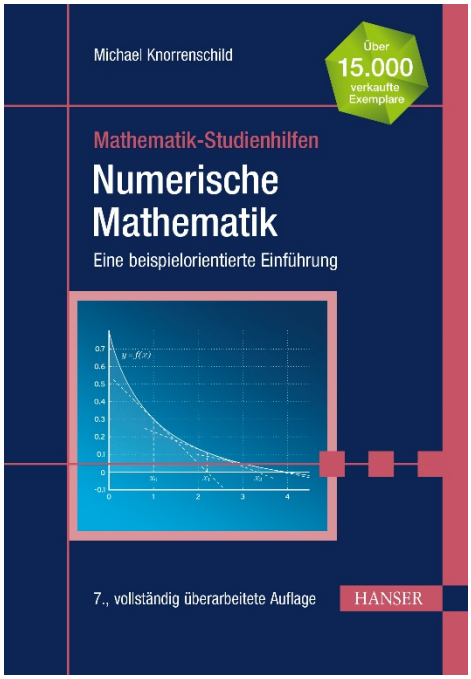
HANSER



Leseprobe

zu

Numerische Mathematik

von Michael Knorrenschild

Print-ISBN: 978-3-446-46916-7
E-Book-ISBN: 978-3-446-46959-4

Weitere Informationen und Bestellungen unter
https://www.hanser-kundencenter.de/fachbuch/artikel/9783446469167
sowie im Buchhandel

© Carl Hanser Verlag, München

Inhalt

1 Rechnerarithmetik und Gleitpunktzahlen 1
1.1 Grundbegriffe und Gleitpunktarithmetik 1
1.2 Auslöschung 8
1.3 Fehlerrechnung 9
1.3.1 Fehlerfortpflanzung in arithmetischen Operationen 10
1.3.2 Fehlerfortpflanzung bei Funktionsauswertungen 11

2 Numerische Lösung von Nullstellenproblemen 17
2.1 Problemstellung 17
2.2 Das Bisektionsverfahren 18
2.3 Die Fixpunktiteration 19
2.4 Das Newton-Verfahren und seine Abkömmlinge 24
2.5 Konvergenzgeschwindigkeit 28
2.6 Das Horner-Schema – schnelle Auswertung von Polynomen 29

3 Numerische Lösung linearer Gleichungssysteme 33
3.1 Problemstellung 33
3.2 Der Gauß-Algorithmus 34
3.3 Fehlerfortpflanzung beim Gauß-Algorithmus und Pivotisierung 39
3.4 Dreieckszerlegungen von Matrizen 41
3.4.1 Die LR-Zerlegung 41
3.4.2 Die Cholesky-Zerlegung 44
3.4.3 Die QR-Zerlegung 46

3.5 Fehlerrechnung bei linearen Gleichungssystemen 53
3.6 Iterative Verfahren 58

4 Numerische Lösung nichtlinearer Gleichungssysteme 67
4.1 Problemstellung 67
4.2 Das Newton-Verfahren für Systeme 69

5 Interpolation 73
5.1 Problemstellung 73
5.2 Polynominterpolation 74
5.2.1 Das Neville-Aitken-Schema 78
5.2.2 Der Fehler bei der Polynominterpolation 80
5.3 Splineinterpolation 84
5.3.1 Problemstellung 84
5.3.2 Interpolation mit kubischen Splines 85

6 Ausgleichsrechnung 93
6.1 Problemstellung 93
6.2 Lineare Ausgleichsprobleme 95
6.3 Nichtlineare Ausgleichsprobleme 101
6.4 Das Gauß-Newton-Verfahren 102

7 Numerische Differenziation und Integration 107
7.1 Numerische Differenziation 107
7.1.1 Problemstellung 107
7.1.2 Differenzenformeln für höhere Ableitungen 112
7.1.3 Differenzenformeln für partielle Ableitungen 112
7.1.4 Extrapolation 113
7.2 Numerische Integration 120
7.2.1 Problemstellung 120
7.2.2 Interpolatorische Quadraturformeln 123
7.2.3 Der Quadraturfehler 124
7.2.4 Transformation auf das Intervall $[a, b]$ 125
7.2.5 Der Fehler der summierten Quadraturformeln 127
7.2.6 Newton-Cotes-Formeln 128

7.2.7 Gauß-Formeln 129
7.2.8 Extrapolationsquadratur 131
7.2.9 Praktische Aspekte 134

8 Anfangswertprobleme gewöhnlicher Differenzialgleichungen 137
8.1 Problemstellung 137
8.2 Das Euler-Verfahren 139
8.3 Praktische Aspekte 144
8.4 Weitere Einschrittverfahren 145
8.5 Weitere Verfahren 151

Lösungen 153

Literatur 171

Stichwortverzeichnis 173

Vorwort

Numerische Mathematik gehört zu den Teilgebieten der Mathematik, die von Ingenieuren im beruflichen Alltag verwendet werden. Durch verstärkte Verwendung von Computer-Simulationen in allen Bereichen erhöht sich die Bedeutung dieses Themas, in dem Fragestellungen der Mathematik und der Informatik zusammenkommen, zunehmend.

