte  $\mathbf{x}^{(k)}$  den Fehler  $\mathbf{f}^{(k)}$  in der Energienorm minimiert, und es gilt

$$\|\mathbf{f}^{(k)}\|_A = \|\mathbf{x}^{(k)} - \mathbf{x}\|_A = \min\{\|\mathbf{z} - \mathbf{x}\|_A \mid \mathbf{z} = \mathbf{x}^{(0)} + \mathbf{v}, \mathbf{v} \in S_k\}. \quad (11.108)$$

Für den Residuenvektor  $\mathbf{r}^{(0)}$  des Krylov-Unterraums gilt

$$\mathbf{r}^{(0)} = \mathbf{Ax}^{(0)} - \mathbf{b} = \mathbf{Ax}^{(0)} - \mathbf{Ax} = \mathbf{Af}^{(0)}, \quad (11.109)$$

und für die Differenz  $\mathbf{z} - \mathbf{x}$  in (11.108) ergibt sich

$$\mathbf{z} - \mathbf{x} = \mathbf{x}^{(0)} + \mathbf{v} - \mathbf{x} = \mathbf{f}^{(0)} + \mathbf{v} \quad \text{mit } \mathbf{v} \in S_k = \mathcal{K}^{(k)}(\mathbf{r}^{(0)}, \mathbf{A}).$$

Die Vektoren  $\mathbf{z} - \mathbf{x}$ , welche zur Minimierung des Fehlers  $\mathbf{f}^{(k)}$  in (11.108) in Betracht kommen, besitzen deshalb wegen (11.109) folgende Darstellung

$$\begin{aligned}\mathbf{z} - \mathbf{x} &= \mathbf{f}^{(0)} + c_1\mathbf{Af}^{(0)} + c_2\mathbf{A}^2\mathbf{f}^{(0)} + \dots + c_k\mathbf{A}^k\mathbf{f}^{(0)} \\ &= [\mathbf{I} + c_1\mathbf{A} + c_2\mathbf{A}^2 + \dots + c_k\mathbf{A}^k]\mathbf{f}^{(0)} =: P_k(\mathbf{A})\mathbf{f}^{(0)}.\end{aligned} \quad (11.110)$$

Also gibt es ein Polynom  $P_k(t)$  mit reellen Koeffizienten vom Höchstgrad  $k$  und mit der Eigenschaft  $P_k(0) = 1$ , weil der Koeffizient von  $\mathbf{f}^{(0)}$  gleich Eins ist, so dass für den Fehler-

vektor  $\mathbf{f}^{(k)}$  im speziellen gilt

$$\mathbf{f}^{(k)} = P_k(\mathbf{A})\mathbf{f}^{(0)}. \quad (11.111)$$

Wegen der Optimalität der Näherung  $\mathbf{x}^{(k)}$  folgt aus (11.108) und (11.111)

$$\|\mathbf{f}^{(k)}\|_A = \min_{P_k(t)} \|P_k(\mathbf{A})\mathbf{f}^{(0)}\|_A, \quad (11.112)$$

wobei das Minimum über alle Polynome  $P_k(t)$  mit der oben genannten Eigenschaft zu bilden ist. Die Energienorm der rechten Seite (11.112) kann mit Hilfe der Eigenwerte von  $\mathbf{A}$  abgeschätzt werden. Seien  $0 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$  die  $n$  reellen Eigenwerte von  $\mathbf{A}$  und  $\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n$  die zugehörigen, orthonormierten Eigenvektoren. Aus der eindeutigen Darstellung von

$$\mathbf{f}^{(0)} = \alpha_1 \mathbf{z}_1 + \alpha_2 \mathbf{z}_2 + \dots + \alpha_n \mathbf{z}_n$$

folgt für die Energienorm

$$\|\mathbf{f}^{(0)}\|_A^2 = \left( \sum_{i=1}^n \alpha_i \mathbf{z}_i, \sum_{j=1}^n \alpha_j \lambda_j \mathbf{z}_j \right) = \sum_{i=1}^n \alpha_i^2 \lambda_i.$$

Da für die Eigenvektoren  $\mathbf{z}_i$  weiter  $P_k(\mathbf{A})\mathbf{z}_i = P_k(\lambda_i)\mathbf{z}_i$  gilt, ergibt sich analog

$$\begin{aligned} \|\mathbf{f}^{(k)}\|_A^2 &= \|P_k(\mathbf{A})\mathbf{f}^{(0)}\|_A^2 = (P_k(\mathbf{A})\mathbf{f}^{(0)}, \mathbf{A}P_k(\mathbf{A})\mathbf{f}^{(0)}) \\ &= \left( \sum_{i=1}^n \alpha_i P_k(\lambda_i) \mathbf{z}_i, \sum_{j=1}^n \alpha_j \lambda_j P_k(\lambda_j) \mathbf{z}_j \right) \\ &= \sum_{i=1}^n \alpha_i^2 \lambda_i P_k^2(\lambda_i) \leq \left[ \max_j \{P_k(\lambda_j)\}^2 \right] \cdot \|\mathbf{f}^{(0)}\|_A^2. \end{aligned} \quad (11.113)$$

Aus (11.112) und (11.113) erhalten wir die weitere Abschätzung

$$\frac{\|\mathbf{f}^{(k)}\|_A}{\|\mathbf{f}^{(0)}\|_A} \leq \min_{P_k(t)} \left\{ \max_{\lambda \in [\lambda_1, \lambda_n]} |P_k(\lambda)| \right\}. \quad (11.114)$$

Die durch eine Approximationssaufgabe definierte obere Schranke in (11.114) kann in Abhängigkeit von  $\lambda_1$  und  $\lambda_n$  mit Hilfe der Tschebyscheff-Polynome  $T_k(x)$  angegeben werden. Das Intervall  $[\lambda_1, \lambda_n]$  wird dazu mittels der Variablensubstitution  $x := (2\lambda - \lambda_1 - \lambda_n)/(\lambda_n - \lambda_1)$  auf das Einheitsintervall  $[-1, 1]$  abgebildet. Wegen der Minimax-Eigenschaft (3.224) besitzt das Polynom

$$P_k(\lambda) := T_k \left( \frac{2\lambda - \lambda_1 - \lambda_n}{\lambda_n - \lambda_1} \right) / T_k \left( \frac{\lambda_1 + \lambda_n}{\lambda_1 - \lambda_n} \right)$$

vom Grad  $k$  mit  $P_k(0) = 1$  im Intervall  $[\lambda_1, \lambda_n]$  die kleinste Betragssnorm, und es gilt insbesondere

$$\max_{\lambda \in [\lambda_1, \lambda_n]} |P_k(\lambda)| = 1 / \left| T_k \left( \frac{\lambda_1 + \lambda_n}{\lambda_1 - \lambda_n} \right) \right|. \quad (11.115)$$

Das Argument von  $T_k$  im Nenner von (11.115) ist betragsmäßig größer als Eins. Wegen (3.202) gelten mit  $x = \cos \varphi$

$$\begin{aligned}\cos \varphi &= \frac{1}{2}(e^{i\varphi} + e^{-i\varphi}) = \frac{1}{2}\left(z + \frac{1}{z}\right), \quad z = e^{i\varphi} \in \mathbb{C}, \\ \cos(n\varphi) &= \frac{1}{2}(e^{in\varphi} + e^{-in\varphi}) = \frac{1}{2}(z^n + z^{-n}), \\ z = \cos \varphi + i \sin \varphi &= \cos \varphi + i\sqrt{1 - \cos^2 \varphi} = x + \sqrt{x^2 - 1}, \\ T_n(x) = \cos(n\varphi) &= \frac{1}{2}\left[(x + \sqrt{x^2 - 1})^n + (x + \sqrt{x^2 - 1})^{-n}\right].\end{aligned}$$

Obwohl die letzte Formel für  $|x| \leq 1$  hergeleitet worden ist, gilt sie natürlich auch für  $|x| > 1$ . Jetzt sei mit  $\kappa(\mathbf{A}) = \lambda_n/\lambda_1$

$$x := -\frac{\lambda_1 + \lambda_n}{\lambda_1 - \lambda_n} = \frac{\lambda_n/\lambda_1 + 1}{\lambda_n/\lambda_1 - 1} = \frac{\kappa(\mathbf{A}) + 1}{\kappa(\mathbf{A}) - 1} > 1.$$

Dann ist

$$x + \sqrt{x^2 - 1} = \frac{\kappa + 1}{\kappa - 1} + \sqrt{\left(\frac{\kappa + 1}{\kappa - 1}\right)^2 - 1} = \frac{\kappa + 2\sqrt{\kappa} + 1}{\kappa - 1} = \frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1} > 1,$$

und folglich

$$T_k\left(\frac{\kappa + 1}{\kappa - 1}\right) = \frac{1}{2}\left[\left(\frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1}\right)^k + \left(\frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1}\right)^{-k}\right] \geq \frac{1}{2}\left(\frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1}\right)^k.$$

Als Ergebnis ergibt sich damit aus (11.114) der

**Satz 11.20.** Im CG-Verfahren (11.96) gilt für den Fehler  $\mathbf{f}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}$  in der Energienorm die Abschätzung

$$\frac{\|\mathbf{f}^{(k)}\|_A}{\|\mathbf{f}^{(0)}\|_A} \leq 2 \left( \frac{\sqrt{\kappa(\mathbf{A})} - 1}{\sqrt{\kappa(\mathbf{A})} + 1} \right)^k. \quad (11.116)$$

Auch wenn die Schranke (11.116) im Allgemeinen pessimistisch ist, so gibt sie doch den Hinweis, dass die Konditionszahl der Systemmatrix  $\mathbf{A}$  eine entscheidende Bedeutung für die Konvergenzgüte hat. Für die Anzahl  $k$  der erforderlichen CG-Schritte, derart dass  $\|\mathbf{f}^{(k)}\|_A/\|\mathbf{f}^{(0)}\|_A \leq \varepsilon$  ist, erhält man aus (11.116) die Schranke

$$k \geq \frac{1}{2} \sqrt{\kappa(\mathbf{A})} \ln \left( \frac{2}{\varepsilon} \right) + 1. \quad (11.117)$$

Neben der Toleranz  $\varepsilon$  ist die Schranke im Wesentlichen von der Wurzel aus der Konditionszahl von  $\mathbf{A}$  bestimmt. Das CG-Verfahren arbeitet dann effizient, wenn die Konditionszahl von  $\mathbf{A}$  nicht allzu groß ist oder aber durch geeignete Maßnahmen reduziert werden kann, sei es durch entsprechende Problemvorbereitung oder durch *Vorkonditionierung*.

**Beispiel 11.16.** Als Modellproblem nehmen wir die auf ein Rechteck verallgemeinerte Randwertaufgabe von Beispiel 11.1 mit  $f(x, y) = 2$ . Das Konvergenzverhalten der CG-Methode soll mit dem der SOR-Methode bei optimaler Wahl von  $\omega$  verglichen werden. In Tabelle 11.5 sind für einige Kombinationen der Werte  $N$  und  $M$  die Ordnung  $n$  der Matrix  $\mathbf{A}$ , ihre Konditionszahl  $\kappa(\mathbf{A})$ , die obere

Schranke  $k$  der Iterationschritte gemäß (11.117) für  $\varepsilon = 10^{-6}$ , die tatsächlich festgestellte Zahl der Iterationschritte  $k_{\text{eff}}$  unter dem Abbruchkriterium  $\|\mathbf{r}^{(k)}\|_2/\|\mathbf{r}^{(0)}\|_2 \leq \varepsilon$ , die zugehörige Rechenzeit  $t_{CG}$  sowie die entsprechenden Zahlen  $k_{SOR}$  und  $t_{SOR}$  zusammengestellt. Da der Residuenvektor im SOR-Verfahren nicht direkt verfügbar ist, wird hier als Abbruchkriterium  $\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|_2 \leq \varepsilon$  verwendet.

Tab. 11.5 Konvergenzverhalten des CG-Verfahrens, Modellproblem.

$N, M$	$n$	$\kappa(\mathbf{A})$	$k$	$k_{\text{eff}}$	$t_{CG}$	$k_{SOR}$	$t_{SOR}$
10, 6	60	28	39	14	0.8	23	1.7
20, 12	240	98	72	30	4.1	42	5.3
30, 18	540	212	106	46	10.8	61	14.4
40, 24	960	369	140	62	23.3	78	30.5
60, 36	2160	811	207	94	74.3	118	98.2
80, 48	3840	1424	274	125	171.8	155	226.1

Wegen (11.47) ist die Konditionszahl gegeben durch

$$\kappa(\mathbf{A}) = \left[ \sin^2\left(\frac{N\pi}{2N+2}\right) + \sin^2\left(\frac{M\pi}{2M+2}\right) \right] / \left[ \sin^2\left(\frac{\pi}{2N+2}\right) + \sin^2\left(\frac{\pi}{2M+2}\right) \right]$$

und nimmt bei Halbierung der Gitterweite  $h$  etwa um den Faktor vier zu, so dass sich dabei  $k$  verdoppelt. Die beobachteten Zahlen  $k_{\text{eff}}$  folgen diesem Gesetz, sind aber nur etwa halb so groß. Der Rechenaufwand steigt deshalb um den Faktor acht an. Dasselbe gilt für das SOR-Verfahren, wie auf Grund der Werte  $m_{SOR}$  in Tab. 11.1 zu erwarten ist.

Die Methode der konjugierten Gradienten löst die Gleichungssysteme mit geringerem Aufwand als das SOR-Verfahren. Für das CG-Verfahren spricht auch die Tatsache, dass kein Parameter gewählt werden muss, dass es problemlos auf allgemeine symmetrisch positiv definite Systeme anwendbar ist und dass die Konvergenz noch verbessert werden kann, wie wir im nächsten Abschnitt sehen werden.  $\triangle$

#### 11.4.4 Vorkonditionierung

Das Ziel, die Konvergenzeigenschaften der CG-Methode durch Reduktion der Konditionszahl  $\kappa(\mathbf{A})$  zu verbessern, erreicht man mit einer *Vorkonditionierung*, indem man das gegebene Gleichungssystem  $\mathbf{A}\mathbf{x} = \mathbf{b}$ ,  $\mathbf{A}$  symmetrisch und positiv definit, mit einer geeignet zu wählenden regulären Matrix  $\mathbf{C} \in \mathbb{R}^{n,n}$  in die äquivalente Form überführt

$$\mathbf{C}^{-1}\mathbf{A}\mathbf{C}^{-T}\mathbf{C}^T\mathbf{x} = \mathbf{C}^{-1}\mathbf{b}. \quad (11.118)$$

Mit den neuen Größen

$$\tilde{\mathbf{A}} := \mathbf{C}^{-1}\mathbf{A}\mathbf{C}^{-T}, \quad \tilde{\mathbf{x}} := \mathbf{C}^T\mathbf{x}, \quad \tilde{\mathbf{b}} := \mathbf{C}^{-1}\mathbf{b} \quad (11.119)$$

lautet das transformierte Gleichungssystem

$$\tilde{\mathbf{A}}\tilde{\mathbf{x}} = \tilde{\mathbf{b}} \quad (11.120)$$

mit ebenfalls symmetrischer und positiv definiter Matrix  $\tilde{\mathbf{A}}$ , welche aus  $\mathbf{A}$  durch eine *Kontrahenztransformation* hervorgeht, so dass dadurch die Eigenwerte und damit die Konditions-

zahl mit günstigem  $\mathbf{C}$  im beabsichtigten Sinn beeinflusst werden können. Eine zweckmäßige Festlegung von  $\mathbf{C}$  muss

$$\kappa_2(\tilde{\mathbf{A}}) = \kappa_2(\mathbf{C}^{-1} \mathbf{A} \mathbf{C}^{-T}) \ll \kappa_2(\mathbf{A})$$

erreichen. Hierbei hilft die Feststellung, dass  $\tilde{\mathbf{A}}$  ähnlich ist zur Matrix

$$\mathbf{K} := \mathbf{C}^{-T} \tilde{\mathbf{A}} \mathbf{C}^T = \mathbf{C}^{-T} \mathbf{C}^{-1} \mathbf{A} = (\mathbf{C} \mathbf{C}^T)^{-1} \mathbf{A} =: \mathbf{M}^{-1} \mathbf{A}. \quad (11.121)$$

Die symmetrische und positiv definite Matrix  $\mathbf{M} := \mathbf{C} \mathbf{C}^T$  spielt die entscheidene Rolle, und man nennt sie die *Vorkonditionierungsmatrix*. Wegen der Ähnlichkeit von  $\tilde{\mathbf{A}}$  und  $\mathbf{K}$  gilt natürlich

$$\kappa_2(\tilde{\mathbf{A}}) = \lambda_{\max}(\mathbf{M}^{-1} \mathbf{A}) / \lambda_{\min}(\mathbf{M}^{-1} \mathbf{A}).$$

Mit  $\mathbf{M} = \mathbf{A}$  hätte man  $\kappa_2(\tilde{\mathbf{A}}) = \kappa_2(\mathbf{I}) = 1$ . Doch ist diese Wahl nicht sinnvoll, denn mit der Cholesky-Zerlegung  $\mathbf{A} = \mathbf{C} \mathbf{C}^T$ , wo  $\mathbf{C}$  eine Linksdreiecksmatrix ist, wäre das Gleichungssystem direkt lösbar, aber das ist für großes  $n$  problematisch. Jedenfalls soll  $\mathbf{M}$  eine Approximation von  $\mathbf{A}$  sein, womöglich unter Beachtung der schwachen Besetzung der Matrix  $\mathbf{A}$ .

Der CG-Algorithmus (11.96) für das vorkonditionierte Gleichungssystem (11.120) lautet bei vorgegebener Matrix  $\mathbf{C}$  wie folgt:

$$\begin{aligned} &\text{Start: Wahl von } \tilde{\mathbf{x}}^{(0)}; \tilde{\mathbf{r}}^{(0)} = \tilde{\mathbf{A}}\tilde{\mathbf{x}}^{(0)} - \tilde{\mathbf{b}}; \tilde{\mathbf{p}}^{(1)} = -\tilde{\mathbf{r}}^{(0)}; \\ &\text{Iteration } k = 1, 2, 3, \dots : \\ &\text{Falls } k > 1 : \begin{cases} \tilde{e}_{k-1} &= (\tilde{\mathbf{r}}^{(k-1)}, \tilde{\mathbf{r}}^{(k-1)}) / (\tilde{\mathbf{r}}^{(k-2)}, \tilde{\mathbf{r}}^{(k-2)}) \\ \tilde{\mathbf{p}}^{(k)} &= -\tilde{\mathbf{r}}^{(k-1)} + \tilde{e}_{k-1} \tilde{\mathbf{p}}^{(k-1)} \end{cases} \\ &\tilde{\mathbf{z}} = \tilde{\mathbf{A}}\tilde{\mathbf{p}}^{(k)} \\ &\tilde{q}_k = (\tilde{\mathbf{r}}^{(k-1)}, \tilde{\mathbf{r}}^{(k-1)}) / (\tilde{\mathbf{p}}^{(k)}, \tilde{\mathbf{z}}) \\ &\tilde{\mathbf{x}}^{(k)} = \tilde{\mathbf{x}}^{(k-1)} + \tilde{q}_k \tilde{\mathbf{p}}^{(k)}; \quad \tilde{\mathbf{r}}^{(k)} = \tilde{\mathbf{r}}^{(k-1)} + \tilde{q}_k \tilde{\mathbf{z}} \\ &\text{Test auf Konvergenz} \end{aligned} \quad (11.122)$$

Im Prinzip kann der CG-Algorithmus in der Form (11.122) auf der Basis der Matrix  $\tilde{\mathbf{A}}$  durchgeführt werden. Neben der Berechnung von  $\tilde{\mathbf{b}}$  als Lösung von  $\mathbf{C}\tilde{\mathbf{b}} = \mathbf{b}$  ist die Multiplikation  $\tilde{\mathbf{z}} = \tilde{\mathbf{A}}\tilde{\mathbf{p}}^{(k)} = \mathbf{C}^{-1} \mathbf{A} \mathbf{C}^{-T} \tilde{\mathbf{p}}^{(k)}$  in drei Teilschritten zu realisieren. Erstens ist ein Gleichungssystem mit  $\mathbf{C}^T$  zu lösen, zweitens ist die Matrixmultiplikation mit  $\mathbf{A}$  auszuführen und schließlich ist noch ein Gleichungssystem mit  $\mathbf{C}$  zu lösen. Am Schluss ist aus der resultierenden Näherung  $\tilde{\mathbf{x}}^{(k)}$  die Näherungslösung  $\mathbf{x}^{(k)}$  des gegebenen Systems aus  $\mathbf{C}^T \mathbf{x}^{(k)} = \tilde{\mathbf{x}}^{(k)}$  zu ermitteln.

Zweckmäßiger ist es, den Algorithmus (11.122) neu so zu formulieren, dass mit den gegebenen Größen gearbeitet wird und dass eine Folge von iterierten Vektoren  $\mathbf{x}^{(k)}$  erzeugt wird, welche Näherungen der gesuchten Lösung  $\mathbf{x}$  sind. Die Vorkonditionierung wird gewissermaßen *implizit* angewandt.

Wegen (11.119) und (11.120) gelten die Relationen

$$\tilde{\mathbf{x}}^{(k)} = \mathbf{C}^T \mathbf{x}^{(k)}, \quad \tilde{\mathbf{r}}^{(k)} = \mathbf{C}^{-1} \mathbf{r}^{(k)}. \quad (11.123)$$

Da der Richtungsvektor  $\tilde{\mathbf{p}}^{(k)}$  mit Hilfe des Residuenvektors  $\tilde{\mathbf{r}}^{(k-1)}$  gebildet wird, führen wir die Hilfsvektoren  $\mathbf{s}^{(k)} := \mathbf{C}\tilde{\mathbf{p}}^{(k)}$  ein, womit wir zum Ausdruck bringen, dass die  $\mathbf{s}^{(k)}$  nicht mit den Richtungsvektoren  $\mathbf{p}^{(k)}$  des nicht-vorkonditionierten CG-Algorithmus identisch zu sein brauchen. Aus der Rekursionsformel für die iterierten Vektoren  $\tilde{\mathbf{x}}^{(k)}$  ergibt sich so

$$\mathbf{C}^T \mathbf{x}^{(k)} = \mathbf{C}^T \mathbf{x}^{(k-1)} + \tilde{q}_k \mathbf{C}^{-1} \mathbf{s}^{(k)}$$

und nach Multiplikationen mit  $\mathbf{C}^{-T}$  von links

$$\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} + \tilde{q}_k (\mathbf{M}^{-1} \mathbf{s}^{(k)}).$$

Desgleichen erhalten wir aus der Rekursionsformel der Residuenvektoren

$$\mathbf{C}^{-1} \mathbf{r}^{(k)} = \mathbf{C}^{-1} \mathbf{r}^{(k-1)} + \tilde{q}_k \mathbf{C}^{-1} \mathbf{A} \mathbf{C}^{-T} \mathbf{C}^{-1} \mathbf{s}^{(k)}$$

nach Multiplikation von links mit  $\mathbf{C}$

$$\mathbf{r}^{(k)} = \mathbf{r}^{(k-1)} + \tilde{q}_k \mathbf{A} (\mathbf{M}^{-1} \mathbf{s}^{(k)}).$$

In beiden Beziehungen treten die Vektoren

$$\mathbf{M}^{-1} \mathbf{s}^{(k)} =: \mathbf{g}^{(k)} \tag{11.124}$$

auf, mit denen für  $\mathbf{x}^{(k)}$  und  $\mathbf{r}^{(k)}$  einfachere Formeln resultieren. Aber auch aus den Definitionsgleichungen für die Richtungsvektoren in (11.122)

$$\tilde{\mathbf{p}}^{(k)} = -\tilde{\mathbf{r}}^{(k-1)} + \tilde{e}_{k-1} \tilde{\mathbf{p}}^{(k-1)} \quad \text{mit} \quad \tilde{e}_{k-1} = (\tilde{\mathbf{r}}^{(k-1)}, \tilde{\mathbf{r}}^{(k-1)}) / (\tilde{\mathbf{r}}^{(k-2)}, \tilde{\mathbf{r}}^{(k-2)})$$

ergibt sich nach Substitution der Größen und Multiplikation von links mit  $\mathbf{C}^{-T}$

$$\mathbf{M}^{-1} \mathbf{s}^{(k)} = -\mathbf{M}^{-1} \mathbf{r}^{(k-1)} + \tilde{e}_{k-1} \mathbf{M}^{-1} \mathbf{s}^{(k-1)}.$$

Mit der weiteren Definition

$$\mathbf{M}^{-1} \mathbf{r}^{(k)} =: \boldsymbol{\varrho}^{(k)} \tag{11.125}$$

wird die letzte Beziehung zu

$$\mathbf{g}^{(k)} = -\boldsymbol{\varrho}^{(k-1)} + \tilde{e}_{k-1} \mathbf{g}^{(k-1)}. \tag{11.126}$$

Dies ist die Ersatzgleichung für die Richtungsvektoren  $\tilde{\mathbf{p}}^{(k)}$ , die durch die  $\mathbf{g}^{(k)}$  ersetzt werden. Schließlich lassen sich die Skalarprodukte in (11.122) durch die neuen Größen wie folgt darstellen:

$$\begin{aligned} (\tilde{\mathbf{r}}^{(k)}, \tilde{\mathbf{r}}^{(k)}) &= (\mathbf{C}^{-1} \mathbf{r}^{(k)}, \mathbf{C}^{-1} \mathbf{r}^{(k)}) = (\mathbf{r}^{(k)}, \mathbf{M}^{-1} \mathbf{r}^{(k)}) = (\mathbf{r}^{(k)}, \boldsymbol{\varrho}^{(k)}) \\ (\tilde{\mathbf{p}}^{(k)}, \tilde{\mathbf{z}}) &= (\mathbf{C}^{-1} \mathbf{s}^{(k)}, \mathbf{C}^{-1} \mathbf{A} \mathbf{C}^{-T} \mathbf{C}^{-1} \mathbf{s}^{(k)}) \\ &= (\mathbf{M}^{-1} \mathbf{s}^{(k)}, \mathbf{A} \mathbf{M}^{-1} \mathbf{s}^{(k)}) = (\mathbf{g}^{(k)}, \mathbf{A} \mathbf{g}^{(k)}) \end{aligned}$$

Für den Start des Algorithmus wird noch  $\mathbf{g}^{(1)}$  anstelle von  $\tilde{\mathbf{p}}^{(1)}$  benötigt. Dafür ergibt sich

$$\begin{aligned} \mathbf{g}^{(1)} &= \mathbf{M}^{-1} \mathbf{s}^{(1)} = \mathbf{C}^{-T} \tilde{\mathbf{p}}^{(1)} = -\mathbf{C}^{-T} \tilde{\mathbf{r}}^{(0)} = -\mathbf{C}^{-T} \mathbf{C}^{-1} \mathbf{r}^{(0)} \\ &= -\mathbf{M}^{-1} \mathbf{r}^{(0)} = -\boldsymbol{\varrho}^{(0)}. \end{aligned}$$

Dieser Vektor muss als Lösung eines Gleichungssystems bestimmt werden. Ein System mit  $\mathbf{M}$  als Koeffizientenmatrix muss in jedem PCG-Schritt aufgelöst werden. Wird das im Algorithmus berücksichtigt und außerdem die Fallunterscheidung für  $k = 1$  durch Hilfsgrößen

eliminiert, so ergibt sich der folgende *PCG-Algorithmus*:

$\text{Start: Festsetzung von } \mathbf{M}; \text{ Wahl von } \mathbf{x}^{(0)};$ $\mathbf{r}^{(0)} = \mathbf{A}\mathbf{x}^{(0)} - \mathbf{b}; \zeta_a = 1; \mathbf{g}^{(0)} = \mathbf{0};$ $\text{Iteration } k = 1, 2, 3, \dots :$ $\mathbf{M}\boldsymbol{\varrho}^{(k-1)} = \mathbf{r}^{(k-1)} \ (\rightarrow \boldsymbol{\varrho}^{(k-1)})$ $\zeta = (\mathbf{r}^{(k-1)}, \boldsymbol{\varrho}^{(k-1)}); \tilde{e}_{k-1} = \zeta/\zeta_a$ $\mathbf{g}^{(k)} = -\boldsymbol{\varrho}^{(k-1)} + \tilde{e}_{k-1}\mathbf{g}^{(k-1)}$ $\mathbf{z} = \mathbf{A}\mathbf{g}^{(k)}$ $\tilde{q}_k = \zeta/(\mathbf{g}^{(k)}, \mathbf{z}); \zeta_a = \zeta;$ $\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} + \tilde{q}_k\mathbf{g}^{(k)}; \mathbf{r}^{(k)} = \mathbf{r}^{(k-1)} + \tilde{q}_k\mathbf{z};$ Test auf Konvergenz	<span style="float: right;">(11.127)</span>
---	---

Die Matrix  $\mathbf{C}$ , von der wir ursprünglich ausgegangen sind, tritt im PCG-Algorithmus (11.127) nicht mehr auf, sondern nur noch die symmetrische positiv definite Vorkonditionierungsmaatrix  $\mathbf{M}$ . Im Vergleich zum normalen CG-Algorithmus (11.96) erfordert jetzt jeder Iterationsschritt die Auflösung des linearen Gleichungssystems  $\mathbf{M}\boldsymbol{\varrho} = \mathbf{r}$  nach  $\boldsymbol{\varrho}$  als so genannten *Vorkonditionierungsschritt*. Die Matrix  $\mathbf{M}$  muss unter dem Gesichtspunkt gewählt werden, dass dieser zusätzliche Aufwand im Verhältnis zur Konvergenzverbesserung nicht zu hoch ist. Deshalb kommen in erster Linie Matrizen  $\mathbf{M} = \mathbf{CC}^T$  in Betracht, welche sich als Produkt einer schwach besetzten Linksdreiecksmatrix  $\mathbf{C}$  und ihrer Transponierten darstellen. Die Prozesse der Vorwärts- und Rücksubstitution werden, zumindest auf Skalarrechnern, effizient durchführbar. Hat  $\mathbf{C}$  die gleiche Besetzungsstruktur wie die untere Hälfte von  $\mathbf{A}$ , dann ist das Auflösen von  $\mathbf{M}\boldsymbol{\varrho} = \mathbf{r}$  praktisch gleich aufwändig wie eine Matrix-Vektor-Multiplikation  $\mathbf{z} = \mathbf{Ag}$ . Der Rechenaufwand pro Iterationsschritt des vorkonditionierten CG-Algorithmus (11.127) verdoppelt sich etwa im Vergleich zum Algorithmus (11.96). Die einfachste, am wenigsten Mehraufwand erfordерnde Wahl von  $\mathbf{M}$  besteht darin,  $\mathbf{M} := \text{diag}(a_{11}, a_{22}, \dots, a_{nn})$  als Diagonalmatrix mit den positiven Diagonalelementen  $a_{ii}$  von  $\mathbf{A}$  festzulegen. Der Vorkonditionierungsschritt erfordert dann nur  $n$  zusätzliche Operationen. Da in diesem Fall  $\mathbf{C} = \text{diag}(\sqrt{a_{11}}, \sqrt{a_{22}}, \dots, \sqrt{a_{nn}})$  ist, so ist die vorkonditionierte Matrix  $\tilde{\mathbf{A}}$  (11.119) gegeben durch

$$\tilde{\mathbf{A}} = \mathbf{C}^{-1}\mathbf{AC}^{-T} = \mathbf{C}^{-1}\mathbf{AC}^{-1} = \mathbf{E} + \mathbf{I} + \mathbf{F}, \quad \mathbf{F} = \mathbf{E}^T \quad (11.128)$$

wo  $\mathbf{E}$  eine strikte untere Dreiecksmatrix bedeutet. Mit dieser Vorkonditionierungsmaatrix  $\mathbf{M}$  wird die Matrix  $\mathbf{A}$  *skaliert*, derart dass die Diagonalelemente von  $\tilde{\mathbf{A}}$  gleich Eins werden. Diese Skalierung hat in jenen Fällen, in denen die Diagonalelemente sehr unterschiedliche Größenordnung haben, oft eine starke Reduktion der Konditionszahl zur Folge. Für Gleichungssysteme, welche aus dem Differenzenverfahren oder der Methode der finiten Elemente mit linearen oder quadratischen Ansätzen für elliptische Randwertaufgaben resultieren, hat die Skalierung allerdings entweder keine oder nur eine minimale Verkleinerung der Konditionszahl zur Folge.

Für die beiden im Folgenden skizzierten Definitionen von Vorkonditionierungsmaatrizen  $\mathbf{M}$  wird vorausgesetzt, dass die Matrix  $\mathbf{A}$  skaliert ist und die Gestalt (11.128) hat. Mit einem

geeignet zu wählenden Parameter  $\omega$  kann  $\mathbf{M}$  wie folgt festgelegt werden [Eva 83, Axe 89]

$$\mathbf{M} := (\mathbf{I} + \omega \mathbf{E})(\mathbf{I} + \omega \mathbf{F}), \quad \text{also } \mathbf{C} := (\mathbf{I} + \omega \mathbf{E}). \quad (11.129)$$

$\mathbf{C}$  ist eine reguläre Linksdreiecksmatrix mit derselben Besetzungsstruktur wie die untere Hälfte von  $\mathbf{A}$ . Zu ihrer Festlegung wird kein zusätzlicher Speicherbedarf benötigt. Die Lösung von  $\mathbf{M}\boldsymbol{\varphi} = \mathbf{r}$  erfolgt mit den beiden Teilschritten

$$(\mathbf{I} + \omega \mathbf{E})\mathbf{y} = \mathbf{r} \quad \text{und} \quad (\mathbf{I} + \omega \mathbf{F})\boldsymbol{\varphi} = \mathbf{y}. \quad (11.130)$$

Die Prozesse der Vorwärts- und Rücksubstitution (11.130) erfordern bei geschickter Beachtung des Faktors  $\omega$  und der schwachen Besetzung von  $\mathbf{E}$  und  $\mathbf{F} = \mathbf{E}^T$  zusammen einen Aufwand von  $(\gamma + 1)n$  wesentlichen Operationen mit  $\gamma$  wie in (11.97). Der Rechenaufwand eines Iterationsschrittes des vorkonditionierten CG-Algorithmus (11.127) beläuft sich auf etwa

$$Z_{\text{PCGS}} = (2\gamma + 6)n \quad (11.131)$$

multiplikative Operationen. Für eine bestimmte Klasse von Matrizen  $\mathbf{A}$  kann gezeigt werden [Axe 84], dass bei optimaler Wahl von  $\omega$  die Konditionszahl  $\kappa(\tilde{\mathbf{A}})$  etwa gleich der Quadratwurzel von  $\kappa(\mathbf{A})$  ist, so dass sich die Verdoppelung des Aufwandes pro Schritt wegen der starken Reduktion der Iterationszahl lohnt. Die Vorkonditionierungsmatrix  $\mathbf{M}$  (11.129) stellt tatsächlich für  $\omega \neq 0$  eine Approximation eines Vielfachen von  $\mathbf{A}$  dar, denn es gilt

$$\mathbf{M} = \mathbf{I} + \omega \mathbf{E} + \omega \mathbf{F} + \omega^2 \mathbf{EF} = \omega[\mathbf{A} + (\omega^{-1} - 1)\mathbf{I} + \omega \mathbf{EF}],$$

und  $(\omega^{-1} - 1)\mathbf{I} + \omega \mathbf{EF}$  ist der Approximationsfehler. Die Abhängigkeit der Zahl der Iterationen von  $\omega$  ist in der Gegend des optimalen Wertes nicht sehr empfindlich, da der zugehörige Graph ein flaches Minimum aufweist. Wegen dieses zur symmetrischen Überrelaxation (=SSOR-Methode) [Axe 84, Sch 72] analogen Verhaltens bezeichnen wir (11.127) mit der Vorkonditionierungsmatrix  $\mathbf{M}$  (11.129) als SSORCG-Methode.

Eine andere Möglichkeit, die Matrix  $\mathbf{M}$  zu definieren, besteht im Ansatz

$$\mathbf{M} := (\mathbf{D} + \mathbf{E})\mathbf{D}^{-1}(\mathbf{D} + \mathbf{F}), \quad (11.132)$$

wo  $\mathbf{E}$  und  $\mathbf{F}$  aus der skalierten Matrix  $\mathbf{A}$  (11.128) übernommen werden und die Diagonalmatrix  $\mathbf{D}$  mit positiven Diagonalelementen unter einer Zusatzbedingung zu ermitteln ist. Als Vorkonditionierungsmatrix für Differenzengleichungen von elliptischen Randwertaufgaben hat es sich als günstig erwiesen,  $\mathbf{D}$  so festzulegen, dass entweder die Zeilensummen von  $\mathbf{M}$  im Wesentlichen, d.h. bis auf ein additives  $\alpha \geq 0$ , mit denjenigen von  $\mathbf{A}$  übereinstimmen oder dass einfacher die Diagonalelemente von  $\mathbf{M}$  gleich  $1 + \alpha$ ,  $\alpha \geq 0$ , sind. Die letztgenannte Forderung liefert nach einfacher Rechnung für die Diagonalelemente  $d_i$  von  $\mathbf{D}$  die Rekursionsformel

$$d_i = 1 + \alpha - \sum_{k=1}^{i-1} a_{ik}^2 d_k^{-1}, \quad i = 1, 2, \dots, n. \quad (11.133)$$

Der Wert  $\alpha$  dient hauptsächlich dazu, in (11.133) die Bedingung  $d_i > 0$  zu erfüllen. Neben den erwähnten Vorkonditionierungsmatrizen  $\mathbf{M}$  existieren viele weitere, den Problemstellungen oder den Aspekten einer Vektorisierung oder Parallelisierung auf modernen Rechenanlagen angepasste Definitionen. Zu nennen sind etwa die *partielle Cholesky-Zerlegung* von  $\mathbf{A}$ , bei welcher zur Gewinnung einer Linksdreiecksmatrix  $\mathbf{C}$  der fill-in bei

der Zerlegung entweder ganz oder nach bestimmten Gesetzen vernachlässigt wird [Axe 84, Eva 83, Gol 96b, Jen 77, Ker 78, Mei 77, Sch 91b]. Weiter existieren Vorschläge für  $\mathbf{M}$ , welche auf blockweisen Darstellungen von  $\mathbf{A}$  und Gebietszerlegung basieren [Axe 85, Bra 86, Con 85, Cha 89]. Für Gleichungssysteme aus der Methode der finiten Elemente sind Varianten der Vorkonditionierung auf der Basis der Elementmatrizen vorgeschlagen worden [Bar 88a, Cri 86, Hug 83, NO 85]. Vorkonditionierungsmatrizen  $\mathbf{M}$ , welche mit Hilfe von so genannten *hierarchischen Basen* gewonnen werden, erweisen sich als äußerst konvergenzsteigernd [Yse 86, Yse 90]. Verschiedene andere Beiträge in dieser Richtung findet man etwa in [Axe 89, Axe 90, Bru 95].

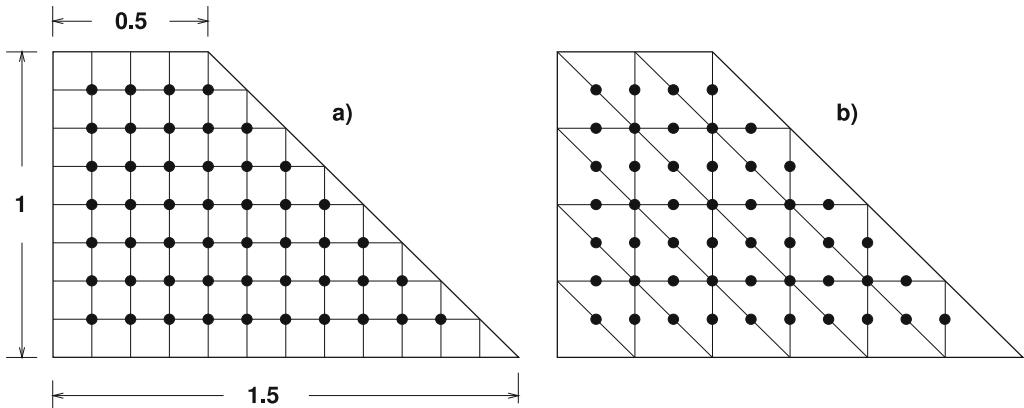


Abb. 11.19 a) Differenzengitter, b) Triangulierung für quadratische finite Elemente.

**Beispiel 11.17.** Zur Illustration der Vorkonditionierung mit der Matrix  $\mathbf{M}$  (11.129) und der Abhängigkeit ihres Effektes von  $\omega$  betrachten wir die elliptische Randwertaufgabe

$$\begin{aligned} -u_{xx} - u_{yy} &= 2 \quad \text{in } G, \\ u &= 0 \quad \text{auf } \Gamma, \end{aligned} \tag{11.134}$$

wo  $G$  das trapezförmige Gebiet in Abb. 11.19 darstellt und  $\Gamma$  sein Rand ist. Die Aufgabe wird sowohl mit dem Differenzenverfahren mit der Fünf-Punkte-Differenzenapproximation (11.1) als auch mit der Methode der finiten Elemente mit quadratischem Ansatz auf einem Dreiecksnetz behandelt, von denen in Abb. 11.19 a) und b) je ein Fall dargestellt sind. Wird die Kathetenlänge eines Dreieckelementes gleich der doppelten Gitterweite  $h$  des Gitters des Differenzenverfahrens gewählt, ergeben sich gleich viele Unbekannte in inneren Gitter- oder Knotenpunkten. In Tab. 11.6 sind für einige Gitterweiten  $h$  die Zahl  $n$  der Unbekannten, die Zahl  $k_{CG}$  der Iterationsschritte des CG-Verfahrens (11.96), der optimale Wert  $\omega_{\text{opt}}$  des SSORCG-Verfahrens und die Zahl  $k_{PCG}$  der Iterationen des vorkonditionierten CG-Verfahrens für die beiden Methoden zusammengestellt. Die Iteration wurde abgebrochen, sobald  $\|\mathbf{r}^{(k)}\|/\|\mathbf{r}^{(0)}\| \leq 10^{-6}$  erfüllt ist. Die angewandte Vorkonditionierung bringt die gewünschte Reduktion des Rechenaufwandes, die für feinere Diskretisierungen größer wird. Die Beispiele sind mit Programmen aus [Sch 91b] gerechnet worden.

In Abb. 11.20 ist die Anzahl der Iterationsschritte in Abhängigkeit von  $\omega$  im Fall  $h = 1/32$  beim Differenzenverfahren dargestellt.  $\omega = 0$  entspricht keiner Vorkonditionierung oder  $\mathbf{M} = \mathbf{I}$ . Das flache Minimum des Graphen ist deutlich.  $\triangle$

Tab. 11.6 Konvergenzverhalten bei Vorkonditionierung.

$h^{-1}$	$n$	Differenzenverfahren			finite Elemente		
		$k_{\text{CG}}$	$\omega_{\text{opt}}$	$k_{\text{PCG}}$	$k_{\text{CG}}$	$\omega_{\text{opt}}$	$k_{\text{PCG}}$
8	49	21	1.30	8	24	1.30	9
12	121	33	1.45	10	38	1.45	11
16	225	45	1.56	12	52	1.56	13
24	529	68	1.70	14	79	1.70	16
32	961	91	1.75	17	106	1.75	19
48	2209	137	1.83	21	161	1.83	23
64	3969	185	1.88	24	-	-	-

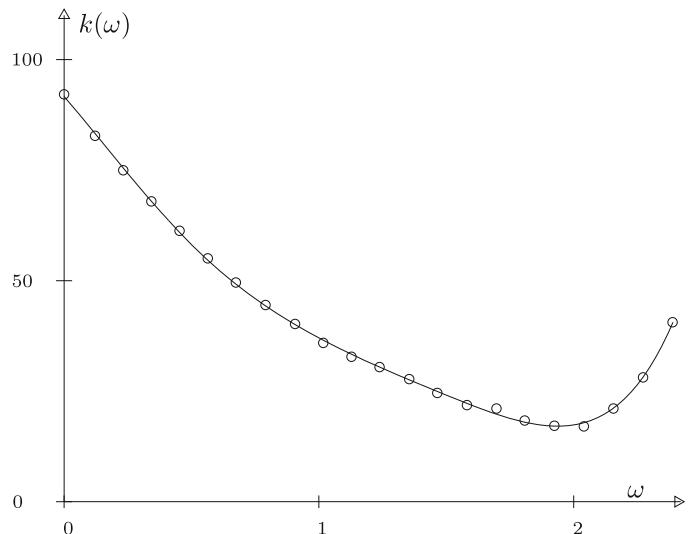


Abb. 11.20 Iterationszahl des SSORCG-Verfahrens.

## 11.5 Methode der verallgemeinerten minimierten Residuen

In diesem Abschnitt betrachten wir eine robuste, in vielen Fällen recht effiziente Methode zur iterativen Lösung von linearen Gleichungssystemen  $\mathbf{A}\mathbf{x} = \mathbf{b}$  mit regulärer, schwach besetzter Matrix  $\mathbf{A}$ , die unsymmetrisch oder symmetrisch und indefinit sein kann. Wir wollen aber nach wie vor voraussetzen, dass die Diagonalelemente  $a_{ii} \neq 0$  sind, obwohl dies für den grundlegenden Algorithmus nicht benötigt wird, aber für die die Konvergenz verbessernende Modifikation gebraucht wird.

### 11.5.1 Grundlagen des Verfahrens

Das Verfahren verwendet einige analoge Elemente wie die Methode der konjugierten Gradienten. In der betrachteten allgemeinen Situation mit  $\mathbf{A} \neq \mathbf{A}^T$  oder  $\mathbf{A}$  symmetrisch und indefinit ist die Lösung  $\mathbf{x}$  von  $\mathbf{Ax} = \mathbf{b}$  nicht mehr durch das Minimum der Funktion  $F(\mathbf{v})$  (11.81) charakterisiert. Doch ist  $\mathbf{x}$  unter allen Vektoren  $\mathbf{w} \in \mathbb{R}^n$  der eindeutige Vektor, welcher das Quadrat der euklidischen *Residuen-Norm* zu null macht und folglich minimiert. Aus diesem Grund legen wir im Folgenden das zu minimierende Funktional

$$J(\mathbf{w}) := \|\mathbf{Aw} - \mathbf{b}\|_2^2 \quad \text{mit } J(\mathbf{x}) = \min_{\mathbf{w} \in \mathbb{R}^n} J(\mathbf{w}) = 0 \quad (11.135)$$

zu Grunde. In Analogie zum CG-Verfahren soll der  $k$ -te iterierte Vektor  $\mathbf{x}^{(k)}$  die Darstellung

$$\mathbf{x}^{(k)} = \mathbf{x}^{(0)} + \mathbf{z} \quad (11.136)$$

besitzen, wo  $\mathbf{z}$  einem Unterraum  $S_k$  angehören soll, dessen Dimension von Schritt zu Schritt zunimmt und wiederum durch die sukzessiv sich ergebenden Residuen-Vektoren  $\mathbf{r}^{(0)}, \mathbf{r}^{(1)}, \dots, \mathbf{r}^{(k-1)}$  aufgespannt sei. Der Vektor  $\mathbf{x}^{(k)}$  wird so bestimmt, dass er das Funktional (11.135) minimiert, d.h. dass gilt

$$\begin{aligned} J(\mathbf{x}^{(k)}) &= \min_{\mathbf{z} \in S_k} J(\mathbf{x}^{(0)} + \mathbf{z}) = \min_{\mathbf{z} \in S_k} \|\mathbf{A}(\mathbf{x}^{(0)} + \mathbf{z}) - \mathbf{b}\|_2^2 \\ &= \min_{\mathbf{z} \in S_k} \|\mathbf{Az} + \mathbf{r}^{(0)}\|_2^2. \end{aligned} \quad (11.137)$$

Durch die Forderung (11.137) ist die *Methode der minimierten Residuen* (MINRES) definiert. Da  $\mathbf{z} \in S_k$  als Linearkombination von  $k$  Basisvektoren von  $S_k$  darstellbar ist, stellt (11.137) eine typische Aufgabe der Methode der kleinsten Quadrate dar, die im Prinzip mit den Hilfsmitteln von Kapitel 6 gelöst werden kann. Da aber die Fehlergleichungsmatrix  $\mathbf{C}$  der Aufgabe (11.137), deren Spalten durch die  $\mathbf{A}$ -fachen der Basisvektoren  $\mathbf{r}^{(i)}$  gegeben sind, die Tendenz hat, schlecht konditioniert zu sein, d.h. fast linear abhängige Spalten aufzuweisen, bietet dieses Vorgehen numerische Schwierigkeiten. Die *Methode der verallgemeinerten minimierten Residuen* (GMRES) [Saa 81, Saa 86, Wal 88] besteht in einer numerisch besonders geschickten Behandlung der Minimierungsaufgabe (11.137).