Der vorliegende Band deckt die wichtigsten Themen der numerischen Mathematik für Studierende der Ingenieurwissenschaften ab und entspricht in etwa dem Umfang einer einsemestrigen Lehrveranstaltung. Das Anliegen ist dabei, die Ideen der wichtigsten numerischen Verfahren zu präsentieren und anhand einer Vielzahl von Beispielen deren charakteristische Eigenschaften zu illustrieren. Auf Beweise und längere Herleitungen wird dabei weitgehend verzichtet. Vorausgesetzt werden Vorkenntnisse zur elementaren Differenzial- und Integralrechnung sowie zur linearen Algebra im Umfang etwa einer Anfängervorlesung zu diesen Themen.

Die Darstellungsweise profitiert von Erfahrungen, die ich in Lehrveranstaltungen zur Numerischen Mathematik für Studierende der Ingenieurwissenschaften an der Rheinisch-Westfälischen Technischen Hochschule Aachen, der Simon Fraser University in Burnaby (Kanada), der Eidgenössischen Technischen Hochschule Zürich und der Hochschule Bochum gesammelt habe. Die Anordnung der Themen folgt der bewährten Reihenfolge von Grundlagen der Gleitpunktarithmetik über die numerische Lösung von eindimensionalen Gleichungen, von linearen und nichtlinearen Gleichungssystemen, die Behandlung von Interpolations- und Ausgleichsproblemen bis hin zu numerischer Differenziation und Integration. Den Abschluss bildet ein Einblick in die numerische Lösung von Anfangswertaufgaben gewöhnlicher Differenzialgleichungen.

Die Entstehung und Weiterentwicklung dieses Buchs wurde im Laufe der Jahre von verschiedener Seite tatkräftig und wohlwollend unterstützt. Zuerst ist das Team des Hanser-Verlags zu nennen, beginnend 2003 mit Frau Christine Fritzsch bis heute mit

Frau Natalia Silakova. Für fachliche Ratschläge danke ich dem Herausgeber Prof. Dr. Bernd Engelmann. Herrn Dr. Thomas Schenk gebührt Dank für die kritische Durchsicht weiter Teile des Manuskripts. Für die vorliegende siebte Auflage wurde das Layout überarbeitet, Fehler korrigiert, Formulierungen verbessert und Ergänzungen vorgenommen. Dabei bin ich vielen aufmerksamen Leserinnen und Lesern dankbar. Hinweise und Anregungen aus dem Leserkreis sind auch weiterhin jederzeit willkommen.

Bochum, im März 2021 Michael Knorrenschild

1 Rechnerarithmetik und Gleitpunktzahlen

In der Numerischen Mathematik geht es in der Regel um die näherungsweise Berechnung von Lösungen von Gleichungen oder anderen Größen wie z. B. Funktionswerte oder Integrale mithilfe von Computern. Dies geschieht aus zwei möglichen Gründen:

- Diese Größen sind auf dem Papier nicht exakt berechenbar, also muss es mit anderen Mitteln geschehen.
- Die Größen sind zwar auf dem Papier exakt bestimmbar, aber die Anwendung erfordert, diese wiederholt und zuverlässig in kurzer Zeit zur Verfügung zu stellen, sodass eine Rechnung von Hand auch wieder nicht infrage kommt.

Der Computer hat jedoch zwei prinzipielle Handicaps:

- Er kann durch die beschränkte Stellenzahl nicht alle Zahlen exakt darstellen.
- Er kann die gewünschten Rechnungen nicht exakt ausführen.

Im Folgenden werden Auswirkungen dieser Handicaps anhand von Beispielen und Aufgaben veranschaulicht.

1.1 Grundbegriffe und Gleitpunktarithmetik

Wir beginnen mit der Frage, wie Zahlen auf dem Rechner dargestellt werden. Vom Taschenrechner kennen wir Formate wie z.B: $1.234\,E\,12$, was für $1.234 \cdot 10^{12}$ steht, die sogenannte wissenschaftliche Darstellung. Die Verwendung des Exponenten erlaubt eine Kommaverschiebung und damit große Zahlenbereiche. Auf dem Rechner ist es ganz ähnlich.

Definition

Eine ***n*-stellige Gleitpunktzahl** zur Basis B hat die Form

$$x = \pm(0.z_1 z_2 \dots z_n)_B \cdot B^E \quad \text{und den Wert} \quad \pm \sum_{i=1}^{n} z_i \cdot B^{E-i} \tag{1.1}$$

wobei $z_i \in \{0, 1, \dots, B-1\}$ und, falls $x \neq 0$, $z_1 \neq 0$ (**normalisierte Gleitpunktdarstellung**). Den Anteil $(0.z_1 z_2 \dots z_n)_B$ bezeichnet man auch als **Mantisse**. Für den Exponenten $E \in \mathbb{Z}$ gilt: $m \leq E \leq M$.