Man stellt leicht fest, dass der Unterraum  $S_k := \text{span}\{\mathbf{r}^{(0)}, \mathbf{r}^{(1)}, \dots, \mathbf{r}^{(k-1)}\}$  identisch ist mit dem *Krylov-Unterraum*  $\mathcal{K}^{(k)}(\mathbf{r}^{(0)}, \mathbf{A})$ . Für  $k = 1$  ist die Aussage trivial. Falls die Behauptung für ein  $k \geq 1$  richtig ist, dann hat der Residuen-Vektor  $\mathbf{r}^{(k)}$  die Darstellung

$$\mathbf{r}^{(k)} = \mathbf{r}^{(0)} + \mathbf{Az} = \mathbf{r}^{(0)} + \sum_{i=1}^k c_i (\mathbf{A}^i \mathbf{r}^{(0)}), \quad (11.138)$$

weil  $\mathbf{z} \in S_k = \mathcal{K}^{(k)}(\mathbf{r}^{(0)}, \mathbf{A})$ , wobei die Koeffizienten  $c_i$  durch die Minimierungsaufgabe (11.137) bestimmt sind. Wegen (11.138) ist

$$\mathbf{r}^{(k)} \in \mathcal{K}^{(k+1)}(\mathbf{r}^{(0)}, \mathbf{A}) = \text{span}\{\mathbf{r}^{(0)}, \mathbf{Ar}^{(0)}, \dots, \mathbf{A}^k \mathbf{r}^{(0)}\},$$

und folglich gilt

$$S_k := \text{span}\{\mathbf{r}^{(0)}, \mathbf{r}^{(1)}, \dots, \mathbf{r}^{(k-1)}\} = \mathcal{K}^{(k)}(\mathbf{r}^{(0)}, \mathbf{A}). \quad (11.139)$$

Um die Minimierungsaufgabe (11.137) bezüglich des Krylov-Unterraumes  $\mathcal{K}^{(k)}(\mathbf{r}^{(0)}, \mathbf{A})$  numerisch sicher zu behandeln, ist es zweckmäßig, in der Folge von Krylov-Unterräumen sukzessive eine orthonormierte Basis zu konstruieren. Dies erfolgt mit Hilfe des Schmidt'schen

Orthogonalisierungsprozesses. Im ersten Schritt wird der Startvektor  $\mathbf{r}^{(0)}$  normiert zum ersten Basisvektor  $\mathbf{v}_1$  gemäß

$$\beta := \|\mathbf{r}^{(0)}\|_2, \quad \mathbf{v}_1 := \mathbf{r}^{(0)}/\beta \quad \text{oder} \quad \mathbf{r}^{(0)} = \beta \mathbf{v}_1. \quad (11.140)$$

Im zweiten Schritt wird anstelle des Vektors  $\mathbf{A}\mathbf{r}^{(0)}$  der dazu äquivalente Vektor  $\mathbf{A}\mathbf{v}_1$  zur Konstruktion des zweiten Basisvektors  $\mathbf{v}_2$  verwendet. Der Orthonormierungsschritt besteht aus den beiden Teilschritten

$$\hat{\mathbf{v}}_2 := \mathbf{A}\mathbf{v}_1 - (\mathbf{v}_1, \mathbf{A}\mathbf{v}_1)\mathbf{v}_1 = \mathbf{A}\mathbf{v}_1 - h_{11}\mathbf{v}_1, \quad h_{11} := (\mathbf{v}_1, \mathbf{A}\mathbf{v}_1), \quad (11.141)$$

$$h_{21} := \|\hat{\mathbf{v}}_2\|_2, \quad \mathbf{v}_2 := \hat{\mathbf{v}}_2/h_{21}. \quad (11.142)$$

Wir setzen hier und im Folgenden stillschweigend voraus, dass die Normierungskonstanten von null verschieden sind. Wir werden später die Bedeutung des Ausnahmefalles analysieren. Weiter halten wir für spätere Zwecke die aus (11.141) und (11.142) folgende Relation zwischen den Basisvektoren  $\mathbf{v}_1$  und  $\mathbf{v}_2$  fest:

$$\mathbf{A}\mathbf{v}_1 = h_{11}\mathbf{v}_1 + h_{21}\mathbf{v}_2. \quad (11.143)$$

Der Vektor  $\mathbf{z} \in S_2 = \mathcal{K}^{(2)}(\mathbf{r}^{(0)}, \mathbf{A}) = \mathcal{K}^{(2)}(\mathbf{v}_1, \mathbf{A})$  kann im zweiten Schritt des Minimierungsverfahrens (11.137) als Linearkombination der beiden Basisvektoren  $\mathbf{v}_1$  und  $\mathbf{v}_2$  dargestellt werden, und für den Residuen-Vektor  $\mathbf{r}^{(2)}$  ergibt sich so

$$\mathbf{r}^{(2)} = \mathbf{A}(c_1\mathbf{v}_1 + c_2\mathbf{v}_2) + \mathbf{r}^{(0)} = \beta\mathbf{v}_1 + c_1\mathbf{A}\mathbf{v}_1 + c_2\mathbf{A}\mathbf{v}_2.$$

Da  $\mathbf{v}_1, \mathbf{A}\mathbf{v}_1 \in \mathcal{K}^{(2)}(\mathbf{v}_1, \mathbf{A})$ , aber  $\mathbf{r}^{(2)} \in \mathcal{K}^{(3)}(\mathbf{v}_1, \mathbf{A})$  gilt, bedeutet dies, dass der Vektor  $\mathbf{A}\mathbf{v}_2$  zur Konstruktion des dritten orthonormierten Basisvektors  $\mathbf{v}_3$  verwendet werden kann. Der betreffende Orthonormierungsschritt lautet somit

$$h_{12} := (\mathbf{v}_1, \mathbf{A}\mathbf{v}_2), \quad h_{22} := (\mathbf{v}_2, \mathbf{A}\mathbf{v}_2), \quad (11.144)$$

$$\hat{\mathbf{v}}_3 := \mathbf{A}\mathbf{v}_2 - h_{12}\mathbf{v}_1 - h_{22}\mathbf{v}_2, \quad (11.145)$$

$$h_{32} := \|\hat{\mathbf{v}}_3\|_2, \quad \mathbf{v}_3 := \hat{\mathbf{v}}_3/h_{32}. \quad (11.146)$$

Daraus folgt die weitere Relation zwischen den ersten drei Basisvektoren

$$\mathbf{A}\mathbf{v}_2 = h_{12}\mathbf{v}_1 + h_{22}\mathbf{v}_2 + h_{32}\mathbf{v}_3. \quad (11.147)$$

Die Verallgemeinerung auf den  $(k+1)$ -ten Orthogonalisierungsschritt liegt jetzt auf der Hand. Wegen der Darstellung des Residuen-Vektors

$$\mathbf{r}^{(k)} = \mathbf{A} \left( \sum_{i=1}^k c_i \mathbf{v}_i \right) + \mathbf{r}^{(0)} = \beta\mathbf{v}_1 + \sum_{i=1}^{k-1} c_i (\mathbf{A}\mathbf{v}_i) + c_k \mathbf{A}\mathbf{v}_k$$

ist es gleichwertig, anstelle von  $\mathbf{r}^{(k)}$  den Vektor  $\mathbf{A}\mathbf{v}_k$  gegen  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$  zu orthonormieren, weil der erste Anteil Element des Krylov-Unterraumes  $\mathcal{K}^{(k)}(\mathbf{v}_1, \mathbf{A})$  ist. Deshalb beschreibt sich die Berechnung von  $\mathbf{v}_{k+1}$  wie folgt:

$$h_{ik} := (\mathbf{v}_i, \mathbf{A}\mathbf{v}_k), \quad i = 1, 2, \dots, k, \quad (11.148)$$

$$\hat{\mathbf{v}}_{k+1} := \mathbf{A}\mathbf{v}_k - \sum_{i=1}^k h_{ik} \mathbf{v}_i \quad (11.149)$$

$$h_{k+1,k} := \|\hat{\mathbf{v}}_{k+1}\|_2, \quad \mathbf{v}_{k+1} := \hat{\mathbf{v}}_{k+1}/h_{k+1,k} \quad (11.150)$$

Allgemein gilt die Relation zwischen den Basisvektoren

$$\mathbf{A}\mathbf{v}_k = \sum_{i=1}^{k+1} h_{ik} \mathbf{v}_i, \quad k = 1, 2, 3, \dots \quad (11.151)$$

Mit den ersten  $k$  Basisvektoren  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k \in \mathcal{K}^{(k)}(\mathbf{v}_1, \mathbf{A})$  bilden wir einerseits die Matrix

$$\mathbf{V}_k := (\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k) \in \mathbb{R}^{n,k} \quad \text{mit } \mathbf{V}_k^T \mathbf{V}_k = \mathbf{I}_k \quad (11.152)$$

und andererseits mit den beim Orthogonalisierungsprozess anfallenden Konstanten  $h_{ij}$  die Matrix

$$\mathbf{H}_k := \begin{pmatrix} h_{11} & h_{12} & h_{13} & \cdots & h_{1k} \\ h_{21} & h_{22} & h_{23} & \cdots & h_{2k} \\ 0 & h_{32} & h_{33} & \cdots & h_{3k} \\ 0 & 0 & h_{43} & \cdots & h_{4k} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & h_{k+1,k} \end{pmatrix} \in \mathbb{R}^{(k+1),k}. \quad (11.153)$$

Die Relationen (11.151) sind damit äquivalent zu

$$\mathbf{A}\mathbf{V}_k = \mathbf{V}_{k+1}\mathbf{H}_k, \quad k = 1, 2, 3, \dots \quad (11.154)$$

Nach dieser Vorbereitung kehren wir zurück zur Minimierungsaufgabe (11.137) im  $k$ -ten Iterationsschritt. Wegen  $\mathbf{z} \in S_k = \mathcal{K}^{(k)}(\mathbf{v}_1, \mathbf{A})$  kann dieser Vektor mit  $\mathbf{V}_k$  und einem Vektor  $\mathbf{c} \in \mathbb{R}^k$  dargestellt werden als

$$\mathbf{z} = \mathbf{V}_k \mathbf{c}. \quad (11.155)$$

Das zu minimierende Funktional lautet damit, wenn (11.154), (11.140), dann  $\mathbf{v}_1 = \mathbf{V}_{k+1} \mathbf{e}_1$  mit  $\mathbf{e}_1 = (1, 0, 0, \dots, 0)^T \in \mathbb{R}^{k+1}$  und schließlich die Tatsache benutzt wird, dass die Spaltenvektoren von  $\mathbf{V}_{k+1}$  orthonormiert sind,

$$\begin{aligned} \|\mathbf{A}\mathbf{z} + \mathbf{r}^{(0)}\|_2^2 &= \|\mathbf{A}\mathbf{V}_k \mathbf{c} + \mathbf{r}^{(0)}\|_2^2 = \|\mathbf{V}_{k+1} \mathbf{H}_k \mathbf{c} + \beta \mathbf{v}_1\|_2^2 \\ &= \|\mathbf{V}_{k+1}(\mathbf{H}_k \mathbf{c} + \beta \mathbf{e}_1)\|_2^2 = \|\mathbf{H}_k \mathbf{c} + \beta \mathbf{e}_1\|_2^2. \end{aligned} \quad (11.156)$$

Der Vektor  $\mathbf{c}$  ist somit Lösung des Fehlergleichungssystems

$$\mathbf{H}_k \mathbf{c} + \beta \mathbf{e}_1 = \mathbf{f} \quad (11.157)$$

nach der Methode der kleinsten Quadrate, und das Quadrat der Norm des Residuen-Vektors  $\mathbf{r}^{(k)}$  ist gleich dem minimalen Längenquadrat des Fehlervektors  $\mathbf{f}$ , d.h. es gilt

$$\|\mathbf{r}^{(k)}\|_2^2 = \|\mathbf{f}\|_2^2. \quad (11.158)$$

Das zu behandelnde Fehlergleichungssystem (11.157) hat für  $k = 4$  die Gestalt

$$\begin{aligned} h_{11}c_1 + h_{12}c_2 + h_{13}c_3 + h_{14}c_4 &+ \beta = f_1 \\ h_{21}c_1 + h_{22}c_2 + h_{23}c_3 + h_{24}c_4 &= f_2 \\ h_{32}c_2 + h_{33}c_3 + h_{34}c_4 &= f_3 \\ h_{43}c_3 + h_{44}c_4 &= f_4 \\ h_{54}c_4 &= f_5 \end{aligned} \quad (11.159)$$

Es wird effizient mit der Methode der Orthogonaltransformation vermittels Givens-Transformationen nach Abschnitt 6.2.1 behandelt unter Beachtung der speziellen Struktur.

Die Grundidee der Methode der verallgemeinerten minimierten Residuen ist damit vollständig beschrieben. Es wird sukzessive die orthonormierte Basis in der Folge der Krylov-Uterräume  $\mathcal{K}^{(k)}(\mathbf{v}_1, \mathbf{A})$  aufgebaut, in jedem Schritt das Fehlergleichungssystem (11.157) nach  $\mathbf{c} \in \mathbb{R}^k$  gelöst, womit durch den Vektor  $\mathbf{z} = \mathbf{V}_k \mathbf{c}$  der iterierte Vektor  $\mathbf{x}^{(k)}$  (11.136) mit dem zugehörigen minimalen Residuen-Vektor  $\mathbf{r}^{(k)}$  festgelegt ist.

### 11.5.2 Algorithmische Beschreibung und Eigenschaften

Die praktische Durchführung der QR-Zerlegung der Matrix  $\mathbf{H}_k = \mathbf{Q}_k \hat{\mathbf{R}}_k$  (6.24) zur Lösung der Fehlergleichungen (11.157) vereinfacht sich ganz wesentlich infolge der speziellen Struktur von  $\mathbf{H}_k$ . Zur Elimination der  $k$  Matrixelemente  $h_{21}, h_{32}, \dots, h_{k+1,k}$  wird nur die sukzessive Multiplikation von  $\mathbf{H}_k$  von links mit den  $k$  Rotationsmatrizen  $\mathbf{U}_1^T, \mathbf{U}_2^T, \dots, \mathbf{U}_k^T$  benötigt, wo  $\mathbf{U}_i^T$  durch das Rotationsindexpaar  $(i, i+1)$  mit geeignet gewähltem Winkel  $\varphi$  festgelegt ist. Die orthogonale Transformation von (11.157) ergibt mit

$$\mathbf{Q}_k^T := \mathbf{U}_k^T \mathbf{U}_{k-1}^T \dots \mathbf{U}_2^T \mathbf{U}_1^T : \quad \mathbf{Q}_k^T \mathbf{H}_k \mathbf{c}_k + \beta \mathbf{Q}_k^T \mathbf{e}_1 = \mathbf{Q}_k^T \mathbf{f}$$

oder

$$\hat{\mathbf{R}}_k \mathbf{c}_k + \hat{\mathbf{d}}_k = \hat{\mathbf{f}} \quad (11.160)$$

mit

$$\hat{\mathbf{R}}_k = \begin{pmatrix} \mathbf{R}_k \\ \mathbf{0}^T \end{pmatrix} \in \mathbb{R}^{(k+1),k}, \quad \hat{\mathbf{d}}_k = \begin{pmatrix} \mathbf{d}_k \\ \hat{d}_{k+1} \end{pmatrix} \in \mathbb{R}^{k+1}. \quad (11.161)$$

Der gesuchte Vektor  $\mathbf{c}_k \in \mathbb{R}^k$  ergibt sich durch Rücksubstitution aus

$$\mathbf{R}_k \mathbf{c}_k = -\mathbf{d}_k, \quad (11.162)$$

und wegen (11.158) gilt für das Normquadrat des Residuen-Vektors

$$\|\mathbf{r}^{(k)}\|_2^2 = \|\hat{\mathbf{f}}\|_2^2 = \hat{d}_{k+1}^2 = \mathbf{J}(\mathbf{x}^{(k)}). \quad (11.163)$$

Der Wert des minimierenden Funktionalen ergibt sich aus der Lösung des Fehlergleichungssystems aus der letzten Komponente der transformierten rechten Seite  $\hat{\mathbf{d}}_k = \beta \mathbf{Q}_k^T \mathbf{e}_1$ , ohne  $\mathbf{x}^{(k)}$  oder  $\mathbf{r}^{(k)}$  explizit zu berechnen.

Weiter ist zu beachten, dass sukzessive Fehlergleichungssysteme (11.157) für zunehmenden Index  $k$  zu lösen sind. Die Matrix  $\mathbf{H}_{k+1}$  geht aus  $\mathbf{H}_k$  durch Hinzufügen der  $(k+1)$ -ten Spalte mit den  $(k+2)$  Werten  $h_{i,k+1}$  und der  $(k+2)$ -ten Teilzeile mit Nullen hervor. Die ersten  $k$  oben genannten Transformationen mit  $\mathbf{U}_1^T, \mathbf{U}_2^T, \dots, \mathbf{U}_k^T$  sind aber dieselben für  $\mathbf{H}_{k+1}$ . Folglich genügt es, diese Rotationen auf die neu hinzukommende Spalte anzuwenden, dann aus dem erhaltenen transformierten Element  $h'_{k+1,k+1}$  und dem unveränderten Element  $h_{k+2,k+1}$  die Rotationsmatrix  $\mathbf{U}_{k+1}^T$  mit dem Winkel  $\varphi_{k+1}$  zu bestimmen, und diese Rotation auf die (transformierte) rechte Seite anzuwenden. Da in der erweiterten rechten Seite die  $(k+2)$ -te Komponente gleich null ist, ergibt sich wegen (5.12) für die letzte Komponente  $\hat{d}_{k+2}$  der transformierten rechten Seite  $\hat{\mathbf{d}}_{k+1}$

$$\hat{d}_{k+2} = \hat{d}_{k+1} \cdot \sin \varphi_{k+1}. \quad (11.164)$$

Die Abnahme des Normquadrates des Residuen-Vektors wird wegen (11.163) und (11.164) durch den Drehwinkel  $\varphi_{k+1}$  der letzten Rotation  $\mathbf{U}_{k+1}^T$  bestimmt. Daraus folgt, dass das

Funktional  $J(\mathbf{x}^{(k)})$  monoton abnimmt, wenn auch nur im schwachen Sinn. Die Situation  $J(\mathbf{x}^{(k+1)}) = J(\mathbf{x}^{(k)})$  kann wegen (11.164) dann eintreten, wenn  $|\sin \varphi_{k+1}| = 1$  ist, d.h. genau dann wenn wegen (5.60) der transformierte Wert  $h'_{k+1,k+1} = 0$  ist.

Aus dem Gesagten wird schließlich klar, dass man weder den Vektor  $\mathbf{c}_k$  aus (11.157) noch vermittels  $\mathbf{z}$  den iterierten Vektor  $\mathbf{x}^{(k)}$  zu berechnen braucht. Dies wird man erst dann tun, wenn  $\|\mathbf{r}^{(k)}\|_2^2 = \hat{d}_{k+1}^2$  genügend klein ist. Damit lautet die algorithmische Formulierung der GMRES-Methode:

Start: Wahl von $\mathbf{x}^{(0)}$ ; $\mathbf{r}^{(0)} = \mathbf{A}\mathbf{x}^{(0)} - \mathbf{b}$ ;	
$\beta = \ \mathbf{r}^{(0)}\ _2$ ; $\mathbf{v}_1 = \mathbf{r}^{(0)}/\beta$ ;	
Iteration: Für $k = 1, 2, 3, \dots$ :	
1. $\mathbf{z} = \mathbf{A}\mathbf{v}_k$	
2. $h_{ik} = (\mathbf{v}_i, \mathbf{z})$ , $i = 1, 2, \dots, k$ ,	
3. $\hat{\mathbf{v}}_{k+1} = \mathbf{z} - \sum_{i=1}^k h_{ik} \mathbf{v}_i$ (Orthogonalität)	(11.165)
4. $h_{k+1,k} = \ \hat{\mathbf{v}}_{k+1}\ _2$ ;	
$\mathbf{v}_{k+1} = \hat{\mathbf{v}}_{k+1}/h_{k+1,k}$ (Normierung)	
5. $\mathbf{H}_k \mathbf{c}_k + \beta \mathbf{e}_1 = \mathbf{f} \rightarrow \hat{d}_{k+1}$ (Nachführung)	
6. Falls $ \hat{d}_{k+1}  \leq \varepsilon \cdot \beta$ :	
$\mathbf{c}_k$ ; $\mathbf{z} = \mathbf{V}_k \mathbf{c}_k$ ; $\mathbf{x}^{(k)} = \mathbf{x}^{(0)} + \mathbf{z}$	
STOP	

Der GMRES-Algorithmus (11.165) ist problemlos durchführbar, falls  $h_{k+1,k} \neq 0$  gilt für alle  $k$ . Wir wollen untersuchen, was die Ausnahmesituation  $h_{k+1,k} = 0$  für den Algorithmus bedeutet unter der Annahme, dass  $h_{i+1,i} \neq 0$  gilt für  $i = 1, 2, \dots, k-1$ . Im  $k$ -ten Iterationsschritt ist also  $\hat{\mathbf{v}}_{k+1} = \mathbf{0}$ , und deshalb gilt die Relation

$$\mathbf{A}\mathbf{v}_k = \sum_{i=1}^k h_{ik} \mathbf{v}_i. \quad (11.166)$$

Die Vektoren  $\mathbf{v}_1, \dots, \mathbf{v}_k$  bilden eine orthonormierte Basis im Krylov-Unterraum  $\mathcal{K}^{(k)}(\mathbf{v}_1, \mathbf{A})$ . Da nach (11.166)  $\mathbf{A}\mathbf{v}_k$  Linearkombination dieser Basisvektoren ist, gilt

$$\mathcal{K}^{(k+1)}(\mathbf{v}_1, \mathbf{A}) = \mathcal{K}^{(k)}(\mathbf{v}_1, \mathbf{A}). \quad (11.167)$$

Anstelle von (11.154) gilt jetzt für diesen Index  $k$  die Matrizengleichung

$$\mathbf{A}\mathbf{V}_k = \mathbf{V}_k \hat{\mathbf{H}}_k, \quad (11.168)$$

wo  $\hat{\mathbf{H}}_k \in \mathbb{R}^{k,k}$  eine *Hessenberg-Matrix* ist, welche aus  $\mathbf{H}_k$  (11.153) durch Weglassen der letzten Zeile hervorgeht. Die Matrix  $\hat{\mathbf{H}}_k$  ist nicht zerfallend und ist regulär wegen (11.168), der Regularität von  $\mathbf{A}$  und des maximalen Rangs von  $\mathbf{V}_k$ . Die Minimierungsaufgabe (11.137) lautet wegen (11.168) mit dem Vektor  $\hat{\mathbf{e}}_1 \in \mathbb{R}^k$

$$\begin{aligned} \|\mathbf{r}^{(k)}\|_2^2 &= \|\mathbf{A}\mathbf{V}_k \mathbf{c}_k + \mathbf{r}^{(0)}\|_2^2 = \|\mathbf{V}_k (\hat{\mathbf{H}}_k \mathbf{c}_k + \beta \hat{\mathbf{e}}_1)\|_2^2 \\ &= \|\hat{\mathbf{H}}_k \mathbf{c}_k + \beta \hat{\mathbf{e}}_1\|_2^2 = \min! \end{aligned}$$

Das Minimum des Fehlerquadrates ist aber gleich null, weil gleich viele Unbekannte wie Fehleregleichungen vorhanden sind und das Gleichungssystem  $\mathbf{H}_k \mathbf{c}_k + \beta \hat{\mathbf{e}}_1 = \mathbf{0}$  eindeutig lösbar ist. Daraus ergibt sich die *Folgerung*: Bricht der GMRES-Algorithmus im  $k$ -ten Schritt mit  $h_{k+1,k} = 0$  ab, dann ist  $\mathbf{x}^{(k)}$  die gesuchte Lösung  $\mathbf{x}$  des Gleichungssystems.

Umgekehrt hat die Matrix  $\mathbf{H}_k \in \mathbb{R}^{(k+1),k}$  im Normalfall  $h_{i+1,i} \neq 0$  für  $i = 1, 2, \dots, k$  den maximalen Rang  $k$ , weil die Determinante, gebildet mit den  $k$  letzten Zeilen, ungleich null ist. Deshalb besitzt das Fehleregleichungssystem (11.157) eine eindeutige Lösung  $\mathbf{c}_k$  im Sinn der Methode der kleinsten Quadrate.

In Analogie zum CG-Verfahren gilt der

**Satz 11.21.** *Der GMRES-Algorithmus (11.165) liefert die Lösung des Minimalproblems (11.135) in der Form (11.157) nach höchstens  $n$  Iterationsschritten*

*Beweis.* Im  $\mathbb{R}^n$  existieren höchstens  $n$  orthonormierte Basisvektoren, und folglich muss spätestens  $\hat{\mathbf{v}}_{n+1} = \mathbf{0}$  und  $h_{n+1,n} = 0$  sein. Auf Grund der obigen Folgerung ist dann  $\mathbf{x}^{(n)} = \mathbf{x}$ .  $\square$

Der Satz 11.21 hat für das GMRES-Verfahren allerdings nur theoretische Bedeutung, weil zu seiner Durchführung alle beteiligten Basisvektoren  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$  abzuspeichern sind, da sie einerseits zur Berechnung der  $k$ -ten Spalte von  $\mathbf{H}_k$  im  $k$ -ten Iterationsschritt und andererseits zur Berechnung des Vektors  $\mathbf{x}^{(k)}$  nach Beendigung der Iteration benötigt werden. Der erforderliche Speicheraufwand im Extremfall von  $n$  Basisvektoren entspricht demjenigen einer vollbesetzten Matrix  $\mathbf{V} \in \mathbb{R}^{n,n}$ , und ist natürlich bei großem  $n$  nicht praktikabel und wenig sinnvoll.

Zudem wächst der Rechenaufwand eines einzelnen Iterationsschrittes des GMRES-Algorithmus (11.165) linear mit  $k$  an, da neben der Matrix-Multiplikation  $\mathbf{A}\mathbf{v}$  noch  $(k+1)$  Skalarprodukte und ebenso viele Multiplikationen von Vektoren mit einem Skalar erforderlich sind. Die Nachführung des Fehleregleichungssystems hat im Vergleich dazu nur einen untergeordneten, zu  $k$  proportionalen Aufwand. Der Rechenaufwand des  $k$ -ten Schrittes beträgt somit etwa

$$Z_{\text{GMRES}}^{(k)} = [\gamma + 2(k+1)]n$$

multiplikative Operationen.

Aus den genannten Gründen wird das Verfahren dahingehend modifiziert, dass man höchstens  $m \ll n$  Schritte durchführt, dann die resultierende Iterierte  $\mathbf{x}^{(m)}$  und den zugehörigen Residuen-Vektor  $\mathbf{r}^{(m)}$  berechnet und mit diesen als Startvektoren den Prozess neu startet. Die Zahl  $m$  im GMRES( $m$ )-Algorithmus richtet sich einerseits nach dem verfügbaren Speicherplatz für die Basisvektoren  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_m$ , und andererseits soll  $m$  unter dem Gesichtspunkt gewählt werden, den Gesamtrechenaufwand zu minimieren. Denn die erzielte Abnahme der Residuen-Norm rechtfertigt bei zu großem  $m$  den Aufwand nicht. Sinnvolle Werte für  $m$  liegen in der Regel zwischen 6 und 20.

Der prinzipielle Aufbau des GMRES( $m$ )-Algorithmus sieht wie folgt aus, falls die Anzahl der Neustarts maximal gleich *neust* sein soll, so dass maximal  $k_{\max} = m \cdot \text{neust}$  Iterationsschritte ausgeführt werden und die Iteration auf Grund des Kriteriums  $\|\mathbf{r}^{(k)}\|/\|\mathbf{r}^{(0)}\| \leq \varepsilon$

abgebrochen werden soll, wo  $\varepsilon$  eine vorzugebende Toleranz,  $\mathbf{r}^{(0)}$  den Residuen-Vektor der Anfangsstartnäherung und  $\mathbf{r}^{(k)}$  denjenigen der aktuellen Näherung  $\mathbf{x}^{(k)}$  bedeuten.

Start: Vorgabe von  $m, neust$ ; Wahl von  $\mathbf{x}^{(0)}$ .

Für  $l = 1, 2, \dots, neust$ :

$$\mathbf{r}^{(0)} = \mathbf{A}\mathbf{x}^{(0)} - \mathbf{b}; \quad \beta = \|\mathbf{r}^{(0)}\|_2; \quad \mathbf{v}_1 = \mathbf{r}^{(0)}/\beta;$$

$$\text{falls } l = 1: \quad \beta_0 = \beta$$

$$1. \quad \mathbf{z} = \mathbf{A}\mathbf{v}_k$$

$$2. \text{ für } i = 1, 2, \dots, k: \quad h_{ik} = (\mathbf{v}_i, \mathbf{z})$$

$$3. \quad \mathbf{z} = \mathbf{z} - \sum_{i=1}^k h_{ik} \mathbf{v}_i$$

$$4. \quad h_{k+1,k} = \|\mathbf{z}\|_2; \quad \mathbf{v}_{k+1} = \mathbf{z}/h_{k+1,k};$$

$$5. \quad \mathbf{H}_k \mathbf{c}_k + \beta \mathbf{e}_1 = \mathbf{f} \rightarrow \hat{\mathbf{d}}_{k+1} \quad (\text{Nachführung})$$

$$6. \text{ falls } |\hat{d}_{k+1}| \leq \varepsilon \cdot \beta_0:$$

$$\mathbf{R}_k \mathbf{c}_k = -\mathbf{d}_k \rightarrow \mathbf{c}_k;$$

$$\mathbf{x}^{(k)} = \mathbf{x}^{(0)} + \sum_{i=1}^k c_i \mathbf{v}_i;$$

STOP

$$\mathbf{R}_m \mathbf{c}_m = -\mathbf{d}_m \rightarrow \mathbf{c}_m;$$

$$\mathbf{x}^{(m)} = \mathbf{x}^{(0)} + \mathbf{V}_m \mathbf{c}_m \rightarrow \mathbf{x}^{(0)} \quad (\text{neuer Startvektor})$$

Keine Konvergenz!

(11.169)

Der theoretische Nachweis der Konvergenz des GMRES( $m$ )-Algorithmus scheint im allgemeinen Fall noch offen zu sein. Unter Zusatzannahmen für die Matrix  $\mathbf{A}$  und deren Spektrum existieren zum CG-Verfahren analoge Aussagen über die Abnahme des Quadrates der Residuen-Norm [Saa 81].

In der Rechenpraxis zeigt sich, dass das Funktional  $\mathbf{J}(\mathbf{x}^{(k)})$  (11.137) oft sehr langsam gegen null abnimmt, weil in (11.164)  $|\sin \varphi_{k+1}| \cong 1$  gilt. Die Konvergenz kann durch eine *Vorkonditionierung* wesentlich verbessert werden, wodurch das Verfahren erst zu einer effizienten iterativen Methode wird. Es wird analog zum CG-Verfahren eine *Vorkonditionierungsmatrix*  $\mathbf{M}$  gewählt, welche eine Approximation von  $\mathbf{A}$  sein soll. Selbstverständlich braucht jetzt  $\mathbf{M}$  nicht symmetrisch und positiv definit zu sein, aber noch regulär. Das zu lösende lineare Gleichungssystem  $\mathbf{Ax} = \mathbf{b}$  wird mit  $\mathbf{M}$  transformiert in das äquivalente System

$$\mathbf{M}^{-1} \mathbf{Ax} = \mathbf{M}^{-1} \mathbf{b} \Leftrightarrow \tilde{\mathbf{A}}\mathbf{x} = \tilde{\mathbf{b}} \quad (11.170)$$

für den unveränderten Lösungsvektor  $\mathbf{x}$ . Da im allgemeinen Fall keine Rücksicht auf Symmetrieeigenschaften genommen werden muss, verzichtet man darauf, den Algorithmus (11.169) umzuformen; stattdessen werden bei der Berechnung von  $\tilde{\mathbf{r}}^{(0)} = \mathbf{M}^{-1} \mathbf{r}^{(0)}$  und von  $\mathbf{z} = \tilde{\mathbf{A}}\mathbf{v} = \mathbf{M}^{-1} \mathbf{A}\mathbf{v}$  die entsprechenden Gleichungssysteme mit der Vorkonditionierungsmatrix  $\mathbf{M}$  nach  $\tilde{\mathbf{r}}^{(0)}$  bzw.  $\mathbf{z}$  aufgelöst.

Hat die nicht skalierte Matrix  $\mathbf{A}$  die Darstellung (11.14)  $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}$ , so ist

$$\mathbf{M} := (\mathbf{D} - \omega \mathbf{L}) \mathbf{D}^{-1} (\mathbf{D} - \omega \mathbf{U}) \quad (11.171)$$

eine zu (11.129) analoge, häufig angewandte Vorkonditionierungsmaatrix. Es kann auch eine Diagonalmatrix  $\tilde{\mathbf{D}}$  im Ansatz

$$\mathbf{M} = (\tilde{\mathbf{D}} - \mathbf{L}) \tilde{\mathbf{D}}^{-1} (\tilde{\mathbf{D}} - \mathbf{U}) \quad (11.172)$$

so bestimmt werden, dass beispielsweise  $\mathbf{M}$  und  $\mathbf{A}$  die gleichen Diagonalelemente haben. Aber auch andere, problemspezifische Festsetzungen von  $\mathbf{M}$  sind möglich [Yse 89, Fre 90].

**Beispiel 11.18.** Die Behandlung der elliptischen Randwertaufgabe

$$-\Delta u - \alpha u_x - \beta u_y = 2 \quad \text{in } G, \quad (11.173)$$

$$u = 0 \quad \text{auf } \Gamma, \quad (11.174)$$

mit dem Differenzenverfahren auf einem Rechteckgebiet  $G$  mit dem Rand  $\Gamma$  entsprechend dem Beispiel 11.7 führt infolge der ersten Ableitungen zwangsläufig auf ein lineares Gleichungssystem mit unsymmetrischer Matrix. Wird für  $-\Delta u$  die Fünf-Punkte-Differenzenapproximation verwendet und  $u_x$  und  $u_y$  durch die zentralen Differenzenquotienten

$$u_x \cong (u_E - u_W)/(2h), \quad u_y \cong (u_N - u_S)/(2h)$$

approximiert, so resultiert die Differenzengleichung in einem inneren Punkt  $P$

$$4u_p - (1 + 0.5\beta h)u_N - (1 - 0.5\alpha h)u_W - (1 - 0.5\beta h)u_S - (1 + 0.5\alpha h)u_E = 2h^2.$$

Tab. 11.7 Konvergenz und Aufwand des GMRES( $m$ )-Verfahrens,  $n = 23 \times 35 = 805$  Gitterpunkte.

$\omega$	$m = 6$		$m = 8$		$m = 10$		$m = 12$		$m = 15$	
	$n_{\text{it}}$	CPU								
0.0	170	95.6	139	84.4	120	79.0	129	90.9	139	109
1.00	53	47.3	51	47.2	55	52.7	51	51.5	59	63.9
1.20	42	37.6	50	46.4	42	40.8	44	44.1	38	40.3
1.40	42	37.6	32	29.8	32	31.2	29	29.0	27	28.8
1.60	26	23.7	23	21.5	23	22.4	22	22.2	21	22.1
1.70	21	19.2	20	18.7	19	18.6	20	20.0	19	20.3
1.75	21	19.2	22	20.5	20	19.7	20	20.0	20	21.2
1.80	23	20.8	23	21.5	23	22.4	22	22.2	22	23.1
1.85	27	24.4	25	23.8	25	24.1	24	24.7	25	26.3
1.90	31	28.3	30	27.9	30	29.3	30	30.0	29	31.5

Die resultierenden linearen Gleichungssysteme wurden für ein Rechteckgebiet  $G$  mit den Seitenlängen  $a = 3$  und  $b = 2$  mit  $\alpha = 5.0$  und  $\beta = 3.0$  für die Gitterweiten  $h = 1/12$  und  $h = 1/16$  mit dem GMRES( $m$ )-Algorithmus (11.169) bei Vorkonditionierung mittels der Matrix  $\mathbf{M}$  (11.171) gelöst. Die festgestellte Zahl der Iterationsschritte  $n_{\text{it}}$  und die Rechenzeiten CPU (Sekunden auf einem PS/2 - 55) sind in den Tabellen 11.7 und 11.8 in Abhängigkeit von  $m$  und  $\omega$  für  $\varepsilon = 10^{-8}$  auszugsweise zusammengestellt.

Die Ergebnisse zeigen, dass die angewandte Vorkonditionierung eine wesentliche Reduktion der Iterationsschritte im Vergleich zum nicht vorkonditionierten GMRES( $m$ )-Verfahren ( $\omega = 0$ ) bringt. Die Zahl der Iterationen in Abhängigkeit von  $\omega$  besitzt wieder ein relativ flaches Minimum in der

Tab. 11.8 Konvergenz und Aufwand des GMRES( $m$ )-Verfahrens,  $n = 31 \times 47 = 1457$  Gitterpunkte.

$\omega$	$m = 6$		$m = 8$		$m = 10$		$m = 12$		$m = 15$	
	$n_{\text{it}}$	CPU								
0.0	254	257	217	237	188	222	179	228	175	247
1.00	63	101	68	113	82	143	75	136	76	150
1.20	53	85.2	63	105	67	116	55	99.3	59	115
1.40	46	74.0	56	93.4	44	76.4	47	85.7	42	81.2
1.60	37	60.4	32	53.7	30	52.7	32	57.6	28	54.3
1.70	28	45.3	27	45.4	27	46.7	25	46.7	24	45.3
1.75	28	45.3	23	38.6	23	40.3	21	37.8	22	41.4
1.80	26	42.5	24	40.5	25	43.3	23	42.1	24	45.3
1.85	27	43.9	27	45.4	27	46.7	26	47.9	26	49.7
1.90	32	52.1	31	51.8	31	54.9	31	55.8	30	59.4

Gegend des optimalen Wertes. Der beste Wert von  $m$  scheint bei diesem Beispiel etwa bei 10 zu sein. Es ist auch zu erkennen, dass die Rechenzeit mit wachsendem  $m$  bei gleicher Iterationszahl  $n_{\text{it}}$  zunimmt als Folge des linear zunehmenden Rechenaufwandes pro Schritt bis zum Neustart. Es sei aber darauf hingewiesen, dass der optimale Wert von  $m$  stark von den zu lösenden Gleichungstypen abhängt. In der betrachteten Problemklasse bestimmten die Werte  $\alpha$  und  $\beta$  den Grad der Nichtsymmetrie der Matrix  $\mathbf{A}$  und beeinflussen den besten Wert  $m$  zur Minimierung der Rechenzeit.  $\triangle$

## 11.6 Speicherung schwach besetzter Matrizen

Die iterativen Verfahren erfordern die Multiplikation der Matrix  $\mathbf{A}$  mit einem Vektor oder im Fall der SOR-Verfahren im Wesentlichen die sukzessive Berechnung der Komponenten des Vektors  $\mathbf{A}\mathbf{z}$ . Zur Vorkonditionierung sind die Prozesse der Vorwärts- und Rücksubstitution für schwach besetzte Dreiecksmatrizen auszuführen. In allen Fällen werden nur die von null verschiedenen Matrixelemente benötigt, für die im folgenden eine mögliche Speicherung beschrieben wird, die zumindest auf Skalarrechnern eine effiziente Durchführung der genannten Operationen erlaubt.

Zur Definition einer symmetrischen Matrix  $\mathbf{A}$  genügt es, die von null verschiedenen Matrixelemente in und unterhalb der Diagonale abzuspeichern. Unter dem Gesichtspunkt der Speicherökonomie und des Zugriffs auf die Matrixelemente ist es zweckmäßig, diese Zahlenwerte zeilenweise, in kompakter Form in einem eindimensionalen Feld anzuordnen (vgl. Abb. 11.21). Ein zusätzliches, ebenso langes Feld mit den entsprechenden Spaltenindizes definiert die Position der Matrixelemente innerhalb der Zeile. Diese Spaltenindizes werden pro Zeile in aufsteigender Reihenfolge angeordnet, so dass das Diagonalelement als letztes einer jeden Zeile erscheint. Die  $n$  Komponenten eines Zeigervektors definieren die Enden der Zeilen und erlauben gleichzeitig einen Zugriff auf die Diagonalelemente der Matrix  $\mathbf{A}$ .

Um mit einer so definierten Matrix  $\mathbf{A}$  das SOR-Verfahren durchführen zu können, wird ein

Hilfsvektor  $\mathbf{y}$  definiert, dessen  $i$ -te Komponente die Teilsummen aus (11.13)

$$y_i := \sum_{j=i+1}^n a_{ij} x_j^{(k)} - b_i, \quad i = 1, 2, \dots, n,$$

sind. Diese können aus Symmetriegründen mit Hilfe der Nicht-Diagonalelemente unterhalb der Diagonale gebildet werden. Die iterierten Werte  $x_i^{(k+1)}$  berechnen sich anschließend nach (11.13) sukzessive für  $i = 1, 2, \dots, n$  in der Form

$$x_i^{(k+1)} = (1 - \omega)x_i^{(k)} - \omega \left[ \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} + y_i \right] / a_{ii},$$

wobei die Nicht-Diagonalelemente der  $i$ -ten Zeile und das entsprechende Diagonalelement auftreten. Der iterierte Vektor  $\mathbf{x}^{(k+1)}$  kann bei diesem Vorgehen am Platz von  $\mathbf{x}^{(k)}$  aufgebaut werden.

Die Matrix-Vektor-Multiplikation  $\mathbf{z} = \mathbf{A}\mathbf{p}$  der CG-Verfahren ist so realisierbar, dass mit jedem Nicht-Diagonalelement der unteren Hälfte zwei Multiplikationen mit zugehörigen Vektorkomponenten und die Addition zu entsprechenden Komponenten ausgeführt werden. Wenn man beachtet, dass die Spaltenindizes der  $i$ -ten Zeile stets kleiner als  $i$  sind, kann die

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{14} & a_{15} & & \\ a_{21} & a_{22} & a_{23} & & a_{26} & \\ & a_{32} & a_{33} & a_{35} & & \\ a_{41} & & a_{44} & & a_{46} & \\ a_{51} & & a_{53} & a_{55} & & \\ & a_{62} & a_{64} & & a_{66} & \end{pmatrix}$$

$$\mathbf{A}: \quad \boxed{\begin{array}{cccccccccc} a_{11} & | & a_{21} & a_{22} & | & a_{32} & a_{33} & | & a_{41} & a_{44} & | & a_{51} & a_{53} & | & a_{55} & | & a_{62} & a_{64} & a_{66} \end{array}}$$

$$k: \quad \boxed{\begin{array}{cccccccccc} 1 & | & 1 & 2 & | & 2 & 3 & | & 1 & 4 & | & 1 & 3 & | & 5 & | & 2 & 4 & 6 \end{array}}$$

$\xi:$  1 3 5 7 10 13

Abb. 11.21 Kompakte, zeilenweise Speicherung einer symmetrischen Matrix  $\mathbf{A}$ , untere Hälfte.

Operation  $\mathbf{z} = \mathbf{A}\mathbf{p}$  wie folgt realisiert werden:

$z_1 = a_1 \times p_1$ <p>für <math>i = 2, 3, \dots, n</math> :</p> $z_i = a_{\xi_i} \times p_i$ <p>für <math>j = \xi_{i-1} + 1, \xi_{i-1} + 2, \dots, \xi_i - 1</math> :</p> $z_i = z_i + a_j \times p_{k_j}$ $z_{k_j} = z_{k_j} + a_j \times p_i$	(11.175)
---	----------

Die Vorwärtssubstitution des Vorkonditionierungsschrittes mit der Matrix  $\mathbf{M}$  (11.129) ist mit den nach Abb. 11.21 gespeicherten Matrixelementen in offensichtlicher Weise realisierbar. Die Rücksubstitution ist so zu modifizieren, dass nach Berechnung der  $i$ -ten Komponente von  $\mathbf{q}$  das  $(\omega \cdot \varrho_i)$ -fache der  $i$ -ten Spalte von  $\mathbf{F}$ , d.h. der  $i$ -ten Zeile von  $\mathbf{E}$  von  $\mathbf{y}$  subtrahiert wird. Auf diese Weise kann wieder mit den aufeinander folgend gespeicherten Matrixelementen der  $i$ -ten Zeile gearbeitet werden [Sch 91b].

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & & a_{14} & \\ a_{21} & a_{22} & & & a_{25} \\ & a_{32} & a_{33} & & \\ & a_{42} & & a_{44} & a_{45} \\ & & a_{53} & & a_{55} \end{pmatrix}$$

$$\mathbf{A}: \quad \boxed{\begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|c|} \hline a_{11} & a_{21} & a_{22} & a_{32} & a_{33} & a_{42} & a_{44} & a_{53} & a_{55} & \parallel a_{12} & a_{14} & a_{25} & a_{45} \\ \hline \end{array}}$$

$$k: \quad \boxed{\begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|c|} \hline 1 & 1 & 2 & 2 & 3 & 2 & 4 & 3 & 5 & \parallel 2 & 4 & 5 & 5 \\ \hline \end{array}}$$

$$\xi: \quad \boxed{\begin{array}{ccccc|ccccc} 1 & 3 & 5 & 7 & 9 & 11 & 12 & 12 & 13 & 13 \end{array}}$$

Abb. 11.22 Kompakte Speicherung einer unsymmetrischen Matrix.

Im Fall einer unsymmetrischen Matrix  $\mathbf{A}$  sind selbstverständlich alle von null verschiedenen Matrixelemente abzuspeichern. Die nahe liegende Idee, die relevanten Matrixelemente zeilenweise kompakt in Verallgemeinerung zur Abb. 11.21 so in einem eindimensionalen Feld anzutragen, dass das Diagonalelement wiederum jeweils das letzte ist, wäre für das SOR-Verfahren und die Matrix-Vektor-Multiplikation  $\mathbf{z} = \mathbf{A}\mathbf{p}$  sicher geeignet. Sobald aber ein Vorkonditionierungsschritt mit einer Matrix  $\mathbf{M}$  (11.171) oder (11.172) auszuführen ist, erweist sich eine solche Anordnung als ungünstig. Es ist zweckmäßiger, die Nicht-Diagonalelemente oberhalb der Diagonale erst im Anschluss an die anderen Matrixelemente zu speichern und die Anordnung von Abb. 11.21 zu erweitern zu derjenigen von Abb. 11.22 einer unsymmetrisch besetzten Matrix  $\mathbf{A}$ . Der Zeigervektor ist um  $n$  Zeiger zu erweitern, welche die Plätze der letzten Nicht-Diagonalelemente oberhalb der Diagonale der einzelnen Zeilen definieren. Zwei aufeinander folgende Zeigerwerte mit dem gleichen Wert bedeuten, dass die betreffende Zeile kein von null verschiedenes Matrixelement oberhalb der Diagonale

aufweist. Die oben beschriebenen Operationen sind mit der jetzt verwendeten Speicherung noch einfacher zu implementieren.

## 11.7 Software

Software zu diesem Kapitel findet man oft unter dem Oberbegriff *Numerische lineare Algebra*, siehe Kapitel 2, oft auch im Zusammenhang mit der numerischen Lösung partieller Differenzialgleichungen, siehe Kapitel 10.

In der NAG-FORTRAN-Bibliothek gibt es im Kapitel F11 etwa 50 Routinen zur Lösung schwach besetzter linearer Gleichungssysteme, in der NAG-C-Bibliothek sind es nur sechzehn Routinen. Das frei erhältliche Paket SLATEC enthält noch mehr FORTRAN-Routinen; die meisten entstammen dem Paket SLAP, das wiederum auf Anne Greenbaums Programm sammlung für LAPACK [And 99] basiert. Auch in den NAG-Bibliotheken werden LAPACK-Routinen verwendet. Damit stimmen die Quellen für alle zuverlässigen Programme wieder überein.

Eine Autorengruppe, darunter einige der LAPACK-Autoren, hat (wie für die algebraischen Eigenwertprobleme, siehe Abschnitt 5.9) Templates für iterative Methoden [Bar 94] entwickelt.

Auch MATLAB verfügt über eine stattliche Sammlung von speziellen Routinen für schwach besetzte Matrizen. Neben der Lösung von großen linearen Gleichungssystemen mit zahlreichen Verfahren können auch Eigenwertprobleme gelöst oder Singulärwerte berechnet werden. Die Besetzungsstruktur der Matrix kann graphisch verdeutlicht werden. Schließlich können die Matrixstrukturen *voll* (*full*) und *schwach* (*sparse*) ineinander übergeführt werden, wenn der Speicherplatz es erlaubt.

Unsere Problemlöseumgebung PAN (<http://www.upb.de/SchwarzKoeckler/>) verfügt über ein Programm zur Lösung schwach besetzter linearer Gleichungssysteme mit einem iterativen Verfahren.

## 11.8 Aufgaben

**Aufgabe 11.1.** Die Matrix eines linearen Gleichungssystems ist

$$\mathbf{A} = \begin{pmatrix} 2 & -1 & -1 & 0 \\ -1 & 2.5 & 0 & -1 \\ -1 & 0 & 2.5 & -1 \\ 0 & -1 & -1 & 2 \end{pmatrix}.$$

Man zeige, dass  $\mathbf{A}$  irreduzibel ist, und dass das Gesamtschrittverfahren und das Einzelschrittverfahren konvergent sind. Wie groß sind die Spektralradien  $\varrho(\mathbf{T}_J)$  und  $\varrho(\mathbf{T}_{ES})$ , und welches sind die Anzahl der Iterationsschritte, welche zur Gewinnung einer weiteren richtigen Dezimalstelle einer Näherungslösung nötig sind?

**Aufgabe 11.2.** Man bestimme die Lösungen der beiden linearen Gleichungssysteme iterativ mit dem JOR- und dem SOR-Verfahren mit verschiedenen  $\omega$ -Werten. Wie groß sind die experimentell ermittelten optimalen Relaxationsparameter?

a)

$$\begin{aligned} 5x_1 - 3x_2 + 2x_4 &= 13 \\ 2x_1 + 6x_2 - 3x_3 &= 16 \\ -x_1 + 2x_2 + 4x_3 - x_4 &= -11 \\ -2x_1 - 3x_2 + 2x_3 + 7x_4 &= 10 \end{aligned}$$

b)

$$\begin{aligned} 4x_1 - x_2 - x_3 &= 6 \\ -x_1 + 4x_2 - 2x_4 &= 12 \\ -x_1 + 4x_3 - x_4 &= -3 \\ -2x_2 - x_3 + 4x_4 - x_5 &= -5 \\ -x_4 + 4x_5 &= 1 \end{aligned}$$

Warum sind das Gesamtschritt- und das Einzelschrittverfahren in beiden Fällen konvergent? Wie groß sind die Spektralradien der Iterationsmatrizen  $T_J$  und  $T_{ES}$ ? Welches ist die jeweilige Anzahl der notwendigen Iterationsschritte, um eine weitere Dezimalstelle in der Näherung zu gewinnen?

Im Fall des zweiten Gleichungssystems transformiere man die Matrix  $\mathbf{A}$  durch eine geeignete Permutation der Unbekannten und der Gleichungen auf die spezielle Blockstruktur

$$\mathbf{A} = \begin{pmatrix} \mathbf{D}_1 & \mathbf{H} \\ \mathbf{K} & \mathbf{D}_2 \end{pmatrix}$$

mit Diagonalmatrizen  $\mathbf{D}_1$  und  $\mathbf{D}_2$  wie in Beispiel 11.7. Welches sind die Eigenwerte der Iterationsmatrix des J-Verfahrens und der optimale Relaxationsparameter  $\omega_{opt}$  (11.46) des SOR-Verfahrens? Welche Reduktion des Rechenaufwandes wird mit dem optimalen  $\omega_{opt}$  des SOR-Verfahrens im Vergleich zum Einzelschrittverfahren erzielt?

**Aufgabe 11.3.** Die elliptische Randwertaufgabe

$$\begin{aligned} -\Delta u &= 1 && \text{in } G, \\ u &= 0 && \text{auf } \Gamma, \end{aligned}$$

für das Gebiet  $G$  und seinen Rand  $\Gamma$  von Abb. 11.23 soll mit dem Differenzenverfahren näherungsweise gelöst werden. Wie lautet das lineare Gleichungssystem für Gitterweiten  $h = 1$  und  $h = 0.5$  bei zeilenweiser oder schachbrettartiger Nummerierung der Gitterpunkte? Welche Blockstrukturen sind festzustellen? Die Gleichungssysteme sind mit dem Gesamtschrittverfahren und mit der Überrelaxation unter Verwendung von verschiedenen Relaxationsparametern  $\omega$  zu lösen, und der optimale Wert  $\omega_{opt}$  ist experimentell zu ermitteln.

Zur rechnerischen Realisierung der Iterationsverfahren für dieses Problem ist es zweckmäßig, die unbekannten Werte in den Gitterpunkten in einem zweidimensionalen Feld anzugeben unter Einbezug der Dirichletschen Randbedingungen. So können die Iterationsformeln mit Hilfe der benachbarten Werte in der Form von einfachen Anweisungen explizit formuliert werden. Auf diese Weise braucht weder die Matrix  $\mathbf{A}$  noch die rechte Seite  $\mathbf{b}$  definiert zu werden. Ein entsprechendes Computerprogramm ist dann leicht so zu konzipieren, dass die entsprechenden Gleichungssysteme für beliebige, natürlich zu  $G$  passende, Gitterweiten  $h$  bearbeitet werden können. Welches sind die optimalen  $\omega$ -Werte der SOR-Methode für kleinere Gitterweiten  $h$ ?

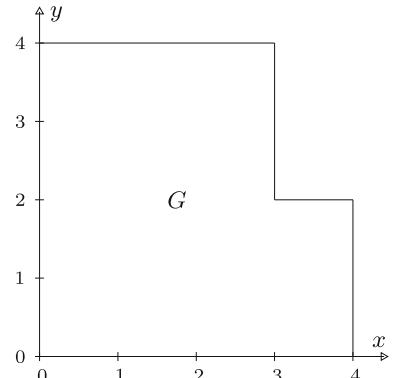


Abb. 11.23  
Gebiet  $G$  der Randwertaufgabe.