Beispielsweise ist also $x = -(0.2345)_{10} \cdot 10^3$ eine 4-stellige Gleitpunktzahl und hat den Wert -234.5.

Übliche Basen sind $B = 2$ (Dualzahlen), $B = 8$ (Oktalzahlen), $B = 10$ (Dezimalzahlen) und $B = 16$ (Hexadezimalzahlen). Für letztere benötigt man für eine eindeutige Schreibweise 16 verschiedene Zeichen, man verwendet dabei die Ziffern $0, 1, \dots, 9$ sowie die Buchstaben $A, \dots, F$, wobei $A \triangleq 10$, $B \triangleq 11$, $\dots$, $F \triangleq 15$. Die Werte n, m, M, B sind maschinenabhängig (wobei unter Maschine der Rechner zusammen mit dem benutzten Compiler zu verstehen ist).

Als Beispiel erwähnen wir die IEC/IEEE-Gleitpunktzahlen. Dabei unterscheidet man zwei Grundformate ($B = 2$):

- **single format** Gesamtlänge der Zahl ist 32 Bit. Dieses teilt sich auf in 1 Bit für das Vorzeichen, 23 Bit für die Mantisse und 8 Bit für den Exponenten.
- **double format** Gesamtlänge der Zahl ist 64 Bit. Dieses teilt sich auf in 1 Bit für das Vorzeichen, 52 Bit für die Mantisse und 11 Bit für den Exponenten.

Das Vorzeichenbit $v \in \{0, 1\}$ erzeugt das Vorzeichen der Zahl über den Faktor $(-1)^v$, d. h. $v = 0$ ergibt positives Vorzeichen, $v = 1$ negatives. Eine umfassende Abhandlung dieser und anderer Formate findet man in [13].

Aufgaben

1.1 Welchen Wert haben die folgenden Gleitpunktzahlen im Dezimalsystem:
$$x_1 = 0.76005 \cdot 10^5, \qquad x_2 = 0.571 \cdot 10^{-3}\,?$$

1.2 Welchen Wert haben die folgenden Gleitpunktzahlen im Dualsystem:
$$x_1 = 0.111 \cdot 2^3, \qquad x_2 = 0.1001 \cdot 2^{-3}\,?$$

1.3 Wie viele Stellen n benötigt man, um die folgenden Zahlen als n-stellige Gleitpunktzahlen im Dezimalsystem darzustellen?
$$x_1 = 0.00010001, \qquad x_2 = 1230001, \qquad x_3 = \frac{4}{5}, \qquad x_4 = \frac{1}{3}$$

Bei der letzten Aufgabe haben Sie festgestellt, dass nicht jede reelle Zahl als Gleitpunktzahl dargestellt werden kann. Dies trifft insbesondere auf Zahlen zu, die

unendlich viele Stellen benötigen würden, beispielsweise kann $\frac{1}{7}$ nicht als Gleitpunktzahl im Dezimalsystem dargestellt werden. Ebenso kann z. B. 12345 nicht als 3-stellige Gleitpunktzahl im Dezimalsystem geschrieben werden. Die Lage ist sogar noch ernster, denn es gilt:

Die Menge der auf einem Rechner darstellbaren Zahlen, die sog. **Maschinenzahlen**, ist endlich. ■

Aufgaben

1.4 Bestimmen Sie alle dualen 3-stelligen Gleitpunktzahlen mit einstelligem Exponenten sowie ihren dezimalen Wert. Hinweis: Sie sollten 9 finden.

1.5 Wie viele verschiedene Maschinenzahlen gibt es auf einem Rechner, der 20-stellige Gleitpunktzahlen mit 4-stelligen Exponenten sowie dazugehörige Vorzeichen im Dualsystem verwendet? Wie lautet die kleinste positive und die größte Maschinenzahl?

Auch sind die Maschinenzahlen ungleichmäßig verteilt. Bild 1.1 zeigt alle binären normalisierten Gleitpunktzahlen mit 4-stelliger Mantisse und 2-stelligem Exponenten.

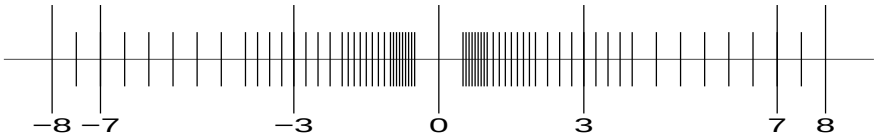


Bild 1.1 Alle binären Maschinenzahlen mit $n = 4$ und $0 \le E \le 3$

Unter den endlich vielen Maschinenzahlen gibt es zwangsläufig eine größte und eine kleinste:

- Die größte Maschinenzahl ist $x_{max} = (1 - B^{-n})\, B^M$,
- die kleinste positive ist $x_{min} = B^{m-1}$. ■

x_{min} basiert auf der normalisierten Gleitpunktdarstellung. Sieht man von der Normalisierung $z_1 \neq 0$ in (1.1) ab, führt dies auf die **subnormalen Zahlen**, die bis hinunter zu B^{m-n} reichen (IEEE Standard 754). Führt eine Rechnung in den Zahlenbereich der subnormalen Zahlen, so bezeichnet man dies als **graduellen Unterlauf** (gradual underflow). Ein (echter) **Unterlauf** (underflow) tritt erst unterhalb der subnormalen Zahlen auf. In diesem Fall wird meist mit Null weitergerechnet.