**Aufgabe 11.4.** Es sei  $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}$  eine Matrix mit positiven Diagonalelementen  $a_{ii}$  und  $\tilde{\mathbf{A}} = \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2} = \mathbf{I} - \tilde{\mathbf{L}} - \tilde{\mathbf{U}}$  die zugehörige skalierte Matrix mit Diagonalelementen  $\tilde{a}_{ii} = 1$ . Man zeige, dass die Spektralradien des J-Verfahrens und der SOR-Methode durch die Skalierung nicht beeinflusst werden, so dass gelten

$$\varrho(\mathbf{T}_J) = \varrho(\tilde{\mathbf{T}}_J), \quad \varrho(\mathbf{T}_{SOR}(\omega)) = \varrho(\tilde{\mathbf{T}}_{SOR}(\omega)).$$

**Aufgabe 11.5.** Die skalierte Matrix eines linearen Gleichungssystems

$$\mathbf{A} = \begin{pmatrix} 1 & -0.7 & 0 & 0.6 \\ 0.7 & 1 & -0.1 & 0 \\ 0 & 0.1 & 1 & 0.4 \\ -0.6 & 0 & -0.4 & 1 \end{pmatrix}$$

hat die Eigenschaft, dass der nicht-diagonale Anteil  $\mathbf{L} + \mathbf{U}$  schiefsymmetrisch ist. Welches sind die Eigenwerte der Iterationsmatrix  $\mathbf{T}_J$  des Gesamtschrittverfahrens und damit der Spektralradius  $\varrho(\mathbf{T}_J)$ ? Mit Hilfe einer geometrischen Überlegung ermittle man denjenigen Wert von  $\omega$ , für welchen das JOR-Verfahren optimale Konvergenz aufweist. In welchem Verhältnis steht die optimale Konvergenzrate des JOR-Verfahrens zu derjenigen des J-Verfahrens und wie groß ist die Reduktion des Rechenaufwandes?

**Aufgabe 11.6.** Der Aliasing-Effekt: Man zeige, dass auf einem homogenen eindimensionalen Gitter mit  $n-1$  inneren Punkten eine Schwingung  $w_{k,j} = \sin(jk\pi/n)$  mit  $n < k < 2n$  die Oszillationsseigenschaft der zu Grunde liegenden Sinus-Funktion nicht diskret übernimmt, sondern dass der Vektor  $\mathbf{w}_k$  statt  $k-1$  nur  $k'-1$  Vorzeichenwechsel hat mit  $k' = 2n-k$ .  $\mathbf{w}_k$  stellt damit die diskrete Schwingung  $w_{k',j} = \sin(jk'\pi/n)$  dar.

**Aufgabe 11.7.** Man füllt die Lücken bei der Herleitung der Gleichungen (11.78), (11.79) und (11.80) für das Gauß-Seidel-Verfahren in Beispiel 11.14.

**Aufgabe 11.8.** Man berechne die Verstärkungsfaktoren  $G(\theta_1, \theta_2)$  für das gedämpfte Jacobi- und für das Gauß-Seidel-Verfahren für den Fall, dass der zweidimensionale Operator  $-\Delta u$  mit einem Neun-Punkte-Stern

$$\frac{1}{3h^2} \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

diskretisiert wird.

**Aufgabe 11.9.** Eine Variationseigenschaft, die wir nicht kennen gelernt haben, ist die Beziehung zwischen zwei Matrizen, die die Differenzialgleichung diskretisieren, auf unterschiedlichen Gittern. Danach definiert man  $A^{2h}$  auf dem groben Gitter als

$$A^{2h} = I_h^{2h} A^h I_{2h}^h. \quad (11.176)$$

Man nehme an, dass der zweidimensionale Operator  $-\Delta u$  mit dem üblichen Fünf-Punkte-Stern

$$\frac{1}{h^2} \begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

diskretisiert wird. Man bestimme den Stern für einen randfernen inneren Punkt des Gitters mit der Gitterweite  $2h$  über die Gleichung (11.176), wenn  $I_h^{2h}$  und  $I_{2h}^h$  die Matrizen für die FW-Restriktion und die lineare Interpolation sind.

Man bestimme in entsprechender Weise den Stern zur Gitterweite  $2h$ , wenn die Matrix  $A^h$  durch Diskretisierung mit dem Neun-Punkte-Stern aus Aufgabe 11.8 entstanden ist.

**Aufgabe 11.10.** Mit Hilfe eines Computerprogramms löse man die linearen Gleichungssysteme von Aufgabe 11.3 mit der Methode der konjugierten Gradienten ohne und mit Vorkonditionierung (SSORCG) für verschiedene Gitterweiten  $h = 1, 1/2, 1/4, 1/6, \dots$ . Zu diesem Zweck sind die zugehörigen Matrizen in kompakter zeilenweiser Speicherung und die rechten Seiten zu generieren. Wie steigt die Zahl der nötigen Iterationsschritte bei Verfeinerung der Gitterweite  $h$  an und welche Reduktion des Rechenaufwandes wird mit der Vorkonditionierung bei optimaler Wahl von  $\omega$  erreicht?

**Aufgabe 11.11.** Man entwickle ein Programm zum GMRES( $m$ )-Algorithmus (11.169) ohne und mit Vorkonditionierung.

Damit löse man die unsymmetrischen Gleichungssysteme der Aufgaben 10.2 und 10.3. Zudem bestimme man die Näherungslösung der Randwertaufgabe

$$\begin{aligned} -\Delta u + 4u_x - 3u_y &= 2 && \text{in } G, \\ u &= 0 && \text{auf } \Gamma, \end{aligned}$$

wo  $G$  das trapezförmige Gebiet der Abb. 11.19 und  $\Gamma$  sein Rand bedeuten und die Aufgabe mit dem Differenzenverfahren für die Gitterweiten  $h = 1/8, 1/12, 1/16, 1/24/1/32/1/48$  diskretisiert wird. Für welche Werte  $m$  und  $\omega$  im Fall der Vorkonditionierungsmatrix  $\mathbf{M}$  (11.129) sind die Rechenzeiten minimal? Durch Variation der beiden Konstanten der ersten partiellen Ableitungen stelle man den Einfluss des resultierenden Grades der Unsymmetrie der Gleichungssysteme auf das Konvergenzverhalten fest.

## Literaturverzeichnis

- [Abr 71] Abramowitz, M./ Stegun, I. A.: Handbook of mathematical functions. New York: Dover Publications 1971
- [Aik 85] Aiken, R. C. (Hrsg.): Stiff computation. New York-Oxford: Oxford University Press 1985
- [Akh 88] Akhiezer, N.: The calculus of variations. Harwood: Harwood Academic Publishers 1988
- [Alb 85] Albrecht, P.: Numerische Behandlung gewöhnlicher Differenzialgleichungen. München: Hanser 1985
- [Ale 02] Alefeld, G./ Lenhardt, I./ Obermaier, H.: Parallele numerische Verfahren. Berlin: Springer 2002
- [Ama 95] Amann, H.: Gewöhnliche Differenzialgleichungen, 2. Aufl. Berlin-New York: de Gruyter 1995
- [And 99] Anderson, E./ Bai, Z./ Bischof, C./ Blackford, S./ Demmel, J./ Dongarra, J. J./ Du Croz, J./ Greenbaum, A./ Hammarling, S./ McKenney, A./ Sorensen, D.: LAPACK User's Guide – 3rd ed. Philadelphia: SIAM 1999
- [Asc 95] Ascher, U. M./ Mattheij, R. M. M./ Russell, R. D.: Numerical solution of boundary value problems for ordinary differential equations. Englewood Cliffs: Prentice-Hall 1995
- [Axe 84] Axelsson, O./ Barker, V. A.: Finite element solution of boundary value problems. New York: Academic Press 1984
- [Axe 85] Axelsson, O.: A survey of preconditioned iterative methods for linear systems of algebraic equations. BIT **25** (1985) 166–187
- [Axe 89] Axelsson, O. (Hrsg.): Preconditioned conjugate gradient methods. Special issue of BIT **29:4** 1989
- [Axe 90] Axelsson, O./ Kolotilina, L. Y. (Hrsg.): Preconditioned conjugate gradient methods. Berlin: Springer 1990
- [Bai 00] Bai, Z./ Demmel, J./ Dongarra, J. J./ Ruhe, A./ van der Vorst, H.: Templates for the solution of algebraic eigenvalue problems: A practical guide. Philadelphia: SIAM 2000
- [Ban 98] Bank, R. E.: PLTMG: A software package for solving elliptic partial differential Equations: Users' Guide 8.0. Philadelphia: SIAM 1998
- [Bar 88a] Barragy, E./ Carey, G. F.: A parallel element by element solution scheme. Int. J. Numer. Meth. Engin. **26** (1988) 2367–2382
- [Bar 88b] Barsky, B. A.: Computer graphics and geometric modeling using Beta-Splines. Berlin: Springer 1988
- [Bar 94] Barrett, R./ Berry, M./ Chan, T. F./ Demmel, J./ Donato, J./ Dongarra, J./

- Eijkhout, V./ Pozo, R./ Romine, C./ van der Vorst, H.: Templates for the solution of linear systems: Building blocks for iterative methods, 2nd ed. Philadelphia: SIAM 1994
- [Bar 95] Bartels, R. H./ Beatty, J. C./ Barsky, B. A.: An introduction to splines for use in computer graphics and geometric modeling. Los Altos: Kaufmann 1995
- [Beno 24] Benoit: Note sur une méthode de résolution des équations normales etc. (Procédé du commandant Cholesky). Bull. géodésique **3** (1924) 67–77
- [Bey 98] Bey, J.: Finite-Volumen- und Mehrgitter-Verfahren für elliptische Randwertprobleme. Stuttgart: Teubner 1998
- [Bol 04] Bollhöfer, M./ Mehrmann, V.: Numerische Mathematik. Wiesbaden: Vieweg 2004
- [Bon 91] Bondeli, S.: Divide and Conquer: a parallel algorithm for the solution of a tridiagonal linear system of equations. Parallel Comp. **17** (1991) 419–434
- [Boor 80] Boor, C. de: FFT as nested multiplication, with a twist. SIAM J. Sci. Stat. Comp. **1** (1980) 173–178
- [Boor 01] Boor, C. de: A practical guide to splines, 2nd ed. New York: Springer 2001
- [Bram 68] Bramble, J. H./ Hubbard, B. E./ Ziamal, M.: Discrete analogues of the Dirichlet problem with isolated singularities. SIAM J. Numer. Anal. **5** (1968) 1–25
- [Bram 86] Bramble, J. H./ Pasciak, J. E./ Schatz, A. H.: The construction of preconditioners for elliptic problems by substructuring I. Math. Comp. **47** (1986) 103–134
- [Bram 93] Bramble, J. H.: Multigrid methods. Harlow: Longman 1993
- [Braess 03] Braess, D.: Finite Elemente: Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie, 3rd ed. Berlin: Springer 2003
- [Bre 71] Brent, R. P.: An algorithm with guaranteed convergence for finding a zero of a function. Comp. J. **14** (1971) 422–425
- [Bre 92] Brehm, J.: Parallele lineare Algebra. Wiesbaden: Dt. Univ.-Verlag 1992
- [Bre 02] Brent, R. P.: Algorithm for minimization without derivatives. Mineola: Dover Publications 2002
- [Bri 95] Brigham, E. O.: FFT: Schnelle Fourier-Transformation, 6. Aufl. München: Oldenbourg 1995
- [Bri 00] Briggs, W. L./ Henson, V. E./ McCormick, S. F.: A multigrid tutorial, 2nd ed. Philadelphia: SIAM 2000
- [Bru 95] Bruaset, A. M.: A survey of preconditioned iterative methods. Harlow: Longman 1995
- [Bun 76] Bunch, J. R./ Rose, D. J. (Hrsg.): Sparse matrix computations. New York: Academic Press 1976
- [Bun 95] Bunse, W./ Bunse-Gerstner, A.: Numerische lineare Algebra. Stuttgart: Teubner 1995
- [Bun 02] Bungartz, H./ Griebel, M./ Zenger, C.: Einführung in die Computergraphik. Wiesbaden: Vieweg 2002
- [Butcher 63] Butcher, J. C.: Coefficients for the study of Runge-Kutta integration processes. J. Austral. Math. Soc. **3** (1963) 185–201
- [Butcher 65] Butcher, J. C.: On the attainable order of Runge-Kutta methods. Math. Comp. **19** (1965) 408–417
- [Butcher 87] Butcher, J. C.: The numerical analysis of ordinary differential equations. Chichester: John Wiley 1987

- [Ces 66] Ceschino, F./ Kuntzmann, J.: Numerical solution of initial value problems. Englewood Cliffs: Prentice-Hall 1966
- [Cha 82] Chan, T. F.: An improved algorithm for computing the singular value decomposition. *ACM Trans. Math. Soft.* **8** (1982) 72–88
- [Cha 89] Chan, T. F./ Glowinski, R./ Péliaux, J./ Widlund, O. B. (Hrsg.): Proceedings of the second international symposium on domain decomposition methods. Philadelphia: SIAM 1989
- [Cho 95] Choi, J./ Dongarra, J. J./ Ostrouchov, S./ Petite, A./ Walker, D./ Whaley, R. C.: A proposal for a set of parallel basic linear algebra subprograms. Working note, LAPACK 1995
- [Cho 96] Choi, J./ Dongarra, J. J./ Ostrouchov, S./ Petite, A./ Walker, D./ Whaley, R. C.: The design and implementation of the SCALAPACK LU, QR and Cholesky factorization routines, Vol. 5. LAPACK working note 80, Oak Ridge National Laboratory 1996
- [Cia 02] Ciarlet, P. G.: The finite element method for elliptic problems. Philadelphia: SIAM 2002
- [Cle 55] Clenshaw, C. W.: A note on the summation of Chebyshev series. *Math. Tab. Wash.* **9** (1955) 118–120
- [Cli 79] Cline, A. K./ Moler, C. B./ Stewart, G. W./ Wilkinson, J. H.: An estimate for the condition number of a matrix. *SIAM J. Numer. Anal.* **16** (1979) 368–375
- [Col 66] Collatz, L.: The numerical treatment of differential equations. Berlin: Springer 1966
- [Col 68] Collatz, L.: Funktionalanalysis und numerische Mathematik. Berlin: Springer 1968
- [Col 90] Collatz, L.: Differentialgleichungen 7. Aufl. Stuttgart: Teubner 1990
- [Con 85] Concus, P./ Golub, G./ Meurant, G.: Block preconditioning for the conjugate gradient method. *SIAM J. Sci. Comp.* **6** (1985) 220–252
- [Coo 65] Cooley, J. W./ Tukey, J. W.: An algorithm for the machine calculation of complex Fourier series. *Math. Comp.* **19** (1965) 297–301
- [Cos 95] Cosnard, M./ Trystam, D.: Parallel algorithms and architectures. New York: Thomson 1995
- [Cou 93] Courant, R./ Hilbert, D.: Methoden der mathematischen Physik, 4. Aufl. (1. Aufl. 1924/1937). Berlin: Springer 1993
- [Cri 86] Crisfield, M. A.: Finite elements and solution procedures for structural analysis, linear analysis. Swansea: Pineridge Press 1986
- [Cul 99] Culler, D. E./ Singh, J. P./ Gupta, A.: Parallel computer architecture: a hardware, software approach. San Francisco: Morgan Kaufmann 1999
- [Dah 85] Dahlquist, G.: 33 years of numerical instability, Part 1. *BIT* **25** (1985) 188–204
- [Dah 03] Dahlquist, G./ Björck, Å.: Numerical methods. Mineola: Dover Publications 2003
- [Dek 84] Dekker, K./ Verwer, J. G.: Stability of Runge-Kutta methods for stiff nonlinear differential equations. Amsterdam: North-Holland 1984
- [Deu 08a] Deuflhard, P./ Bornemann, F.: Numerische Mathematik II, 3. Aufl. Berlin: de Gruyter 2008
- [Deu 08b] Deuflhard, P./ Hohmann, A.: Numerische Mathematik I. Eine algorithmisch orientierte Einführung, 4. Aufl. Berlin: de Gruyter 2008
- [Dew 86] Dew, P. M./ James, K. R.: Introduction to numerical computation in PASCAL.

- New York: Springer 1986
- [Dew 88] Dewey, B. R.: Computer graphics for engineers. New York: Harper and Row 1988
- [Die 89] Dierckx, P.: FITPACK User Guide. Part 1: Curve fitting routines. Report TW 89. Part 2: Surface fitting routines. Report TW 122. Katholieke Universiteit Leuven, department of computer Sscience Celestijnenlaan 200A, B-3030 Leuven (Belgium) 1987, 1989
- [Die 06] Dierckx, P.: Curve and surface fitting with splines. Oxford: Clarendon Press 2006
- [Don 79] Dongarra, J. J. / Du Croz, J. / Duff, I. S. / Hammarling, S.: LINPACK User's Guide. Philadelphia: SIAM 1979
- [Don 93] Dongarra, J. J. / Duff, I. S. / Sorensen, D. C. / van der Vorst, H. A.: Solving linear systems on vector and shared memory computers. Philadelphia: SIAM 1993
- [Dor 78] Dormand, J. R. / Prince, P. J.: New Runge-Kutta algorithms for numerical simulation in dynamical astronomy. *Celestial Mechanics* **18** (1978) 223–232
- [Dor 80] Dormand, J. R. / J., P. P.: A family of embedded Runge-Kutta formulas. *J. Comp. Appl. Math.* (1980) 19–26
- [Duf 76] Duff, I. S. / Reid, J. K.: A comparison of some methods for the solution of sparse over-determined systems of linear equations. *J. Inst. Math. Applic.* **17** (1976) 267–280
- [Duf 86] Duff, I. S. / Erisman, A. M. / Reid, J. K.: Direct methods for sparse matrices. Oxford: Clarendon Press 1986
- [EM 90] Engeln-Müllges, G. / Reutter, F.: Formelsammlung zur numerischen Mathematik mit C-Programmen 2. Auflage. Mannheim: B.I. Wissenschaftsverlag 1990
- [Enc 96] Encarnaçao, J. / Straßer, W. / Klein, R.: Graphische Datenverarbeitung, 4. Auflage. München: Oldenbourg 1996
- [Eng 69] England, R.: Error estimates for Runge-Kutta type solutions to systems of ordinary differential equations. *Comp. J.* **12** (1969) 166–170
- [Eva 83] Evans, D. J. (Hrsg.): Preconditioning methods: Analysis and applications. New York: Gordon and Breach 1983
- [Far 94] Farin, G.: Kurven und Flächen im Computer Aided Geometric Design 2. Aufl. Wiesbaden: Vieweg 1994
- [Fat 88] Fatunla, S. O.: Numerical methods for initial value problems in ordinary differential equations. London: Academic Press 1988
- [Feh 64] Fehlberg, E.: New high-order Runge-Kutta formulas with step size control for systems of first and second order differential equations. *ZAMM* **44** (1964) T17–T29
- [Feh 68] Fehlberg, E.: Classical fifth-, sixth-, seventh-, and eighth order Runge-Kutta formulas with step size control. *NASA Techn. Rep.* 287 (1968)
- [Feh 69a] Fehlberg, E.: Klassische Runge-Kutta-Formeln fünfter und siebenter Ordnung mit Schrittweiten-Kontrolle. *Computing* **4** (1969) 93–106
- [Feh 69b] Fehlberg, E.: Low-order classical Runge-Kutta formulas with step size control and their application to some heat transfer problems. *NASA Techn. Rep.* 315 (1969)
- [Feh 70] Fehlberg, E.: Klassische Runge-Kutta-Formeln vierter und niedrigerer Ordnung mit Schrittweiten-Kontrolle und ihre Anwendung auf Wärmeleitungsprobleme. *Computing* **6** (1970) 61–71
- [Fle 00] Fletcher, R.: Practical methods of optimization. Vol. 1: Unconstrained Optimiza-

- tion. Vol. 2: Constrained Optimization., 2nd ed. Chichester: Wiley 2000
- [For 77] Forsythe, G. E./ Malcolm, M. A./ Moler, C. B.: Computer methods for mathematical computations. Englewood Cliffs: Prentice-Hall 1977
- [Fox 79] Fox, L./ Parker, I. B.: Chebyshev polynomials in numerical analysis. London: Oxford University Press 1979
- [Fox 88] Fox, L./ Mayers, D. F.: Numerical solution of ordinary differential equations. London: Chapman and Hall 1988
- [Fra 62] Francis, J.: The QR transformation. A unitary analogue to the LR transformation, Parts I and II. *Comp. J.* **4** (1961/62) 265–271 and 332–345
- [Fre 90] Freund, R.: On conjugate gradient type methods and polynomial preconditioners for a class of complex non-Hermitian matrices. *Numer. Math.* **57** (1990) 285–312
- [Fre 92] Freeman, T. L./ Phillips, C.: Parallel numerical algorithms. New York: Prentice Hall 1992
- [Fro 90] Frommer, A.: Lösung linearer Gleichungssysteme auf Parallelrechnern. Braunschweig: Vieweg 1990
- [Fuc 87] Fuchssteiner, B.: Solitons in interaction. *Progr. Theoret. Phys.* **78** (1987) 1022–1050
- [Fun 70] Funk, P.: Variationsrechnung und ihre Anwendung in Physik und Technik 2. Aufl. Berlin: Springer 1970
- [Gal 90a] Gallivan, K. A./ Heath, M. T./ Ng, E./ Ortega, J. M./ Peyton, B. W./ Plemmons, R. J./ Romine, C. H./ Sameh, A. H./ Voigt, R. G.: Parallel algorithms for matrix computations. Philadelphia: SIAM 1990
- [Gal 90b] Gallivan, K. A./ Plemmons, R. J./ Sameh, A. H.: Parallel algorithms for dense linear algebra computations. *SIAM Review* **32** (1990) 54–135
- [Gan 92] Gander, W.: Computermathematik, 2. Aufl. Basel: Birkhäuser 1992
- [Gau 70] Gautschi, W.: On the construction of Gaussian quadrature rules from modified moments. *Math. Comp.* **24** (1970) 245–260
- [Gea 71] Gear, C. W.: Numerical initial value problems in ordinary differential equations. Englewood Cliffs: Prentice Hall 1971
- [Gen 73] Gentleman, M.: Least squares computations by Givens transformations without square roots. *J. Inst. Math. Appl.* **12** (1973) 329–336
- [Geo 80] George, A./ Heath, M. T.: Solution of sparse linear least squares problems using Givens rotations. *Lin. Alg. Appl.* **34** (1980) 69–83
- [Gil 76] Gill, P. E./ Murray, W.: The orthogonal factorization of a large sparse matrix. In: Bunch und Rose [Bun 76], S. 201–212
- [Giv 54] Givens, J. W.: Numerical computation of the characteristic values of a real symmetric matrix. Rep. ORNL 1574, Oak Ridge Nat. Lab., Oak Ridge 1954
- [Giv 58] Givens, W.: Computation of plane unitary rotations transforming a general matrix to triangular form. *SIAM J. Appl. Math.* **6** (1958) 26–50
- [Gla 79] Gladwell, I./ Wait, R. (Hrsg.): A survey of numerical methods for partial differential equations. Oxford: Clarendon Press 1979
- [Gol 65] Golub, G. H./ Kahan, W.: Calculating the singular values and pseudo-inverse of a matrix. *SIAM J. Numer. Anal. Ser.B.* **2** (1965) 205–224
- [Gol 69] Golub, G. H./ Welsch, J. A.: Calculation of Gauss quadrature rules. *Math. Comp.* **23** (1969) 221–230

- [Gol 73] Golub, G. H./ Pereyra, V.: The differentiation of pseudo-inverses and nonlinear least square problems whose variabales separate. SIAM, J. Numer. Anal. **10** (1973) 413–432
- [Gol 80] Golub, G. H./ Plemmons, R. J.: Large-scale geodetic least-square adjustment by dissection and orthogonal decomposition. Lin. Alg. Appl. **34** (1980) 3–27
- [Gol 95] Golub, G. H./ Ortega, J. M.: Wissenschaftliches Rechnen und Differentialgleichungen. Eine Einführung in die Numerische Mathematik. Berlin: Heldermann 1995
- [Gol 96a] Golub, G. H./ Ortega, J. M.: Scientific Computing. Eine Einführung in das wissenschaftliche Rechnen und die parallele Numerik. Stuttgart: Teubner 1996
- [Gol 96b] Golub, G. H./ Van Loan, C. F.: Matrix computations. 3rd ed. Baltimore: John Hopkins University Press 1996
- [Goo 49] Goodwin, E. T.: The evaluation of integrals of the form  $\int_{\infty}^{\infty} f(x)c^{-x^2} dx$ . Proc. Cambr. Philos. Soc. (1949) 241–256
- [Goo 60] Good, I. J.: The interaction algorithm and practical Fourier series. J. Roy Statist. Soc. Ser. B, **20**, **22** (1958/1960) 361–372 and 372–375
- [Gou 79] Gourlay, A. R./ Watson, G. A.: Computational methods for matrix eigenproblems. London: John Wiley 1979
- [Gri 72] Grigorieff, R. D.: Numerik gewöhnlicher Differentialgleichungen, Band 1: Einschrittverfahren. Stuttgart: Teubner 1972
- [Gro 05] Großmann, C./ Roos, H.-G.: Numerische Behandlung partieller Differentialgleichungen. 3.Aufl. Wiesbaden: Teubner 2005
- [Hac 85] Hackbusch, W.: Multigrid methods and applications. Berlin: Springer 1985
- [Hac 93] Hackbusch, W.: Iterative Lösung großer schwach besetzter Gleichungssysteme, 2. Aufl. Stuttgart: Teubner 1993
- [Hac 96] Hackbusch, W.: Theorie und Numerik elliptischer Differentialgleichungen, 2. Aufl. Stuttgart: Teubner 1996
- [Hag 04] Hageman, L. A./ Young, D. M.: Applied iterative methods. Mineola: Dover Publications 2004
- [Hai 93] Hairer, E./ Nørsett, S./ Wanner, G.: Solving ordinary differential equations I: Nonstiff problems. 2nd ed. Berlin: Springer 1993
- [Hai 96] Hairer, E./ Wanner, G.: Solving odinary differential equations II: Stiff and differential-algebraic problems. 2nd ed. Berlin: Springer 1996
- [Ham 74] Hammarling, S.: A note on modifications to the Givens plane rotation. J. Inst. Math. Appl. **13** (1974) 215–218
- [Häm 91] Hämmerlin, G./ Hoffmann, K.-H.: Numerical mathematics. Berlin: Springer 1991
- [Häm 94] Hämmerlin, G./ Hoffmann, K.-H.: Numerische Mathematik, 4. Aufl. Berlin: Springer 1994
- [Heg 91] Hegland, M.: On the parallel solution of tridiagonal systems by wrap-around partitioning and incomplete LU-factorization. Num. Math. **59** (1991) 453–472
- [Hen 58] Henrici, P.: On the speed of convergence of cyclic and quasicyclic Jacobi methods for computing the eigenvalues of Hermitian matrices. SIAM J. App. Math. **6** (1958) 144–162
- [Hen 62] Henrici, P.: Discrete variable methods in ordinary differential equations. New York: Wiley 1962