Taucht im Verlauf einer Rechnung eine Zahl auf, die betragsmäßig größer als x_{max} ist, so bezeichnet man dies als **Überlauf** (overflow). Mit IEEE 754 konforme Systeme setzen diese Zahl dann auf eine spezielle Bitsequenz **inf** und geben diese am Ende aus.[1]
Jede reelle Zahl, mit der im Rechner gerechnet werden soll und die selbst keine Maschinenzahl ist, muss also durch eine Maschinenzahl ersetzt werden. Idealerweise wählt man diese Maschinenzahl so, dass sie möglichst nahe an der reellen Zahl liegt (Rundung).

Definition

Hat man eine Näherung $\tilde{x}$ zu einem exakten Wert x, so bezeichnet $|\tilde{x}-x|$ den **absoluten Fehler** dieser Näherung. ■

Beispiel 1.1
Gesucht ist eine Näherung $\tilde{x}$ zu $x = \sqrt{2} = 1.414213562\ldots$ mit einem absoluten Fehler von höchstens 0.001.

Lösung: $\tilde{x}_1 = 1.414$ erfüllt das Verlangte, denn $|\tilde{x} - x| = 0.000213562\ldots \leq 0.001$. Andere Möglichkeiten sind $\tilde{x}_2 = 1.4139$. $\tilde{x}_1$ stimmt auf 4 Ziffern mit dem exakten Wert überein, $\tilde{x}_2$ nur auf 3. Eine größere Anzahl an übereinstimmenden Ziffern bedeutet aber keinesfalls immer einen kleineren absoluten Fehler, wie das Beispiel $x = \sqrt{3} = 1.732050808\ldots$ und $\tilde{x}_1 = 2.0$, $\tilde{x}_2 = 1.2$ zeigt: $\tilde{x}_1$ hat keine gültige Ziffer, $\tilde{x}_2$ hat eine gültige Ziffer, trotzdem besitzt $\tilde{x}_1$ den kleineren absoluten Fehler. ■

Beim Runden einer Zahl x wird eine Näherung $\mathrm{rd}(x)$ unter den Maschinenzahlen gesucht, die einen minimalen absoluten Fehler $|x - \mathrm{rd}(x)|$ aufweist. Dabei entsteht ein (unvermeidbarer) Fehler, der sog. **Rundungsfehler**. ■

[1] Achtung: IEEE 754 regelt nicht die Rechnung mit integer-Größen. Ein overflow in einer integer-Variablen kann zu falschen Ergebnissen ohne jede Fehlermeldung führen. Hier ist also die besondere Aufmerksamkeit des Benutzers gefordert.

Eine n-stellige dezimale Gleitpunktzahl $\tilde{x} = \pm(0.z_1 z_2 \dots z_n)_B \cdot 10^E = \mathrm{rd}(x)$, die durch Rundung eines exakten Wertes x entstand, hat also einen absoluten Fehler von höchstens

$$|x - \mathrm{rd}(x)| \le 0.\underbrace{00\dots00}_{n\,\text{Nullen}}5 \cdot 10^E = 0.5 \cdot 10^{-n+E}.$$

Rechnet man mit diesen Maschinenzahlen weiter, so werden die entstandenen Rundungsfehler weiter durch die Rechnung getragen. Unter ***n*-stelliger Gleitpunktarithmetik** versteht man, dass jede einzelne Operation (wie z. B. $+, -, *, \dots$) auf $n+1$ Stellen genau gerechnet wird und das Ergebnis dann auf n Stellen gerundet wird. Erst dann wird die nächste Operation ausgeführt. Jedes Zwischenergebnis wird also auf n Stellen gerundet, nicht erst das Endergebnis einer Kette von Rechenoperationen. Von nun an werden wir uns, wenn nichts anderes gesagt ist, auf dezimale Gleitpunktarithmetik beziehen.

Aufgabe

1.6 Bekanntlich ist $\lim\limits_{n\to\infty}(1+\frac{1}{n})^n = \mathrm{e}$. Versuchen Sie damit auf Ihrem Rechner näherungsweise e zu berechnen, indem Sie immer größere Werte für n einsetzen. Erklären Sie Ihre Beobachtung.

Beispiel 1.2
Es soll $2590 + 4 + 4$ in 3-stelliger Gleitpunktarithmetik (im Dezimalsystem) gerechnet werden und zwar zum einen mit Rechnung von links nach rechts und zum anderen von rechts nach links.

Lösung: Alle 3 Summanden sind exakt darstellbar. Als Ergebnis erhält man, bei Rechnung von links nach rechts:

$$2590 + 4 = 2594 \overset{\text{runden}}{\longrightarrow} 2590, \qquad 2590 + 4 = 2594 \overset{\text{runden}}{\longrightarrow} 2590.$$

Die beiden kleinen Summanden gehen damit gar nicht sichtbar in das Ergebnis ein. Rechnet man jedoch in anderer Reihenfolge

$$4 + 4 = 8 \overset{\text{runden}}{\longrightarrow} 8, \qquad 8 + 2590 = 2598 \overset{\text{runden}}{\longrightarrow} 2600$$

so erhält man einen genaueren Wert, sogar den in 3-stelliger Gleitpunktarithmetik besten Wert (2598 wird bestmöglich durch die Maschinenzahl 2600 dargestellt). ∎

Es kommt also bei n-stelliger Gleitpunktarithmetik auf die Reihenfolge der Operationen an, anders als beim exakten Rechnen. Man sieht, dass in der zweiten Rechnung die kleinen Summanden sich erst zu einem größeren Summanden finden, der sich dann auch in der Gesamtsumme auswirkt. Beginnt man die Rechnung mit dem größten Summanden, so werden die kleinen nacheinander vom größten verschluckt und spielen gar keine Rolle mehr. Als Faustregel kann man daher festhalten:

Beim Addieren sollte man die Summanden in der Reihenfolge aufsteigender Beträge addieren. ■

Dadurch erreicht man – bei gleicher Rechenzeit! – ein wesentlich genaueres Ergebnis. Ein eindrucksvolles Beispiel ist das folgende.

Beispiel 1.3

Es soll $s_{300} := \sum_{i=1}^{300} \frac{1}{i^2}$ berechnet werden.

Lösung: Mit dezimaler Gleitpunktarithmetik erhält man

$$s_{300} = 1.6416062828976228698\ldots \quad \text{bei exakter Rechnung}$$
$$s_{141} = s_{142} = \ldots = s_{300} = 1.6390 \quad \text{5-stellig gerechnet, addiert von 1 bis 300}$$
$$s_{300} = 1.6416 \quad \text{5-stellig gerechnet, addiert von 300 bis 1}$$
$$s_{14} = s_{15} = \ldots = s_{300} = 1.59 \quad \text{3-stellig gerechnet, addiert von 1 bis 300}$$
$$s_{300} = 1.64 \quad \text{3-stellig gerechnet, addiert von 300 bis 1.}$$

Bei 3- bzw. 5-stelliger Rechnung und geeigneter Wahl der Summationsreihenfolge wird also das auf 3 bzw. 5 Stellen genaue exakte Ergebnis erzielt.
Dagegen wird bei 3- bzw. 5-stelliger Rechnung und ungeschickter Wahl der Summationsreihenfolge das exakte Ergebnis nur auf 1 bzw. 2 Stellen genau erreicht. Dies macht den Einfluss deutlich, den die Summationsreihenfolge bei der Rechnung auf einem Computer besitzt. ■

Aufgabe

1.7 Weisen Sie durch Betrachtung von Rundungsfehler und Stellenzahl nach, dass in obigem Beispiel der Summenwert bei der Summation von 1 bis 300 bei 3- bzw. 5-stelliger Rechnung ab s_{14} bzw. s_{141} stagniert. Ab welchem Index stagniert der Summenwert bei n-stelliger Rechnung?

Man beachte, dass die verschiedenen Möglichkeiten der Berechnung der Summe in obigem Beispiel genau gleiche Gleitpunktoperationen benötigen, die Rechenzeit ist also stets die gleiche. Der einzige Unterschied besteht in der Reihenfolge der Operationen.

Als ein Maß für den Rechenaufwand kann man die Anzahl der durchgeführten Rechenschritte in der Gleitpunktarithmetik heranziehen, d. h. die Anzahl der Gleitpunktoperationen, im Englischen kurz „Flops“ („floating point operations“) genannt. Manchmal bezeichnet man auch eine Operation der Form $a+b\cdot c$, also eine Addition und eine Multiplikation zusammen, als einen Flop. Wir werden hier der Einfachheit

halber aber nicht alle Flops zählen, sondern nur die Punktrechenoperationen, also Multiplikationen und Divisionen. Als Maß für die Rechengeschwindigkeit eines Rechners ist die Einheit „flops per second“, also die Anzahl der möglichen Gleitpunktoperationen pro Sekunde, üblich. Der derzeit (Anfang 2021) weltweit schnellste Rechner („Fugaku“) steht beim RIKEN Center for Computational Science (Japan) und hat eine Leistung von 415.5 Petaflops, also mehr als $415 \cdot 10^{15}$ Operationen pro Sekunde, der schnellste Rechner in Deutschland („SuperMUC-NG“) steht im Leibniz-Rechenzentrum in München und belegt weltweit Platz 13 mit ca. 19.5 Petaflops[2].