- [Hen 66] Henrici, P.: Elements of numerical analysis. New York: Wiley 1966
- [Hen 72] Henrici, P.: Elemente der numerischen Analysis I, II. Heidelberg: Bibliographisches Institut 1972
- [Hen 82] Henrici, P.: Essentials of numerical analysis. New York: Wiley 1982
- [Hen 91] Henrici, P.: Applied and computational complex analysis, Vol.2. New York: John Wiley 1991
- [Hen 97] Henrici, P.: Applied and computational complex analysis, Vol.3. New York: John Wiley 1997
- [Hes 52] Hestenes, M. R./ Stiefel, E.: Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Standards* **49** (1952) 409–436
- [Heu 08] Heuser, H.: Lehrbuch der Analysis, Teil 2, 14. Aufl. Wiesbaden: Vieweg+Teubner 2008
- [Heu 09] Heuser, H.: Gewöhnliche Differentialgleichungen, 5. Aufl. Wiesbaden: Vieweg+Teubner 2009
- [Hig 02] Higham, N. J.: Accuracy and stability of numerical algorithms, 2nd ed. Philadelphia: SIAM 2002
- [Hil 88] Hill, D. R.: Experiments in computational matrix algebra. New York: Random House 1988
- [Hoc 65] Hockney, R.: A fast direct solution of Poisson's equation using Fourier analysis. *J. ACM* **12** (1965) 95–113
- [Hoc 88] Hockney, R./ Jesshope, C.: Parallel computers 2. Bristol: Adam Hilger 1988
- [Hod 92] Hodnett F. (Hrsg.): Proc. of the sixth european conference on mathematics in industry. Stuttgart: Teubner 1992
- [Hop 88] Hopkins, T./ Phillips, C.: Numerical methods in practice using the NAG library. Wokingham: Addison-Wesley 1988
- [Hos 92] Hoschek, J./ Lasser, D.: Grundlagen der geometrischen Datenverarbeitung, 2. Aufl. Stuttgart: Teubner 1992
- [Hou 58] Householder, A. S.: Unitary triangularization of a nonsymmetric matrix. *J. Assoc. Comp. Mach.* **5** (1958) 339–342
- [Hug 83] Hughes, T. J. R./ Levit, I./ Winget, J.: An element-by-element solution algorithm for problems of structural and solid mechanics. *Comp. Meth. Appl. Mech. Eng.* **36** (1983) 241–254
- [Hym 57] Hyman, M. A.: Eigenvalues and eigenvectors of general matrices. Texas: Twelfth National Meeting A.C.M. 1957
- [IMSL] IMSL: User's Manual. IMSL, Customer Relations, 14141 Southwest Freeway, Suite 3000, Sugar Land, Texas 77478–3498, USA.
- [Iri 70] Iri, M./ Moriguti, S./ Takasawa, Y.: On a certain quadrature formula (japanisch) [English transl.: *Comput. Appl. Math.* **17** (1987) 3–30]. RIMS Kokyuroku Kyoto Univ. **91** (1970) 82–118
- [Jac 46] Jacobi, C. G. J.: Über ein leichtes Verfahren, die in der Theorie der Säkularstörungen vorkommenden Gleichungen numerisch aufzulösen. *Crelle's Journal* **30** (1846) 51–94
- [Jac 72] Jacoby, S. L. S./ Kowalik, J. S./ Pizzo, J. T.: Iterative methods for nonlinear optimization problems. Englewood Cliffs: Prentice-Hall 1972
- [Jai 84] Jain, M. K.: Numerical solution of differential equations. 2nd ed. New York: John

- Wiley 1984
- [JE 82] Jordan-Engeln, G./ Reutter, F.: Numerische Mathematik für Ingenieure. 3. Aufl. Mannheim: Bibliographisches Institut 1982
- [Jen 77] Jennings, A./ Malik, G. M.: Partial elimination. *J. Inst. Math. Applies.* **20** (1977) 307–316
- [Joh 87] Johnson, S. L.: Solving tridiagonal systems on ensemble architectures. *SIAM J. Sci. Stat. Comp.* **8** (1987) 354–389
- [Kan 95] Kan, J. J. I. M. v./ Segal, A.: Numerik partieller Differentialgleichungen für Ingenieure. Stuttgart: Teubner 1995
- [Kau 79] Kaufman, L.: Application of dense Householder transformations to a sparse matrix. *ACM Trans. Math. Soft.* **5** (1979) 442–450
- [Kei 56] Keitel, G. H.: An extension of Milne's three-point method. *J. ACM* **3** (1956) 212–222
- [Kel 92] Keller, H. B.: Numerical methods for two-point boundary value problems. New York: Dover Publications 1992
- [Ker 78] Kershaw, D. S.: The incomplete Cholesky-conjugate gradient method for the iterative solution of systems of linear equations. *J. Comp. Physics* **26** (1978) 43–65
- [Kie 88] Kielbasinski, A./ Schwetlick, H.: Numerische lineare Algebra. Frankfurt: Verlag Harri Deutsch 1988
- [Kli 88] Klingbeil, E.: Variationsrechnung. 2. Aufl. Mannheim: Bibliographisches Institut 1988
- [Köc 90] Köckler, N.: Numerische Algorithmen in Softwaresystemen. Stuttgart: Teubner 1990
- [Köc 94] Köckler, N.: Numerical methods and scientific computing – using software libraries for problem solving. Oxford: Clarendon Press 1994
- [Kro 66] Kronrod, A. S.: Nodes and weights of quadrature formulas. New York: Consultants Bureau 1966
- [Kut 01] Kutta, W.: Beitrag zur näherungsweisen Integration totaler Differentialgleichungen. *Z. Math. Phys.* **46** (1901) 435–453
- [Lam 91] Lambert, J. D.: Numerical methods for ordinary differential systems. New York: John Wiley 1991
- [Lap 71] Lapidus, L./ Seinfeld, J. H.: Numerical solution of ordinary differential equations. New York: Academic Press 1971
- [Lap 99] Lapidus, L./ Pinder, G. F.: Numerical solution of partial differential equations in science and engineering. New York: Wiley 1999
- [Law 66] Lawson, J. D.: An order five Runge-Kutta process with extended region of stability. *SIAM J. Numer. Anal.* **3** (1966) 593–597
- [Law 95] Lawson, C. L./ Hanson, R. J.: Solving least squares problems, Unabridged corr. republ. Philadelphia: SIAM 1995
- [Leh 96] Lehoucq, R. B./ Sorensen, D. C.: Deflation Techniques for an Implicitly Re-Started Arnoldi Iteration. *SIAM J. Matrix Analysis and Applications* **17** (1996) 789–821
- [Leh 98] Lehoucq, R. B./ Sorensen, D. C./ Yang, C.: ARPACK Users' Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods. Philadelphia: SIAM 1998
- [Lei 97] Leighton, F. T.: Einführung in parallele Algorithmen und Architekturen. Bonn:

- Thomson 1997
- [Lin 01] Linz, P.: Theoretical numerical analysis. Mineola: Dover Publications 2001
- [Loc 93] Locher, F.: Numerische Mathematik für Informatiker. Berlin: Springer 1993
- [Lud 71] Ludwig, R.: Methoden der Fehler- und Ausgleichsrechnung, 2. Aufl. Berlin: Deutscher Verlag der Wissenschaften 1971
- [Lyn 83] Lyness, J. N.: When not to use an automatic quadrature routine. SIAM Rev. **25** (1983) 63–87
- [Mae 54] Maehly, H. J.: Zur iterativen Auflösung algebraischer Gleichungen. ZaMP **5** (1954) 260–263
- [Mae 85] Maess, G.: Vorlesungen über numerische Mathematik Band I: Lineare Algebra. Basel: Birkhäuser 1985
- [Mag 85] Magid, A.: Applied matrix models. New York: Wiley 1985
- [Mar 63] Marquardt, D. W.: An algorithm for least-squares estimation of nonlinear parameters. J. Soc. Indust. Appl. Math. **11** (1963) 431–441
- [Mar 86] Marsal, D.: Die numerische Lösung partieller Differentialgleichungen in Wissenschaft und Technik. Mannheim: Bibliographisches Institut 1986
- [McC 88] McCormick, S. F.: Multigrid methods: theory, applications, and supercomputing. New York: Dekker 1988
- [Mei 77] Meijerink, J. A./ van der Vorst, H. A.: An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-Matrix. Math. Comp. **31** (1977) 148–162
- [Mit 80] Mitchell, A. R./ Griffiths, D. F.: The finite difference method in partial differential equations. Chichester: Wiley 1980
- [Mit 85] Mitchell, A. R./ Wait, R.: The finite element method in partial differential Equations. London: Wiley 1985
- [Mol] Moler, C. B.: MATLAB – User’s Guide. The Math Works Inc., 3 Apple Drive, Natick, Mass. 01760-2098, USA
- [Mol 73] Moler, C. B./ Stewart, G. W.: An algorithm for generalized matrix eigenvalue problems. SIAM J. Numer. Anal. **10** (1973) 241–256
- [Mor 78] Mori, M.: An IMT-type double exponential formula for numerical integration. Publ. RIMS Kyoto Univ. **14** (1978) 713–729
- [Mor 94] Morton, K. W./ Mayers, D. F.: Numerical solution of partial differential equations. Cambridge: Cambridge University Press 1994
- [NAGa] NAG: C Library manual. The Numerical Algorithm Group Ltd., Wilkinson House, Jordan Hill Road, Oxford OX2 8DR, U.K.
- [NAGb] NAG: Fortran library manual. The Numerical Algorithm Group Ltd., Wilkinson House, Jordan Hill Road, Oxford OX2 8DR, U.K.
- [NO 85] Nour-Omid, B./ Parlett, B. N.: Element preconditioning using splitting techniques. SIAM J. Sci. Stat. Comp. **6** (1985) 761–771
- [Ort 88] Ortega, J.: Introduction to parallel and vector solution of linear systems. New York: Plenum Press 1988
- [Ort 00] Ortega, J. M./ Rheinboldt, W. C.: Iterative solution of nonlinear equations in several variables. Philadelphia: SIAM 2000
- [Par 98] Parlett, B. N.: The symmetric eigenvalue problem. Philadelphia: SIAM 1998
- [Pat 69] Patterson, T. N. L.: The optimum addition of points to quadrature formulae.

- Math. Comp. [errata Math. Comp.] **22** [**23**] (1968 [1969]) 847–856 [892]
- [Pea 55] Peaceman, D. W./ Rachford, H. H.: The numerical solution of parabolic and elliptic differential equations. J. Soc. Industr. Appl. Math. **3** (1955) 28–41
- [Pic 86] Pickering, M.: An introduction to fast fourier transform methods for PDE, with applications. New York: Wiley 1986
- [Pie 83] Piessens, R./ Doncker-Kapenga von (ed.), E./ Ueberhuber, C. W./ Kahaner, D. K.: QUADPACK. A subroutine package for automatic integration. Berlin: Springer 1983
- [Rab 88] Rabinowitz (ed.), P.: Numerical methods for nonlinear algebraic equations. New York: Gordon and Breach 1988
- [Rao 71] Rao, C. R./ Mitra, S. K.: Generalized inverse of matrices and its applications. New York: Wiley 1971
- [Rat 82] Rath, W.: Fast Givens rotations for orthogonal similarity transformations. Numer. Math. **40** (1982) 47–56
- [Reu 88] Reuter, R.: Solving tridiagonal systems of linear equations on the IBM 3090 VF. Parallel Comp. **8** (1988) 371–376
- [Rob 91] Robert, Y.: The impact of vector and parallel architectures on the Gaussian elimination algorithm. New York: Halsted Press 1991
- [Roo 99] Roos, H.-G./ Schwetlick, H.: Numerische Mathematik. Stuttgart: Teubner 1999
- [Rüd 93] Rüde, U.: Mathematical and computational techniques for multilevel adaptive methods. Philadelphia: SIAM 1993
- [Run 95] Runge, C.: Über die numerische Auflösung von Differentialgleichungen. Math. Ann. **46** (1895) 167–178
- [Run 03] Runge, C.: Über die Zerlegung empirisch gegebener periodischer Funktionen in Sinuswellen. Z. Math. Phys. **48** (1903) 443–456
- [Run 05] Runge, C.: Über die Zerlegung einer empirischen Funktion in Sinuswellen. Z. Math. Phys. **52** (1905) 117–123
- [Run 24] Runge, C./ König, H.: Vorlesungen über numerisches Rechnen. Berlin: Springer 1924
- [Rut 52] Rutishauser, H.: Über die Instabilität von Methoden zur Integration gewöhnlicher Differentialgleichungen. ZaMP **3** (1952) 65–74
- [Rut 66] Rutishauser, H.: The Jacobi method for real symmetric matrices. Numer. Math. **9** (1966) 1–10
- [Rut 69] Rutishauser, H.: Computational aspects of F. L. Bauer's simultaneous iteration method. Num. Math. **13** (1969) 4–13
- [Saa 81] Saad, Y.: Krylov subspace methods for solving large unsymmetric linear systems. Math. Comp. **37** (1981) 105–126
- [Saa 86] Saad, Y./ Schultz, M. H.: GMRES: A generalized minimal residual method for solving nonsymmetric linear systems. SIAM J. Sci. Statist. Comp. **7** (1986) 856–869
- [Sag 64] Sag, W. T./ Szekeres, G.: Numerical evaluation of high-dimensional integrals. Math. Comp. **18** (1964) 245–253
- [Sau 68] Sauer, R./ Szabó, I.: Mathematische Hilfsmittel des Ingenieurs. Band 3. Berlin: Springer 1968
- [Sch 09] Schur, I.: Über die charakteristischen Wurzeln einer linearen Substitution mit einer

- Anwendung auf die Theorie der Integralgleichungen. *Math. Annalen* **66** (1909) 488–510
- [Sch 61] Schönhage, A.: Zur Konvergenz des Jacobi-Vefahrens. *Numer. Math.* **3** (1961) 374–380
- [Sch 64] Schönhage, A.: Zur quadratischen Konvergenz des Jacobi-Vefahrens. *Numer. Math.* **6** (1964) 410–412
- [Sch 69] Schwartz, C.: Numerical integration of analytic functions. *J. Comp. Phys.* **4** (1969) 19–29
- [Sch 72] Schwarz, H. R./ Rutishauser, H./ Stiefel, E.: Numerik symmetrischer Matrizen. 2. Aufl. Stuttgart: Teubner 1972
- [Sch 76] Schmeisser, G./ Schirmeier, H.: Praktische Mathematik. Berlin: de Gruyter 1976
- [Sch 91a] Schwarz, H. R.: FORTRAN-Programme zur Methode der finiten Elemente. 3.Aufl. Stuttgart: Teubner 1991
- [Sch 91b] Schwarz, H. R.: Methode der finiten Elemente. 3.Aufl. Stuttgart: Teubner 1991
- [Sch 97] Schwarz, H. R.: Numerische Mathematik. 4. Aufl. Stuttgart: Teubner 1997
- [Sch 05] Schaback, R./ Wendland, H.: Numerische Mathematik. 5. Aufl. Berlin: Springer 2005
- [Sch 08] Schwarz, H. R./ Köckler, N.: Numerische Mathematik. 7. Aufl. Wiesbaden: Vieweg+Teubner 2008
- [Sew 05] Sewell, G.: The numerical solution of ordinary and partial differential equations, 2nd ed. New York: Wiley 2005
- [Sha 84] Shampine, L. F./ Gordon, M. K.: Computer-Lösungen gewöhnlicher Differentialgleichungen. Das Anfangswertproblem. Braunschweig: Friedr. Vieweg 1984
- [Sha 94] Shampine, L. F.: Numerical solution of ordinary differential equations. New York: Chapman & Hall 1994
- [Sha 00] Shampine, L. F./ Kierzenka, J./ Reichelt, M. W.: Solving boundary value problems for ordinary differential equations in MATLAB with `bvp4c`. <ftp://ftp.mathworks.com/pub/doc/papers/bvp/> (2000)
- [Ske 01] Skeel, R. D./ Keiper, J. B.: Elementary numerical computing with mathematica. Champaign: Stipes 2001
- [Smi 85] Smith, G. D.: Numerical solution of partial differential equations: Finite difference method. 3rd ed. Oxford: Clarendon Press 1985
- [Smi 88] Smith, B. T./ Boyle, J. M./ Dongarra, J. J./ Garbow, B. S./ Ikebe, Y./ Klema, V. C./ Moler, C. B.: Matrix eigensystem routines – EISPACK Guide. Berlin: Springer 1988
- [Smi 90] Smirnow, W. I.: Lehrgang der höheren Mathematik, Teil II. Berlin: VEB Deutscher Verlag der Wissenschaften 1990
- [Sta 94] Stammbach, U.: Lineare Algebra. 4. Aufl. Stuttgart: Teubner 1994.  
Im Internet: <http://www.math.ethz.ch/~stammb/linalg.shtml>
- [Ste 73a] Stenger, F.: Integration formulas based on the trapezoidal formula. *J. Inst. Math. Appl.* **12** (1973) 103–114
- [Ste 73b] Stetter, H. J.: Analysis of discretization methods for ordinary differential equation. Berlin: Springer 1973
- [Ste 76] Stewart, G. W.: The economical storage of plane rotations. *Numer. Math.* **25** (1976) 137–138

- [Ste 83] Stewart, G. W.: *Introduction to matrix computations*. New York: Academic Press 1983
- [Sti 76] Stiefel, E.: *Einführung in die numerische Mathematik* 5. Aufl. Stuttgart: Teubner 1976
- [Sto 73] Stone, H.: An efficient parallel algorithm for the solution of a triangular system of equations. *J. ACM* **20** (1973) 27–38
- [Sto 75] Stone, H.: Parallel tridiagonal equation solvers. *ACM Trans. Math. Software* **1** (1975) 280–307
- [Sto 02] Stoer, J. and Bulirsch, R.: *Introduction to numerical analysis*, 3rd ed. New York: Springer 2002
- [Sto 05] Stoer, J. and Bulirsch, R.: *Numerische Mathematik 2*, 5. Aufl. Berlin: Springer 2005
- [Sto 07] Stoer, J.: *Numerische Mathematik 1*, 10. Aufl. Berlin: Springer 2007
- [Str 66] Stroud, A. H./ Secrest, D.: *Gaussian quadrature formulas*. Englewood Cliffs: Prentice-Hall 1966
- [Str 74] Stroud, A. H.: *Numerical quadrature and solution of ordinary differential equations*. New York: Springer 1974
- [Str 95] Strehmel, K./ Weiner, R.: *Numerik gewöhnlicher Differentialgleichungen*. Stuttgart: Teubner 1995
- [Stu 82] Stummel, F./ Hainer, K.: *Praktische Mathematik*. 2. Aufl. Stuttgart: Teubner 1982
- [Swi 78] Swift, A./ Lindfield, G. R.: Comparison of a continuation method with Brent's method for the numerical solution of a single nonlinear equation. *Comp. J.* **21** (1978) 359–362
- [Sze 62] Szekeres, G.: Fractional iteration of exponentially growing functions. *J. Australian Math. Soc.* **2** (1961/62) 301–320
- [Tak 73] Takahasi, H./ Mori, M.: Quadrature formulas obtained by variable transformation. *Numer. Math.* **21** (1973) 206–219
- [Tak 74] Takahasi, H./ Mori, M.: Double exponential formulas for numerical integration. *Publ. RIMS Kyoto Univ.* **9** (1974) 721–741
- [Tör 88] Törnig, W./ Spellucci, P.: *Numerische Mathematik für Ingenieure und Physiker*, Band 1: *Numerische Methoden der Algebra*. 2. Aufl. Berlin: Springer 1988
- [Tve 02] Tveito, A./ Winther, R.: *Einführung in partielle Differentialgleichungen. Ein numerischer Zugang*. Berlin: Springer 2002
- [Übe 95] Überhuber, C. W.: *Computer-Numerik 1 und 2*. Heidelberg: Springer 1995
- [Var 00] Varga, R. S.: *Matrix iterative analysis*, 2nd ed. Berlin: Springer 2000
- [vdV 87a] van der Vorst, H. A.: Analysis of a parallel solution method for tridiagonal linear systems. *Parallel Comp.* **5** (1987) 303–311
- [vdV 87b] van der Vorst, H. A.: Large tridiagonal and block tridiagonal linear systems on vector and parallel computers. *Parallel Comp.* **5** (1987) 45–54
- [vdV 90] van der Vorst, H. A./ van Dooren, P. (Hrsg.): *Parallel algorithms for numerical linear algebra*. Amsterdam: North Holland 1990
- [Wal 88] Walker, H. F.: Implementation of the GMRES method using Householder transformations. *SIAM J. Sci. Statist. Comp.* **9** (1988) 152–163
- [Wal 00] Walter, W.: *Gewöhnliche Differentialgleichungen*, 7. Aufl. Berlin: Springer 2000

- [Wan 81] Wang, H.: A parallel method for tridiagonal equations. *ACM Trans. Math. Software* **7** (1981) 170–183
- [Wes 04] Wesseling, P.: An Introduction to Multigrid Methods. Flourtown: R. T. Edwards 2004
- [Wil 68] Wilkinson, J. H.: Global convergence of tridiagonal QR-algorithm with origin shifts. *Lin. Alg. and Its Appl.* **1** (1968) 409–420
- [Wil 69] Wilkinson, J. H.: Rundungsfehler. Berlin-Heidelberg-New York: Springer 1969
- [Wil 86] Wilkinson, J. H./ Reinsch, C. (Hrsg.): Linear Algebra. Handbook for Automatic Computation Vol. II. Berlin: Springer 1986
- [Wil 88] Wilkinson, J. H.: The algebraic eigenvalue problem. Oxford: University Press 1988
- [Wil 94] Wilkinson, J. H.: Rounding errors in algebraic processes. New York: Dover Publications 1994
- [Win 78] Winograd, S.: On computing the discrete Fourier transform. *Math. Comp.* **32** (1978) 175–199
- [You 71] Young, D. M.: Iterative solution of large linear systems. New York: Academic Press 1971
- [You 90] Young, D. M./ Gregory, R. T.: A survey of numerical mathematics Vol. 1 + 2. New York: Chelsea Publishing Co. 1990
- [Yse 86] Yserentant, H.: Hierachical basis give conjugate gradient type methods a multigrid speed of convergence. *Applied Math. Comp.* **19** (1986) 347–358
- [Yse 89] Yserentant, H.: Preconditioning indefinite discretization matrices. *Numer. Math.* **54** (1989) 719–734
- [Yse 90] Yserentant, H.: Two preconditioners based on the multi-level splitting of finite element spaces. *Numer. Math.* **58** (1990) 163–184
- [Zie 05] Zienkiewicz, O. C./ Taylor, R. L.: The finite element method, 6th ed. Oxford: Butterworth-Heinemann 2005
- [Zon 79] Zonneveld, J. A.: Automatic numerical integration. math. centre tracts, No. 8. Amsterdam: Mathematisch Centrum 1979
- [Zur 65] Zürmühl, R.: Praktische Mathematik für Ingenieure und Physiker 5. Aufl. Berlin: Springer 1965
- [Zur 86] Zürmühl, R./ Falk, S.: Matrizen und ihre Anwendungen. Teil 2: Numerische Methoden. 5. Aufl. Berlin: Springer 1986
- [Zur 97] Zürmühl, R./ Falk, S.: Matrizen und ihre Anwendungen. Teil 1: Grundlagen. 7. Aufl. Berlin: Springer 1997