Wir haben bisher nur den absoluten Fehler betrachtet. Dieser für sich allein sagt aber nicht viel aus – man kann z. B. die Qualität eines Messwertes nicht beurteilen, wenn man nur weiß, dass ein Widerstand R auf z. B. $\pm 2\,\Omega$ genau gemessen wurde. Zur Beurteilung muss man berücksichtigen, wie groß der Wert, den man messen möchte, wirklich ist. Man muss also den absoluten Fehler in Relation zur Größe der zu messenden Werte sehen, und dazu dient der relative Fehler:

Definition

Hat man eine Näherung $\tilde{x}$ zu einem exakten Wert $x \neq 0$, so bezeichnet $\left|\frac{\tilde{x}-x}{x}\right|$ den **relativen Fehler** dieser Näherung. ■

In der Literatur findet man oft auch $\tilde{x}$ im Nenner statt x. Der relative Fehler wird auch gern in % angegeben, d. h. statt von einem relativen Fehler von z. B. 0.15 redet man auch von 15 %.
Der maximal auftretende relative Fehler bei Rundung kann bei n-stelliger Gleitpunktarithmetik als

$$eps := \tfrac{B}{2} \cdot B^{-n}$$

angegeben werden. eps ist die kleinste positive Zahl, für die auf dem Rechner $1 + eps \neq 1$ gilt. Man bezeichnet eps auch als **Maschinengenauigkeit**. Es gilt dann:

$$\mathrm{rd}(x) = (1+\varepsilon)\,x \quad \text{mit } |\varepsilon| \leq eps.$$

Dies besagt, dass ε, also der relative Fehler der Näherung $\mathrm{rd}(x)$ an x, stets durch die Maschinengenauigkeit beschränkt ist.

[2] Eine aktuelle Liste der 500 schnellsten Rechner findet man unter *www.top500.org*

Aufgabe

1.8 Schreiben Sie ein kurzes Programm, das auf Ihrem Rechner näherungsweise die Maschinengenauigkeit eps berechnet. Schließen Sie aus dem Ergebnis, ob Ihr Rechner im Dual- oder Dezimalsystem rechnet und mit welcher Stellenzahl er operiert.

1.2 Auslöschung

Dieses Phänomen tritt bei der Subtraktion zweier fast gleich großer Zahlen auf (siehe auch Beispiel 7.2):

Beispiel 1.4
$\Delta_1 f(x,h) := f(x+h) - f(x)$ soll für $f = \sin$, $x = 1$ und $h = 10^{-i}$, $i = 1,\ldots,8$ mit 10-stelliger dezimaler Gleitpunktarithmetik berechnet werden und absoluter und relativer Fehler beobachtet werden.

Lösung: Man erhält

h	$\Delta_1 f(1,h)$	abs. Fehler	rel. Fehler
10^{-1}	$4.97363753 \cdot 10^{-2}$	$4.6461 \cdot 10^{-11}$	$9.3414 \cdot 10^{-10}$
10^{-2}	$5.36085980 \cdot 10^{-3}$	$1.1186 \cdot 10^{-11}$	$1.8875 \cdot 10^{-9}$
10^{-3}	$5.39881500 \cdot 10^{-4}$	$1.9639 \cdot 10^{-11}$	$3.6378 \cdot 10^{-8}$
10^{-4}	$5.40260000 \cdot 10^{-5}$	$2.3141 \cdot 10^{-11}$	$4.2834 \cdot 10^{-7}$
10^{-5}	$5.40300000 \cdot 10^{-6}$	$1.9014 \cdot 10^{-11}$	$3.5193 \cdot 10^{-6}$
10^{-6}	$5.40300000 \cdot 10^{-7}$	$1.8851 \cdot 10^{-12}$	$3.4890 \cdot 10^{-6}$
10^{-7}	$5.40000000 \cdot 10^{-8}$	$3.2263 \cdot 10^{-11}$	$5.5943 \cdot 10^{-4}$
10^{-8}	$5.40000000 \cdot 10^{-9}$	$3.2301 \cdot 10^{-12}$	$5.5950 \cdot 10^{-4}$

Hier sind verschiedene Phänomene zu beobachten:

- Der berechnete Wert hat immer weniger von Null verschiedene Ziffern. Grund: Wenn man zwei 10-stellige Zahlen voneinander subtrahiert, die annähernd gleich sind, fallen die gleichen Ziffern weg und nur die wenigen verschiedenen bleiben übrig. Mit fallendem h liegen die beiden Funktionswerte immer näher beieinander und daher wird die Anzahl der von Null verschiedenen Ziffern immer kleiner. Wird dagegen im IEEE-Standard gerechnet, also insb. im Dualsystem, so findet die Auslöschung bei der internen Rechnung in den Dualzahlen statt und ist für den Benutzer, der ja auf dem Bildschirm Dezimalzahlen sieht, nicht ohne Weiteres erkennbar.

- Der absolute Fehler ändert sich mit fallendem h kaum; er liegt etwas geringer als die theoretische Schranke $5 \cdot 10^{-10}$ erwarten ließe.
- Der relative Fehler steigt indes stark an. Dies war zu erwarten, denn der relative Fehler ist ja der Quotient aus dem absoluten Fehler dividiert durch den exakten Wert. Er muss hier ansteigen, denn der absolute Fehler bleibt in etwa gleich, während der exakte Wert fällt. ∎

Beispiel 1.5
Zur Lösung der quadratischen Gleichung $x^2 - 2\,p\,x + q = 0$ kann bekanntlich die Formel $x_{1,2} := p \pm \sqrt{p^2 - q}$ benutzt werden. Prüfen Sie, ob dabei Auslöschung auftreten kann und vergleichen Sie mit der Alternative $x_1 := p + \text{sign}(p) \cdot \sqrt{p^2 - q}$, $x_2 := \frac{q}{x_1}$.

Lösung: In der ersten Formel tritt Auslöschung auf, wenn eine der beiden Nullstellen nahe bei 0 liegt, d. h. wenn q klein gegenüber p^2 ist. In der Alternative werden Differenzen vermieden, x_1 wird ohne Differenzen berechnet, und x_2 aus x_1 (Satz von Vieta). ∎

Aufgabe

1.9 Versuchen Sie mit Ihrem Rechner den Grenzwert $\lim\limits_{x \to 0} \frac{e^x - 1}{x}$ (der 1 ist) näherungsweise zu berechnen, indem Sie immer kleinere Werte für x einsetzen. Erklären Sie Ihre Beobachtung.

1.3 Fehlerrechnung

Wie schon gesehen, wird beim Rechnen mit fehlerbehafteten Werten der Fehler weitergetragen. In den wenigsten Fällen verkleinert er sich dabei, in der Regel muss man mit einer Vergrößerung des Fehlers rechnen. Wir haben schon oben gesehen wie man in manchen Fällen durch Umstellen von Formeln Verbesserungen erzielen kann, jedoch an der Tatsache der Fehlerfortpflanzung an sich kann man wenig ändern. Es ist jedoch in der Praxis wichtig, wenn man schon die Fehler durch die endliche Rechnerarithmetik nicht vermeiden kann, wenigstens eine Vorstellung zu bekommen, wie groß denn der entstandene Fehler höchstens sein kann.

1.3.1 Fehlerfortpflanzung in arithmetischen Operationen

Gegeben seien zwei fehlerbehaftete Zahlen $\tilde{x}$, $\tilde{y}$ und zugehörige exakte Werte x, y. Bei der Addition sieht man dann aus

$$x + y - (\tilde{x} + \tilde{y}) = x - \tilde{x} + y - \tilde{y},$$

dass im günstigsten Fall, nämlich wenn die Vorzeichen von $x - \tilde{x}$ und $y - \tilde{y}$ entgegengesetzt sind, der Fehler der Summe kleiner sein kann als die Fehler der Summanden. Im Regelfall sind die Vorzeichen der Fehler aber nicht bekannt, sodass wir vom ungünstigen Fall ausgehen. Das bedeutet, dass sich die Fehler addieren. Da wir also das Vorzeichen außer Betracht lassen – daher haben wir den absoluten Fehler ja auch als Absolutbetrag des Fehlers definiert – gilt das Gleiche für die Subtraktion. Hierbei ist aber zusätzlich das in 1.2 besprochene Phänomen der Auslöschung zu beachten.
Im Falle der Multiplikation gilt:

$$x\,y - \tilde{x}\,\tilde{y} = x\,(y - \tilde{y}) + y\,(x - \tilde{x}) - (x - \tilde{x})\,(y - \tilde{y})$$

Insbesondere hat das Produkt von $\tilde{y}$ mit einer Maschinenzahl $x = \tilde{x}$ also den x-fachen absoluten Fehler von $\tilde{y}$. In obiger Formel ist das Produkt der beiden absoluten Fehler normalerweise klein gegenüber den anderen Größen. Bei der Multiplikation mit einer fehlerbehafteten Größe $\tilde{y}$ muss man sogar mit einem noch größeren absoluten Fehler des Produktes rechnen.
Für den relativen Fehler des Produktes gilt:

$$\frac{x\,y - \tilde{x}\,\tilde{y}}{x\,y} = \frac{x - \tilde{x}}{x} + \frac{y - \tilde{y}}{y} - \frac{x - \tilde{x}}{x} \cdot \frac{y - \tilde{y}}{y}.$$

Das Produkt der relativen Fehler von $\tilde{x}$ und $\tilde{y}$ ist in der Regel klein gegenüber den anderen Größen. Eine analoge Betrachtung für die Division führt auf ein ähnliches Ergebnis. Wir halten also fest:

- Bei der Addition und Subtraktion addieren sich die absoluten Fehler der Summanden in erster Näherung.
- Der absolute Fehler eines Produktes liegt in der Größenordnung des Produktes des größeren der beiden Faktoren mit dem größeren der beiden absoluten Fehler.
- Beim Multiplizieren addieren sich die relativen Fehler der Faktoren in erster Näherung.