## Sachverzeichnis

- 0/0-Situation 25  
1. und 2. erzeugendes Polynom *siehe* charakteristisches Polynom
- a posteriori Fehler 186, 188, 495  
a priori Fehler 186f., 191, 495  
 $A(\alpha)$ -stabil 384  
A-B-M-Verfahren 366  
A-orthogonal **532**  
Abbildung, kontrahierende 186  
Abbruchfehler 15  
Ableitungen in Randbedingungen 421  
Ableitungswert, Interpolation 99  
Abschneidefehler *siehe* Diskretisierungsfehler  
absolute Stabilität 350, 378f., 381, 452f., 455f., 461, 463  
absoluter Fehler **16**, 20, 53f.  
Abstiegs-, Methode des steilsten 206, **301**, 532  
Abstiegsrichtung 302  
Adams-Verfahren 363, **363**, 365f., 375, 381, 391  
adaptive Integration **332**, 333f., 339  
Adaptivität 482  
adjungiert 515, 521  
ähnliche Matrix 52  
Ähnlichkeitstransformation **219**, 221, 254, 257, 542  
orthogonale 232  
Akustik 483  
algebraische Substitution 320  
ALGOL 87  
Algorithmus 13, 15, 19  
Aliasing-Effekt 510  
allgemeines Eigenwertproblem 218, **261**  
alternierende Divergenz 193  
alternierenden Richtungen, Methode der 464  
AMS-Guide 14  
Anfangsrandwertaufgabe, parabolische 448
- Anfangszustand 342  
Anlaufrechnung *siehe* Startrechnung  
ANSYS 483  
Approximation  
    bikubische Spline- 125  
    diskrete Gauß- 118, **142**  
    Gauß- 124  
    kontinuierliche Gauß- **144**  
    sukzessive 184  
    trigonometrische **145**  
Approximationssatz, Weierstraßscher 127  
Äquivalenzoperation 31  
Archimedesspirale 126  
Architektur 67  
Arnold-Iteration 270  
Artenverhalten 342  
Artillerie 409  
asymptotisch optimal 489  
asymptotische Konvergenz 496  
asymptotische Konvergenzrate 509  
asymptotische Stabilität 352, 372  
Aufstartzeit 71  
Aufwand *siehe* Rechenaufwand  
Ausgleichsprinzip 274  
Ausgleichung  
    lineare 143  
    nichtlineare 296  
Auslöschung 18, 20, 39, 41, 223, 283, 288, 411  
Automobilbau 483  
autonome Differenzialgleichung 388  
axiale Druckkraft 399
- B-Splines **112**, 406  
    Ableitung 115  
    bikubische **123**  
    Integral 115  
    kubische **114**  
backward analysis **24**  
backward differentiation formulae 374

- Bahnlinien 389  
Bairstow-Verfahren 211  
    Zusammenfassung 213  
Balken 399, 517  
Banach-Raum 185, **185**  
Banachscher Fixpunktsatz **185f.**, 199  
Bandbreite 62, **62**, 63, 86, 442, 445, 477, 488  
Bandmatrix **62**, 85, 406, 442, 445, 477, 487  
    bei Interpolation 124  
    Speicherung 63  
Bankkonflikt 71f.  
Basis 16  
    hierarchische 546  
    orthonormierte 51  
Basisfunktion 470, 473  
BDF-Verfahren 374, 376, 383  
Belastung, transversale 399  
Benutzerführung 13  
Beobachtungsfehler 274  
Bernoulli-Zahl 313f.  
Bernstein-Polynom **127**, 134  
Bessel-Funktion 167, 317  
Bevölkerungsentwicklungsmodell 342  
Bézier-Flächen **137**  
Bézier-Kurven **131**  
Bézier-Polygon **129**  
Bézier-Polynom **127**  
Bézier-Punkt **129**  
Bibliothek *siehe* NAG, IMSL  
bidiagonal 64, 67, 76  
Biegesteifigkeit 399  
biharmonische Differenzialgleichung 447  
bikubische Splinefunktion **123**, 125  
bikubische Tensorsplines **123**  
bilineare Tensorsplines **120**  
Binärdarstellung 159  
Binomialkoeffizient 103  
binomische Formel 127  
Biologie 274, 342  
Biomedizin 483  
Bisektion **190**, 191, 195ff.  
Bitumkehr 161  
Black-box-Verfahren 183, 193, 205, 304, 335  
BLACS 70, 87  
BLAS 87  
Blockgröße 67, 70  
Blockmatrix 68  
Bogenlänge 125  
Brent-Verfahren **196**, 198  
Cache 67  
CAD-FEM 483  
Casteljau-Algorithmus **130**, 131, 139  
Cauchy-Folge 185, 187  
Cauchy-Randbedingung 429, 434, 438ff.,  
    444, 449, 460, 469  
Cauchy-Schwarzsche-Ungleichung 143  
Ceschino-Kuntzmann 360  
CG-Verfahren **530**, 534, 536, 538, 540, 553f.,  
    557  
charakteristische Funktion *siehe* charakteris-  
    tisches Polynom  
charakteristische Gleichung 371  
charakteristisches Polynom **219**, 265, **369**,  
    371f.  
Chemie 342, 387, 483  
Cholesky 59  
Cholesky-Verfahren 420, 433, 477, 516  
Cholesky-Zerlegung **59**, 62f., 261, 275, 277f.,  
    283  
    partielle 545  
Clenshaw-Algorithmus 169  
Cooley 159  
Cramersche Regel 201  
Crank-Nicolson 455f., 463  
    Rechenaufwand 456  
CSIRO 483  
  
Dachfunktion 148, 153f., 170  
Dahlquist-stabil 372  
Datenfehler **15**, 19, 21, 266  
Datenorganisation 13  
Datentabelle 91, 142  
Datenverteilung 68  
Datenwolke 91  
Deflation 209f.  
Determinante 36f.  
    Berechnung **30**  
    Vandermondesche 371  
diagonal dominant 64, 78, 80, 433, 442, 456,  
    459, 463f., 506  
    schwach **39**, 496, 498, 500, 502  
    strikt **39**, 502  
diagonalähnlich **219**, 265  
Diagonalstrategie 38f., 58, 64, 72f., 442, 453,  
    456  
Differenz, dividierte **95**, **418**  
Differenzen, dividierte 508, 525  
Differenzenapproximation 419

- Differenzengleichung 371, 381, 430, 436ff., 441, 444, 450, 506, 545  
 Differenzenquotient 103, 411, 430, 450  
   zentraler 103  
 Differenzenschema 98  
 Differenzenstern 268, 430, 432, 498  
 Differenzenverfahren 395, **418**, 420, **427**, 429, 487, 498, 505, 508, 514, 530, 544, 555  
 Differenzialgleichung 115  
   autonome 388  
   elliptische 428  
   explizite 343  
   hyperbolische 428  
   implizite 343  
   parabolische 428  
   partielle 124, 268  
   steife 385  
 Differenzialgleichungssystem 218  
   erster Ordnung 343  
   m-ter Ordnung 343  
 Differenziation, numerische 91, **101**, 374, 409  
 differenzielle Fehleranalyse 19, **21**, 23f.  
 Diffusion 427, 448, 459  
 Dirichlet-Randbedingung 429, 449, 460, 470, 477  
 Dirichletsche Randwertaufgabe 429  
 diskrete Eigenfunktion 270  
 diskrete Fehlerquadratmethode 405  
 diskrete Fourier-Transformation 154ff.  
 diskrete Gauß-Approximation 118, **142**  
 Diskretisierung partieller Differenzialgleichungen 268, **487**, 498  
 Diskretisierungsfehler **15**, 352, **359**, 364f., 367, 369f., 388, 411, 420  
   globaler **351**, 452, 455  
   lokaler **351**, 356, 369f., 444f., 452, 455  
   Schätzung 361, 366  
 Divergenz, alternierende 193  
 Divide-and-Conquer 76  
 dividierte Differenz **95**, **418**, 487  
 dividierte Differenzen 508, 525  
 Divisionsalgorithmus 211  
 Dopingtest 342  
 doppelte Nullstellen 189, 193  
 Drei-Punkte-Stern 432  
 Dreiecke, Integration über 337  
 Dreieckelement 469, 474  
 Dreiecknetz 429  
 Dreieckszerlegung **30**, 46, 55, 64  
 Dreigittermethode 522  
 Druckkraft, axiale 399  
 Druckverlust 84  
 Dualsystem 16, 41  
 dünn besetzt *siehe* schwach besetzt  
 Durchbiegung 399  
 ebenes Fachwerk 85  
 Eigenfrequenz 268  
 Eigenfunktion einer Differenzialgleichung 270, 396  
 Eigensystem einer Differenzialgleichung 397  
 Eigenvektor 51  
   QR-Algorithmus 260  
   Rechenaufwand 260  
 Eigenwert 51f.  
 Eigenwert einer Differenzialgleichung 396  
 Eigenwertproblem **395**  
   allgemeines 218, **261**  
   Konditionszahl **266**  
   spezielles 218, 261  
   symmetrisches 220, 261  
 Einbettungsmethode 416  
 Eindeutigkeit 184  
 Einfach-Schießverfahren **408**, 412, 416  
 Eingabefehler 19  
 eingebettete Gauß-Regel **331**  
 eingebettete Verfahren 360  
 eingebettetes Runge-Kutta-Verfahren **359**  
 Einheitswurzel 155f.  
 Einschließung 191, 197  
 Einschrittverfahren 346, **351**, 368, 378f.  
   äquidistantes 352  
   implizites 380  
   Konstruktion 353  
   Startrechnung 375  
 einstufiges Iterationsverfahren 184, 493  
 Einzelschrittverfahren **203**, 359, 368, 489, **490**, 492, 502, 507f.  
   Newtonisches 204  
   nichtlineares 204  
 EISPACK 87  
 Elastizität 268, 399  
 Elektronik 483  
 Elektrostatik 428  
 elementare Rechenoperation **15**, 21  
 Elementvektor 475  
 Elimination **30**, 32, 36, 38f.

- Ellipse 135
- Ellipsoid 317
- elliptische Differenzialgleichung 428, 468
- elliptische Randwertaufgabe 467, 487, 505, 530, 545f., 555
- Differenzenverfahren **427**
- Empfindlichkeitsmaß 413
- Energieerhaltungssatz 56
- Energiemethode 466ff.
- Energiemiminierungsprinzip 107
- Energienorm **531**, 538ff.
- England 360
- Epidemie 342
- Erdvermessung 284
- erreichbare Konsistenzordnung 357
- erzeugendes Polynom *siehe* charakteristisches Polynom
- euklidische Norm **48**, 49, 51f.
- Euler 108
- Euler-Collatz 357
- Euler-MacLaurin 313
- Euler-Verfahren 347f., 354, 452
  - explizites **345**, 347, 349, 354
  - Konvergenz 349
  - implizites **346**, 350
  - Konvergenz 350
- Eulersche Differenzialgleichung 468
- Evolutionsproblem 427
- Existenz 30, 184
- exp-Substitution 322
- Experimentalphysik 274
- explizite Differenzialgleichung 343
- explizite Methode von Richardson 450f., 461f.
  - lokaler Diskretisierungsfehler 451
- explizites Adams-Verfahren 363
- explizites Euler-Verfahren 345, 347, 349, 369
- explizites Mehrschrittverfahren 368
- explizites Runge-Kutta-Verfahren **354**, 356, 379
- Exponent 16
- Extrapolation 105, 420
  - auf  $h^2 = 0$  313
  - Richardson- 313
- Extrapolationsverfahren 363
- Extremalaufgabe 467
- Extremalstelle 94
- Fachwerk, ebenes 85
- Faktorisierung *siehe* Zerlegung
- FASTFLO 483
- Fehlberg 360
- Fehler
  - a posteriori 186, 188, 495
  - a priori 186f., 191, 495
  - absoluter 16, 20, 53f.
  - relativer 16, 20, 53ff.
- Fehlerabschätzung **52**, 54, 352
  - Romberg-Integration 314
  - Simpson-Regel 310
  - Trapezregel 310
- Fehleranalyse 14
  - differenzielle 19, 21, 23f.
  - Rückwärts- **24**, 55
- Fehlerarten **15**
- Fehlerentwicklung, Trapezregel 314
- Fehlerfortpflanzung 15, 345
- Fehlerfunktion 404, 406f.
- Fehlergleichungen 274f., 280, 282, 290, 292, 295, 303, 405, 548, 550, 553
- Rechenaufwand 276
- Fehlerkonstante **186**, 194, 364f., 367, 370, 375
- Fehlerkontrolle 335
- Fehlerkurve 94
- Fehlerordnung 345ff.
  - Fünf-Punkte-Formel 445
- Fehlerquadratmethode **405**
  - diskrete 405
  - kontinuierliche 405
- Fehlerschätzung 23, 359ff.
  - bei adaptiver Integration 335
  - zur Schrittweitensteuerung 360
- Fehlertheorie **15**
- FEMLAB 480, 482
- Fermatsches Prinzip **467**
- FFT (fast Fourier transform) *siehe* schnelle Fourier-Transformation
- fill-in 290, 487, 545
- Finde-Routine 197
- finite Elemente 307, 337, 406, **466**, 487, 530, 544
- FITPACK 179
- Fixpunkt **184**, 187, 192, 199
- Fixpunktgleichung 489, 494
- Fixpunktiteration 184, 192, 198f., 493ff., 509
  - lineare 493
  - stationäre 493

- Fixpunktsatz 494  
 Banachscher **185f.**
- Fläche, glatte 107
- Flächenträgheitsmoment 399
- Flachwasserwelle 124, 336
- flops 70, 75
- Flugzeugbau 107
- FMG-Zyklus 523
- Formfunktion 473f., 476
- fortgepflanzte Rundungsfehler 15, 21
- Fortpflanzungsproblem 427
- Fourier-Analyse 524
- Fourier-Integral 319
- Fourier-Koeffizient **147**, 154, 166, 316ff.
- Fourier-Polynom **147**, 152
- Fourier-Reihe **145**, **147**, 162, 316, 318, 510
- Fourier-Term, glatt 510
- Fourier-Term, hochfrequent 510
- Fourier-Term, niederfrequent 510
- Fourier-Term, oszillatorisch 510
- Fourier-Transformation
- diskrete 154ff.
  - komplexe 155, 157, 161
  - schnelle **155**, 159, 161f.
- Francis, QR-Algorithmus von 248
- Frequenz-Analyse 524
- Frobenius-Norm **48**, 49, 303
- Full Multigrid 523
- full-weighting 515
- Fundamentalmatrix 400
- Fundamentalsystem 396, 400f.
- Fünf-Punkte-Formel 463
- Fehlerordnung 445
- Fünf-Punkte-Stern 487
- Funktion, trigonometrische 145
- Funktionalmatrix 19, 21, 199, 201ff., 386, 411
- Funktionen, Schnittpunkt zweier 184
- Funktionensystem 140, 142
- orthogonales 143, 145f., 151, 164, 174
- Funktionsauswertung 198, 206
- FW-Operator 515, 520
- Galerkin-Methode **406**
- GAMS-Software-Index 14, 391
- Gander 333
- garantierte Konvergenz 196
- Gauß-Algorithmus **30**, 33, 36ff., 43, **44**, 58, 65ff., 70, 72, 76, 201f., 260, 453, 516
- Rechenaufwand 38
- Gauß-Approximation 124, **140**, 405
- diskrete 118, **142**
  - kontinuierliche **144**, 165
- Gauß-Form
- Runge-Kutta 358
- Gauß-Integration 177, 307, **323**, 332, 405
- eingebettete **331**, 332
  - mehrdimensionale 337
- Gauß-Jordan-Verfahren 71f.
- Gauß-Kronrod-Integration 332, 335, 338
- Gauß-Newton-Methode **297**, 298, 300ff.
- Gauß-Patterson-Integration 335
- Gauß-Seidel 359, 489f., 511
- Gebiet absoluter Stabilität 379, 381, 383
- Gebietsdiskretisierung 470
- GECR-Methode 82
- gedämpfte Jacobi-Iteration **491**, 509, 511, 525
- Genauigkeit 16, 270
- doppelte 28
  - einfache 28
- Genauigkeitsgrad
- Quadraturformel 324f., 331
- Genauigkeitsgrad einer Quadraturformel 309
- Genauigkeitssteuerung 333f.
- Romberg-Integration 315
- Genauigkeitsverlust 25
- Gentleman 239
- geometrische Folge 191
- geometrische Reihe 150
- gerichtete Kante 497
- gerichteter Graph 497
- Gershgorin 111
- Gershgorin, Satz von 264
- Gesamtfehler
- bei Anfangswertproblemen 352
- Gesamtnorm **48**, 49
- Gesamtschrittverfahren 489, **490**, 492, 507
- Gesamtsteifigkeitsmatrix 477
- gestaffelte QR-Transformation 250
- gewichtetes  $L_2$ -Skalarprodukt 176
- Gewichtsfunktion 330
- Gitter 68, 269
- Rechteck- 120, 123, 125
- Gittererzeugung 482
- Gitterpunkte 418, 429, 462, 505
- Nummerierung 432f.
- Gitterweite 268, 429, 487
- Givens, Methode von 237

- Givens-Rotation 233, 254  
Givens-Transformation 258, 279f., 283, 303f., 550  
    Rechenaufwand 281  
    schnelle 256  
glatte Fläche 107  
glatte Kurve 91  
glatter Fourier-Term 510  
Glättungsfaktor 511, 524f.  
Gleichgewichtsproblem 427  
Gleichgewichtszustand 389f.  
Gleichungssystem  
    lineares 118, 124  
    nichtlineares **199**, 205, 346  
    Normal- 141, 205  
    quadratisches 124  
    schlecht konditioniertes 206  
    tridiagonales **64**  
Gleitpunktarithmetik 55, 61  
Gleitpunktrechnung 54  
Gleitpunktzahl 28  
Gleitpunktzahl, normalisierte 16  
globaler Diskretisierungsfehler *siehe* Diskretisierungsfehler  
globales Minimum 205  
GMRES( $m$ )-Algorithmus **553**, 555  
    Konvergenz 554, 556  
    Rechenaufwand 556  
GMRES-Algorithmus 548, **552**, 553  
    Rechenaufwand 553  
Good 159  
Gradient, konjugierter 530, 532, 534, 548  
Graph, gerichteter 497  
Graph, zusammenhängender 497  
Graphik-Software 126  
Grenzennorm 50  
Grenzwert 186, 205  
  
halbimplizites Runge-Kutta-Verfahren **358**  
Hamiltonsches Prinzip **467**  
Hardware **14**, 482  
Hauptachsensatz 220, 222  
Helmholtz-Gleichung 428  
Hermite-Interpolation 94, **98**, 108  
Hermite-Polynom 331  
Hessenberg-Form **233**  
    Rechenaufwand 234  
Hessenberg-Matrix 232, 244, 254, 552  
Hessesche Matrix 275, 530  
  
Hestenes 532  
Heun-Verfahren 354, 357  
hierarchische Basis 546  
Hilbert-Matrix 145, 266  
Hilbert-Raum 140, 142, 144  
hochfrequent 510, 515  
Höchstleistungsrechner 67  
Horner-Schema 97, 208  
    doppelzeiliges 214  
Householder 303  
Householder, Rücktransformation  
    Rechenaufwand 290  
Householder-Matrix **286**, 288f.  
Householder-Transformation 256, **286**, 290, 304  
Hülle 284  
hüllenorientierte Rechentechnik 477  
Hutfunktion 112, 406  
hyperbolische Differenzialgleichung 427f.  
  
ideal elastisch 268, 399  
implizite Deflation 210  
implizite Differenzialgleichung 343, 397  
implizite Methode 456  
implizite Skalierung 42  
implizite Spektralverschiebung 257  
implizite Vorkonditionierung 542  
implizites Adams-Verfahren 363  
implizites Einschrittverfahren 380  
implizites Euler-Verfahren **346**, 350  
implizites Mehrschrittverfahren 368  
implizites Runge-Kutta-Verfahren 380  
IMSL 14, 27, 87, 338, 392  
Infoplaner 483  
inhärente Instabilität 376f.  
inkorrekt gestelltes Problem 206, 296  
Instabilität 219  
    inhärente 376  
    numerische 15  
instationär 427  
Integralgleichung 115, 307, 344  
Integraltransformation 307  
Integrand  
    exponentiell abklingend 312  
    mit Singularität 315, 320  
    mit Sprungstelle 315  
    periodischer **316**  
Integration  
    adaptive **332**, 334

- Aufwand 308
- im  $\mathbb{R}^n$  336
- mehrdimensionale **336**
- numerische 91, 177, **307**
- Transformationsmethode 315
- über Standardgebiete **337**
- über unbeschränkte Intervalle 320
- von Tabellendaten 307
- Integrationsbereich, unendlicher 330
- Integrationsgewicht 307, 330
- Integrationsregel
  - Genauigkeitsgrad 309, 324f., 331
  - optimale 331
- Interpolation 365, 513f., 516, 521
  - Hermite- **98**
  - inverse **100**
  - inverse quadratische 197
  - Kurven- **125**
  - Lagrange- 94
  - Newton- **95**
  - Spline- 126
  - Tschebyscheff- **170**
- Interpolationsfehler 93
- Interpolationspolynom 92, 119, 308, 373
- Interpolationsverfahren 363
- Intervalhalbierung 191, 333
- inverse Interpolation **100**, 197
- inverse Vektoriteration **231**, 260
- Inversion einer Matrix 45
- involtorische Matrix 287
- Inzidenzmatrix 82
- irreduzibel 433, **496**, 497f., 500, 502, 506
- Iterationsmatrix 492f., 496
- Iterationsverfahren **184**
  - einstufiges 184, 493
  - stationäres 184
  - zweistufiges 195
- J-Verfahren *siehe* Jacobi-Verfahren
- Jacobi 222, 489
- Jacobi-Iteration, gedämpft **491**, 509, 511, 525
- Jacobi-Matrix 199, 202, 205, 215, 298, 300, 386f., 389
  - diskretisierte 206
  - funktionale 206
  - singuläre 205
- Jacobi-Rotation 222, 233
- Jacobi-Verfahren **490**, 496, 500, 505
  - klassisches 224
  - Rechenaufwand 226, 228
  - zyklisches 228
- JOR-Verfahren **491**, 493, 496, 500
- Kante, gerichtete 497
- Kehrwert, iterative Berechnung 194
- Kennlinie 278
- Kettenregel 21
- kinetische Reaktion 387
- Kirchhoff 82
- klassisches Jacobi-Verfahren 224
- klassisches Runge-Kutta-Verfahren 357
- kleinste-Quadrate-Methode 118, 124, 205, 215, 274, 280, 296, 302, 304, 405, 548, 550
- Knoten eines Graphen 497
- Knotenpunkt 114, 124f., 470f.
- Knotenvariable 470f., 477
- Koeffizienten-Bedingungen bei Runge-Kutta-Verfahren 354
- Kollokation **404**, 407, 424
- Kommunikation 68, 70
- kompatible Matrixnorm **49**, 494
- kompatible Vektornorm 494
- Kompilationsprozess 477, 483
- komplexe Fourier-Transformation 155, 157, 161
- komplexe Nullstellen 190, 211
- Komplexität 523
- Kondition **52**, **189**, 220, 261, 377
- Konditionszahl 19f., 25, **53**, 54ff., 61, 190, 218, 540, 544
- Eigenwertproblem **266**
- lineares Gleichungssystem 265
  - Normalgleichungen 277f.
- konforme Abbildung 307
- Kongruenztransformation 541
- konjugierte Richtung 532
- konjugierter Gradient 530, 532, 534, 548
- konsistente Fixpunktiteration 495
- Konsistenz **350**, 352, 355, 369, 373
  - Einschrittverfahren **351**
  - Mehrschrittverfahren **369**
- Konsistenzordnung **351**, 352, 354, 357, 366, 369f., 372
  - erreichbare 357
  - Mehrschrittverfahren 369
- kontinuierliche Fehlerquadratmethode 405

- kontinuierliche Gauß-Approximation **144**,  
**165**
- kontrahierend **186**, 192, 200, 494
- Konvergenz 189, 199, 205, 349, **350f.**, 352,  
**358**  
 asymptotische 496  
 des SOR-Verfahrens 505  
 GMRES( $m$ )-Algorithmus 554, 556  
 kubische **186**  
 lineare **186**, 188, 193  
 quadratische **186**, 227, 250  
 superlineare 193, 197, 315
- Konvergenzgeschwindigkeit 186, 505
- Konvergenzordnung **186**, 188, 191f., 196,  
 200f., 214, **351**, 366, 372
- Konvergenzrate **186**, 200, 511, 524
- Konvergenzrate, asymptotische 509
- konvexe Hülle 130, 132, 138
- Konzentrationsfunktion 459
- Korrektur-Schema 513, 516
- Korrekturansatz 297
- Korteveg-de Vries-Gleichung 124
- Kräfte im Fachwerk 85
- Krogh 376
- Kronrod-Integration 334
- Krylov-Unterraum 537f., 548f., 551f.
- Kubikwurzel, iterative Berechnung 194
- kubische B-Splines **114**
- kubische Konvergenz **186**, 259
- kubische Splines 106, 114, 424
- Kuntzmann 360
- Kurve  
 glatte 91  
 kleinster Krümmung 107  
 Parameter- 125
- Kurveninterpolation **125**
- Kutta 353
- L-stabil *siehe* Lipschitz-stabil
- $L_1$ -Norm **48**
- $L_2$  144, 164
- $L_2$ -Skalarprodukt 174  
 gewichtetes 176
- Lagrange-Interpolation 94, **95**, 114, 418
- Lagrange-Polynom **95**, 102, 308, 325, 470,  
**473**
- Laguerre-Polynom 331
- laminare Strömung 84
- Landvermessung 284
- LAPACK 68, 87, 559
- Laplace-Gleichung 428
- Laplace-Operator 487
- Legendre 140
- Legendre-Polynom 162, **174**, 323ff., 327, 331  
 Nullstellen 176  
 Rekursion 176
- linear unabhängig 140, 230, 274
- lineare Ausgleichung 143  
 Normalgleichungen **274**
- lineare Konvergenz **186**, 193, 200
- lineare Transformation 147, 157, 163, 326,  
**473**
- lineares Mehrschrittverfahren **368**
- Linearisierung 200, 297, 416f.
- Linksdreiecksmatrix 34, 37
- Linkssingulärvektor 294
- LINPACK 87
- Lipschitz-Bedingung 199, **344**, 345, **352**, 354  
 an  $f_h$  352
- Lipschitz-Funktion 352
- Lipschitz-Konstante 186f., 202, **344**, 345,  
**369**, 494
- Lipschitz-stabil 372, 384
- Lipschitz-stetig **186**
- Logarithmentafel 97, 106
- lokale Stabilität 389
- lokaler Abschneidefehler *siehe* Diskretisierungsfehler
- lokaler Diskretisierungsfehler *siehe* Diskretisierungsfehler
- lokales Minimum 205
- Lösungen  
 Eindeutigkeit von 184  
 Existenz von 184
- Lotka-Volterras Wettbewerbsmodell 388
- lowest upper bound 50
- $LR$ -Zerlegung 68, 76, 202, 456, 463
- lss *siehe* kleinste Quadrate-Methode
- lub-Norm 50
- Luftfahrt 483
- $M$ -Matrix 78
- Maehly 209
- Mantisse 16
- Marquardt-Verfahren 302f.
- Maschinengenauigkeit 16, 23, 28, 214, 333
- Maschinenkonstanten 28
- Maske *siehe* Template