■

Stichwortverzeichnis

A
Ableitung, partielle 112
Abschätzung
– a-posteriori- 22, 62
– a-priori- 22, 62
Abschneidefehler 108
Anfangswertproblem 138
Ansatzfunktion 94
Ausgleichsfunktion 94
Ausgleichsgerade 94
Ausgleichsproblem 93
– allgemeines 101
– lineares 96
Auslöschung 8, 109

B
Bisektion 18

C
Cholesky-Zerlegung 44

D
Determinante 38
Dezimalzahl 2
diagonaldominant 62
Differenzen, dividierte 76
Differenzenformel 107
Differenzialgleichung, gewöhnl. 138
direkte Verfahren 33
Diskretisierung 138
Diskretisierungsfehler 108
Dreieckszerlegung 41
Dualzahl 2

E
Einschrittverfahren 145
Einzelschrittverfahren 61
Euler-Verfahren 139
– modifiziertes 146
Extrapolation 113
– bei Anfangswertproblemen 151
– bei Quadratur 131

F
Fehler
– absoluter 4
– bei Rundung 4
– globaler 142
– lokaler 142
– relativer 7
Fehlerfortpflanzung 11
Fehlerfunktional 94
Fehlerordnung 108, 124
Fehlerquadrate, kleinste 94
Fehlerrechnung 9
Fixpunkt 19
– abstoßender 22
– anziehender 22
Fixpunktiteration 19, 20
Fixpunktsatz, Banachscher 22
Flop (floating point operation) 6

G
Gauß-Algorithmus 34, 37, 40
Gauß-Formeln 129
Gauß-Newton-Verfahren 102
Gauß-Seidel-Verfahren 61
Gesamtschrittverfahren 59
Gitterpunkte 139
Gleitpunkt
– -arithmetik 5
– -operation 6
– -zahl 2

H
Horner-Schema 30
Householder-Matrix 47

I
IEEE-Format 2, 3
Implizite Verfahren 151
Interpolationsfehler 80
Interpolationspolynom 74
– Lagrangesches 75
– Newtonsches 77
Interpolationsproblem 73
Interpolierende 73

J
Jacobi-Matrix 69
Jacobi-Verfahren 59

K
Konditionszahl 13, 55
Konsistenzordnung 142
kontraktiv 22
Konvergenzgeschwindigkeit 28
Konvergenzordnung 28, 142

L
Laplace-Operator 113
Legendre-Polynom 130
Linearisierung 24, 140
Lipschitzbedingung 143
LR-Zerlegung 42

M
Mantisse 2
Maschinengenauigkeit 7
Maschinenzahl 3
Matrix, orthogonale 46
Mehrschrittverfahren 152
Mittelpunktsregel 121, 126, 146
– summierte 122, 127
Momente 86

N
Neville-Aitken-Schema 79
Newton-Cotes-Formeln 128
Newton-Verfahren 24, 69
– vereinfachtes 26
Newton-Verfahren für Systeme 69
– vereinfachtes 71
Norm 53, 54
Normalgleichungen 97

O
$O(h^k)$ 108
orthogonal 46

P
Polynomdivision 31
positiv definit 44
Punktoperation 7

Q
QR-Zerlegung 46, 47, 97
Quadratmittelproblem 103
Quadratur, adaptive 135
Quadraturfehler 124
Quadraturformel, interpolat. 123
Quadraturverfahren 120

R
Rückwärtseinsetzen 34
Rechteckregel 121
– summierte 122, 127
rechts-obere Dreiecksmatrix 34
Regressionsgerade 94
regula falsi 27
Restglied, Taylorsches 108
Richtungsfeld 138
Romberg-Extrapolation 132

Rundungsfehler 4
Runge-Kutta-Verfahren
– allgemeines 149
– klassisches 148

S
Satz von Taylor 107
Schrittweite 139
Schrittweitensteuerung 144
Sekantenverfahren 27
Simpson-Regel 123, 126
– summierte 127
Spaltenpivotisierung 39
Spaltensummenkriterium 64
Spektralradius 54
Spline
– interpolierender 85
– kubischer 85
– natürlicher 85
– periodischer 85
– vollständiger 85
Splinefunktion 85
Splineinterpolation 84
Stützstellen 73

T
Trapezregel 121, 126
– summierte 122, 127

V
Verfahren von Heun 147

Z
Zeilensummenkriterium 62
Zwischenwertsatz 17