- Massenelementmatrix 475f.  
 MATLAB 14, 28, 81, 87, 179, 215, 270, 305,  
     339, 391, 424, 478, 482, 559  
**Matrix**  
     Hessesche 275  
     inverse 45  
     involutorische 287  
     Inzidenz- 82  
     orthogonale 220  
     positiv definite **56**  
     rechteckige 118  
     reguläre 30  
     schwach besetzte *siehe* dort  
     Speicherung 61, 284  
     symmetrische 52  
     tridiagonale 64, 66, 73, 76, 78, 232, 237,  
         244, 246, 420, 453, 455f., 459, 463f.  
**Matrix-Vektor-Multiplikation** 557f.  
**Matrixmultiplikation** 68  
**Matrixnorm** **48**, 49f.  
     kompatibel 494  
     natürliche 50  
     submultiplikative 48, 264, 344  
     verträgliche 51, 344  
     zugeordnete 51f.  
**Matrixprodukt** 87  
**Maximalrang** 274f., 279, 292f., 303  
**Maximumnorm** **48**, 49, 51, 53  
**Medizin** 342  
**mehrdimensionale Integration** **336**  
**Mehrzahl-Schießverfahren** **413**  
     Startwerte 416  
     Zwischenstelle 416  
**Mehrgittermethode** 487, 491  
**Mehrgittermethoden** **508**  
**Mehrgitterzyklen** 522  
**Mehrschrittverfahren** **346**, 363, 368, 380  
     lineares **368**  
     lokaler Diskretisierungsfehler 370  
     Startrechnung 375  
     vom Adams-Typ 363  
**Mehrstellenoperator** 446  
**Mehrstufenmethode** 522  
**Membranschwingung** 268  
**Messfehler** 15, 91  
**Messung** 274  
**Methode der kleinsten Quadrate** 118, 124,  
     205, 215, 274, 280, 296, 302, 304,  
     405, 548, 550  
     Methode des steilsten Abstiegs 206, **301**, 532  
     Minimalpunkt 531f.  
     Minimax-Eigenschaft 539  
     minimierte Residuen 548  
         verallgemeinerte 547  
     Minimierungsaufgabe 548, 550, 552  
     Minimierungsverfahren 300  
     Minimum, lokales 206  
     MINRES 548  
     Mittelpunktregel **346**, 347f., 363  
     Mittelpunktssumme 310  
     Modalmatrix 264ff.  
     Modellierung **13**  
     Modellproblem 508, 513, 520  
     Modellproblem, Anfangswertaufgabe 348,  
         371, 379, 382  
     Modellproblem, Randwertaufgabe 505, 540  
     modifiziertes Newton-Verfahren 195, 205  
     monoton fallend 205  
     Monte-Carlo-Methode 338  
     MSV *siehe* Mehrschrittverfahren  
     multiple shooting **413**  
     Nachiteration **45**, 47, 56, 61, 512  
     NAG-Bibliotheken 14, 27, 87, 161, 179, 215,  
         270, 304, 339, 391, 483, 559  
     Näherungswert 184  
     natürliche Matrixnorm 50  
     Naturgesetz 274  
     natürliche Splinefunktion 112  
     NETLIB 14  
     Netzerzeugung 482f.  
     Neumann-Randbedingung 429, 434, 436,  
         440, 444f., 469  
     Neumannsche Randwertaufgabe 429  
     Neun-Punkte-Formel 447  
     Newton-Cotes-Formel **308**, 323, 325  
     Newton-Interpolation 94, **95**, 105  
     Newton-Maehly, Verfahren von 209  
     Newton-Schema 97, 99  
     Newton-Verfahren 192, **192**, 196ff., 200, **200**,  
         205, 207, 211, 298, 358f., 409, 411,  
         414ff., 422  
         modifiziertes 195, 205  
         Varianten 195  
         vereinfachtes 195, 202  
     nichtlineare Ausgleichung **296**  
     nichtlineare Randwertaufgabe 422  
     nichtlineares Gleichungssystem 183, **199**,  
         205, 358f., 409

- niederfrequent 510  
Niveaulinie 477, 479ff.  
Norm 140, 185  
  verträgliche 49  
Normäquivalenz 48f.  
Normalableitung 436  
Normaldreieck 471, 473  
Normalgleichungen 141, 205, **275**, 278, 282, 295, 302  
  Konditionszahl 277f.  
  lineares Ausgleichsproblem **274**  
  Rechenaufwand 276  
normalisierte Gleitpunktzahl 16  
Normalverteilung 274  
Normen **47**  
Null-stabil 372  
null/null-Situation 25  
Nullraum 515  
Nullstellen **183**  
  doppelte 189  
  enge 189  
  gut getrennte 189  
  komplexe 190, 211  
  Legendre-Polynom 176  
  Polynom 207  
  Schranken für 213  
Nullstellenpaare 190  
numerische Differenziation 91, **101**, 374, 409  
numerische Integration 149, 177, **307**  
numerische Stabilität 15, 23, **25**, 39, 115, 169, 178, 219, 261, 327, 349f.  
Nummerierung  
  Gitterpunkte 432f., 487  
  optimale 477  
Nyström 380  
  
Oberflächen 307  
Odd-Even-Elimination 79  
ODE 392  
ODEPACK 392  
Ohm 82  
optimale Integrationsregel 331  
optimale Nummerierung 477  
optimaler Relaxationsparameter 505, **505**  
optimales Runge-Kutta-Verfahren 357  
Optimalitätseigenschaft 538  
Optimierung 304, 336  
orthogonal 220  
orthogonal-ähnlich 245  
orthogonale Ähnlichkeitstransformation 232  
orthogonale Matrix 220  
orthogonale Transformation 278, 283, 304, 405, 550f.  
  Speicherplatzbedarf 286  
orthogonales Funktionensystem 143, 145f., 151, 164, 174  
orthogonales Polynom **161**, 331  
Orthogonalisierung 550  
  Schmidtsche 549  
Orthogonalitätsrelation 536  
orthonormierte Basis 51, 548  
Oszillationen 106, 119  
Oszillationseigenschaft 397, 510  
oszillatorischer Fourier-Term 510  
  
parabolische Anfangsrandwertaufgabe **448**  
parabolische Differenzialgleichung 428  
Parallelisierung 545  
Parallelrechner 14, 67, 76, 87, 482  
Parameter 274  
Parameter-Identifikation 424  
Parameterkurve 125  
Partial Differential Equation Toolbox 482  
partielle Cholesky-Zerlegung 545  
partielle Differenzialgleichung 124, 268, 337  
  Diskretisierung 487  
Partitionsverfahren 82  
Patterson-Integration 332, 334f., 339  
PB-BLAS 87  
PBLAS 70, 87  
PCG-Algorithmus 544, **544**  
Peaceman und Rachford 463  
periodische Splinefunktion 112  
periodischer Integrand **316**  
Permutationsmatrix 37, 292  
Pharmakologie 342  
physiologische Indidaktormodelle 342  
Pivotelement 32, 35f., 38f., 58  
Pivotstrategie **38**  
Plattengleichung 447  
PLTMG 480, 482  
Poisson-Gleichung 428, 436, 444, 447, 462  
Poissonsche Summenformel 318f.  
Polarkoordinaten 126  
Polygonzug-Verfahren **345**  
  verbessertes 357  
Polynom  
  Bézier- **127**

- Bernstein- **127**
- charakteristisches **219**, 265, 369
- erzeugendes 369
- Hermite- 331
- interpolierendes 308
- komplexe Nullstellen 211
- komplexes 207
- Laguerre- 331
- Legendre- 162, **174**, 331
- orthogonales **161**, 331
- stückweises 106
- Tschebyscheff- **162**, 331
- Polynomdivision 207
- Polynominterpolation 171, 363
- Polynomnullstellen 190, **207**
- Polynomwertberechnung 97
- Populationsmodell 388
- Portabilität 67
- positiv definit **56**, 57f., 62, 64, 78, 80, 82f., 115, 261, 275, 283, 420, 433, 500, 530f., 541, 544
- Potenzial 82f.
- Potenzreihe 105, 167, 307, 313, 420
- Powell 205
- Prädiktor-Korrektor-Verfahren 366, 382, 391
- Primfaktorzerlegung 161
- Produktintegration **336**
- Programmiersprachen 28
- Programmpakete 14
- Prolongation 513ff., 521
- Prozessor 68, 78
- Pumpe 84
  
- QR*-Algorithmus **243**, 328
  - Eigenvektoren 260
  - mit expliziter Spektralverschiebung 250
  - tridiagonale Matrix 256
    - Rechenaufwand 259
    - von Francis 248
- QR*-Doppelschritt 255
- QR*-Transformation **243**, **245**, 248, 255
  - gestaffelte 250
  - komplexe Eigenwerte 253
  - Rechenaufwand 245
  - reelle Eigenwerte 248
- QR*-Zerlegung 124, **243**, 280, 551
- QUADPACK 338
- Quadrat, Methode der kleinsten 118, 124, 205, 215, 274, 280, 296, 302, 304, 405, 548, 550
  
- quadratische Form 56ff., 275
- quadratische Interpolation 197
- quadratische Konvergenz **186**, 192, 227, 250
- quadratisches Gleichungssystem 124
- Quadratur *siehe* Integration
- Quadratwurzel 194
- QZ*-Algorithmus 261
  
- Räuber-Beute-Modell 388
- Rückwärtendifferenzierungsmethode 374
- Randbedingungen mit Ableitungen 421
- Randelementmethode 307
- Randkurve 138
- Randwertaufgabe 540
  - Dirichletsche 429
  - elliptische 487, 505, 530, 545f., 555
  - Neumannsche 429
  - nichtlineare 205, 422
- Randwertlinearisierung 416
- Randwertproblem **395**
  - Linearisierung 417
- Raumfahrt 483
- Rayleighsches Prinzip **467**
- Reaktion, kinetische 387
- Reaktions-Diffusionsgleichung 483
- Realisierung **13**
- Rechenaufwand 487, 524
  - QR*-Algorithmus
    - tridiagonale Matrix 259
  - QR*-Transformation 245
  - Band-Cholesky 63
  - CG-Algorithmus 534
  - Cholesky-Verfahren 60
  - Cholesky-Zerlegung 60
  - Clenshaw-Algorithmus 169
  - Crank-Nicolson 456
  - Eigenvektor 260
  - Fehlergleichungen 276
  - Fourier-Transformation 158
  - Gauß-Algorithmus 38
  - Gauß-Jordan 72
  - Givens 237, 281
  - Givens, schnell 284
  - GMRES( $m$ )-Algorithmus 556
  - GMRES-Algorithmus 553
  - Hessenberg-Form 234
  - Householder, Fehlergleichungen 290
  - Householder, Rücktransformation 290
  - Householder-Transformation 290

- Matrixinversion 45
- Normalgleichungen 276
- Odd-Even-Elimination 81
- Polynomwertberechnung 207
- pro CG-Schritt 534
- QR*-Doppelschritt 256
- Rücksubstitution 38
- Relaxation 541
- RQ*-Zerlegung 78
- schnelle Givens-Transformation 242
- tridiagonales Gleichungssystem 65, 67
  - Parallelrechner 78
  - Vektorrechner 76
  - vorkonditionierter CG-Algorithmus 545
- Vorwärtssubstitution 38
- Zerlegung 37
- Rechenoperation, elementare **15**, 21
- Rechentechnik, hüllenorientierte 477
- Rechteckgitter 120, 123, 125
- rechteckige Matrix 118
- Rechtsdreiecksmatrix 34, 37
- Rechtssingulärvektor 294
- Reduktion, zyklische 73, 76, 82
- reduzibel 498
- reguläre Matrix 30
- reguläre Lösung 468
- Regularisierung 206, 296, 302
- Reinsch-Algorithmus 155
- Rekursion 95f., 114, 128, 164, 169, 327, 333
  - Legendre-Polynom 176
- rekursive Verdoppelung 81
- relative Spaltenmaximumstrategie 42, 46, 65f., 232
- relativer Fehler **16**, 20, 53ff.
- Relaxation 422, 516
- Relaxationsfaktor 204f.
- Relaxationsparameter 491
  - optimaler 501, 505, **505**
- Relaxationsverfahren **489**
- Residuen
  - minimierte 548
  - verallgemeinerte minimierte 547
- Residuen-Norm 548
- Residuenvektor 45, 530, 532, 535, 541
- Residuum 52, 267, 274, 295
- Ressource 388
- Restabbildung 21
- Restriktion 513, 515f., 520
- Resultatrundung 21
- Richardson
  - explizite Methode von 450f., 461f.
- Richardson-Extrapolation 313, 421
- Richtung, konjugierte 532
- Richtungsvektoren, konjugierte 535
- Rohrnetz 84
- Rolle, Satz von 93, 176
- Romberg-Integration **313**, 314, 323, 331
  - Fehlerabschätzung 314
  - Genauigkeitssteuerung 315
- Rotation 87, 551
- Rotationsmatrix **220**, 279
- RQ*-Zerlegung 76, 78
- Rücksubstitution 33, 37, 43f., 46, 59f., 71, 76, 78
- Rückwärts-Euler-Verfahren 383
- Rückwärts-Fehleranalyse **24**, 55
- Rückwärtsdifferenz 346, **418**
- Rückwärtsdifferenziationsmethode 383
- Rundungsfehler **15**, **18**, 53, 55, 145, 169, 189, 206, 209, 266
  - bei Anfangswertproblemen 352
  - fortgepflanzte 19, 21, 40
  - Matrix der 21
- Runge-Kutta-Nyström 391
- Runge-Kutta-Verfahren **353**, 354, 368, 378, 388, 410f.
  - 2. Ordnung 356f.
  - 3. Ordnung 356
  - 4. Ordnung 360
  - eingebettete **359**
  - explizites **354**, 356
  - Gauß-Form 358
  - halbimplizites 358, **358**
  - implizites 358, 380
  - klassisches 357f., 362
  - Koeffizienten-Bedingungen 354
  - m*-stufiges 354
  - optimales 2. Ordnung 357
- Rutishauser 13
- Satz von Rolle 176
- Satz von Taylor 19
- SCALAPACK 70, 87
- Schachbrett-Nummerierung 506
- Schädlingsbekämpfung 342
- Schaltkreistheorie 82
- Schätzung, Diskretisierungsfehler 361
- Schema

- Horner- 97
- Newton- 99
- Schießverfahren 395, **408**
  - Einfach- **408**, 412, 416
  - Mehrfach- **413**
- Schiffsbau 107
- schlecht konditioniert 206
- Schmidtsche Orthogonalisierung 549
- schnelle Fourier-Transformation **155**, 159, 161f.
- schnelle Givens-Transformation 256
  - Rechenaufwand 242
- Schnittpunkt 184
- Schrittweite
  - Halbierung der 375
  - variable 376
  - Verdoppelung der 375
- Schrittweitensteuerung **359**, **361**, 366, 375
  - Fehlerschätzung 360
  - Strategie 359
- Schur, Satz von 246
- schwach besetzt 86, 124, 230, 270, 284, 433, 482f., 559
  - Speicherung 556
- schwach diagonal dominant **39**, 496, 498, 500, 502
- schwach gekoppelt 204
- schwache Lösung 468
- Schwarz-Christoffel-Formel 307
- Schwarzsche Ungleichung 49
- Schwingung 218, 268
- Scientific Computing 13
- Sechsecknetz 429
- Sekantenverfahren **195**, 197f., 410
- selbstadjungiert 396
- Separatrix 389
- shooting 409
- Simpson-Regel 308f., 314f., 332, 339, 476
  - Fehlerabschätzung 310
- singuläre Jacobi-Matrix 205
- Singularität 307, 330
  - bei einspringender Ecke 445, 447, 479
  - Integrand mit 315, 320
- Singulärwerte 270, 294, 559
- Singulärwertzerlegung 261, **293**, 294, 301, 304
- sinh-Substitution 321
- Skalarprodukt 87, 140, 165, 276
  - euklidisches
- gewichtetes 142
- Skalarrechner 556
- Skalierung 41, 544
  - implizit 42
- SLATEC 338, 559
- smoothing factor 511
- Sobolev-Raum 468
- Software **14**
  - Software-Bibliothek *siehe* NAG, IMSL
  - Softwareengineering 482
  - Softwaresystem 206, 344
- Soliton 125
- SOR-Newton-Verfahren **203**, 204
- SOR-Verfahren **491**, 493, 502f., 507, 540, 556, 558
  - Spaltenmaximumstrategie 40
    - relative 42, 46, 65, 232
  - Spaltenpivotisierung 70
  - Spaltensummennorm **48**
  - Spannung 82
  - Spannungszustand 428, 432
  - Speicherplatzbedarf 271, 285, 487, 523
    - orthogonale Transformation 286
  - Speicherung
    - Bandmatrix- 63
    - Matrix- 61, 284
    - schwach besetzter Matrizen 556
  - Spektralnorm 52, 55f., 218, 494
  - Spektralradius 200, **264**, 494, 505
  - Spektralverschiebung 249, 253, 256
    - implizite 255, 257
  - Spektrum 264
  - spezielles Eigenwertproblem 218, 261
  - Spiegelung 287
  - Spirale, Archimedes- 126
  - Spline Toolbox 179
  - Splinefunktion **106**
    - bikubische 125
    - kubische 106, 424
    - natürliche 112
    - periodische 112
    - vollständige 112
    - zweidimensionale **119**
  - Splineinterpolation 126
  - Sprungstelle
    - Integrand mit 315
  - SSOR-Verfahren 545
  - SSORCG-Verfahren 545, 547
  - stabil 219, 261, 309

- Stabilität 19, 56, 348, **350**, 352  
absolute 350, 378, 452f., 455, 461, 463  
asymptotische 352, 372  
Dahlquist- 372  
der Trapezmethode 346  
für endliche  $h$  378, 380  
Gebiet absoluter 379, 381, 383  
Lipschitz- (L-) 372  
lokale 389  
Null- 372  
numerische 23, 178  
Stabilitätsgebiet 380  
Stabilitätsintervall 380  
Stammfunktion 307  
Standarddreieck 337  
Standardfunktion 15  
Standardgebiete  
Integration über **337**  
Startintervall 191  
Startrampen 418  
Startrechnung 368, 375  
mit Einschrittverfahren 375  
Startwerte, beim Mehrfach-Schießverfahren 416  
stationär *siehe* zeitunabhängig  
stationärer Punkt 389  
stationäres Iterationsverfahren 184  
Statistik 218, 274, 295  
steife Differenzialgleichung 358, **384**, 385, 391  
Steifheitsmaß 385, 388  
Steifigkeitselementmatrix 474, 478  
steinster Abstieg 206, 301, **532**  
Stiefel 532  
Stirling'sche Formel 26  
Stochastik 23, 338  
Stone-Verfahren 82  
Straklatte 107  
strikt diagonal dominant **39**, 502  
Stromquelle 82  
Stromstärke 82  
Strömung, laminare 84  
Strömungslehre 428  
Strömungsmechanik 483  
Strukturmechanik 483  
stückweises Polynom 106  
Sturm-Liouvillesche Eigenwertaufgabe 397  
Sturmsche Randwertaufgabe 396, 399, 406  
Stützstelle 92, 99  
Stützstellenverteilung 93  
Stützwert 92, 99  
submultiplikative Matrixnorm 48, 264  
Substitution 320, 322  
algebraische 320  
exp- 322  
sinh- 321  
tanh- 321  
successive overrelaxation 491  
sukzessive Approximation 184  
sukzessive Deflation 209  
sukzessive Überrelaxation 491  
superlineare Konvergenz 193, 315  
Symbiosemodell 342  
symmetrisch **56**, 62, 83  
symmetrische Überrelaxation 545  
symmetrische Matrix 52  
symmetrisches Eigenwertproblem 220, **261**
- T-Entwicklung *siehe* Tschebyscheff-Entwicklung  
T-Polynom *siehe* Tschebyscheff-Polynom  
Tabellendaten, Integration von 307  
Tangente 192, 195  
tanh-Substitution 321  
Taylor, Satz von 19  
Taylor-Entwicklung 435, 441, 444, 455  
Taylor-Reihe 104, 346, 369, 378, 380, 420  
Teilintervallmethode **405**  
Temperaturverteilung 428, 432, 439, 449  
Template 87, 271, 559  
Tensorprodukt 122, 137  
Tensorspline 123  
bikubischer **123**  
bilinearer **120**  
Tetraeder, Integration über 337  
Textverarbeitung 136  
Torsion 428, 432  
Trägerintervall 115  
Tragflügel 136  
Transformation  
lineare 147, 157, 163, 326, 473  
orthogonale 551  
Transformationsmethode 232  
bei Integration 315  
transversale Belastung 399  
Trapez-Simpson-Integration  
adaptive 333  
Trapezapproximation, sukzessive 311

- Trapezmethode **346**, 348, 380, 455  
 Stabilität 346
- Trapezregel 149, 168, 309, 313, 315, 321f., 332, 346, 420  
 Fehlerabschätzung 310
- Triade 71
- Triangulierung 469, 471
- tridiagonale Matrix 64, 66, 73, 76, 78, 232, 237, 246, 420, 453, 455f., 459, 463f.  
**QR**-Algorithmus 256
- trigonometrische Approximation **145**
- trigonometrische Funktion 145
- Trivialzustand 389
- Tschebyscheff-Abszisse 171
- Tschebyscheff-Approximation 274
- Tschebyscheff-Entwicklung 166  
 Fehlerabschätzung 166
- Tschebyscheff-Polynom 94, **162**, 331, 407, 539  
 Extremalstellen 163  
 Interpolation **170**
- Tschebyscheff-Punkt 94
- Tukey 159
- überbestimmt 274, 296
- Übergangsbedingung 413
- Überrelaxation 204, **491**  
 symmetrische 545
- unbeschränkte Intervalle, Integration über 320
- unendliche Schleife 206
- unendlicher Integrationsbereich 330
- Unterrelaxation **491**
- unzerlegbar *siehe* irreduzibel
- V-Zyklus 522
- Validierung **14**
- Vandermondesche Determinante 371
- variable Schrittweite 376
- Variablensubstitution 320
- Varianzanalyse 218
- Variationseigenschaft 515
- Variationsproblem 468, 478, 480
- Variationsrechnung 108, 467
- Vektorfunktion 183, 206
- Vektorisierung 545
- Vektoriteration **229**, 249  
 inverse 231, 260
- Vektornorm **47**, 49  
 euklidische 49, 51f.
- kompatible 494
- Vektoroperation 71
- Vektorprozessor 67
- Vektorraum 185
- Vektorrechner 14, 67, 70, 73, 87
- verallgemeinerte minimierte Residuen **547**
- verbessertes Polygonzug-Verfahren 357
- Verdoppelung, rekursive 81
- vereinfachtes Newton-Verfahren 195
- Verfeinerung, Gitter- 270
- Vergleich von Iterationsverfahren 198
- Verstärkungsfaktors 525
- verträgliche Matrixnorm **49**, 51
- Verzweigungsproblem 344
- Vierecke, Integration über 337
- Viergittermethode 522
- Viskosität 84
- Visualisierung 13
- vollständig konsistent 494f.
- vollständige Splinefunktion 112
- von Mises, Vektoriteration nach 229
- Vorkonditionierung 540, **541**, 544, 546f., 554ff., 558  
 implizite 542  
 Konvergenz 547
- Vorkonditionierungsmaatrix 542, 544f.
- Vorwärtssubstitution 35, 37, 43f., 46, 59f., 66, 78
- Vorwärtsdifferenz 130, 345, **418**, 450
- Vorwärtsdifferenzenquotient 462
- W-Zyklus 522
- Wachstumsgesetz 388
- Wahrscheinlichkeit 274, 307
- Wärmeabstrahlung 449
- Wärmeleitung 439, 448f., 483
- Weierstraßscher Approximationssatz 127
- Wellengleichung 427
- Werkzeug **14**
- wesentliche Stellen 18
- wesentliche Wurzel 372
- Wettbewerbsmodell 342, 388
- Widerstand 82
- Wilkinson 260
- Wirkungsquerschnitt 307
- wissenschaftliches Rechnen 13
- Wraparound 82
- Wronsky-Matrix 402
- Wurzel *siehe* Nullstelle

- wesentliche 372  
Wurzel, *m*-te 193  
Wurzelbedingung 372f.
- Zahldarstellung **16**  
Zahlensystem 16  
Zeichenentwurf 136  
Zeigervektor 556, 558  
Zeilenpermutation *siehe* Zeilenvertauschung  
Zeilensummennorm **48**, 51, 53, 494  
Zeilenvertauschung 35f., 40, 43, 46, 58, 64  
zeitunabhängig 395, 427, 439  
zentrale Differenz **418**  
zentraler Differenzenquotient 103
- Zerlegung 35, 37, 43, 67  
Zerlegung der Eins 127  
Zufallsgenerator 338  
zugeordnete Matrixnorm 50ff.  
zusammenhängender Graph 497  
zweidimensionale Splinefunktion **119**  
Zweigitter-Korrektur-Schema 516  
Zweigittermethode 511, 513, 522  
Zweischritt-Verfahren 346  
zweistufiges Iterationsverfahren 195  
Zwischenstelle  
    Mehrfach-Schießverfahren 416  
zyklische Reduktion 73, 76, 82  
zyklisches Jacobi-Verfahren 228