Kapitel 1

Einleitung

Zunächst stellt sich natürlich die Frage: Was ist eigentlich *numerische* Mathematik. In der Numerik beschäftigt man sich mit der Entwicklung und Analyse von Lösungsverfahren für mathematische Problemstellungen, bei denen *numerische* Werte (reelle oder komplexe Zahlen) konkret berechnet werden sollen, sowie mit der Umsetzung dieser Verfahren auf Computern.

Die Ziele dieser Vorlesung sind,

- die wichtigsten numerischen Verfahren für wesentliche Grundaufgaben der Mathematik kennenzulernen,
- die Möglichkeiten und Grenzen numerischer Verfahren abschätzen zu können,
- die Unterschiede zur analytischen Lösung zu verstehen, und
- Fehlerquellen erkennen und vermeiden zu können.

Gerade das Wissen um die Grenzen und Fehlerquellen numerischer Verfahren ist unerläßlich, um die Qualität und Zuverlässigkeit der berechneten Ergebnisse einschätzen zu können.

Man kann natürlich die Frage stellen, wozu wir überhaupt numerische Verfahren benötigen. Dies wird im Verlauf der Vorlesung anhand einiger Beispiele verdeutlicht. Kurz zusammengefaßt kann man aber festhalten, daß viele mathematische, naturwissenschaftliche, technische oder auch ökonomische Aufgabenstellungen nicht in geschlossener Form analytisch gelöst werden können, so daß oft nur eine approximative Lösung möglich ist. Andererseits erfordert aber auch eine analytische (symbolische) Lösung numerische Rechnung, wenn für gegebene Daten eine konkrete Lösung ausgerechnet werden muß. Dazu muß die analytische Lösung dann wiederum in einen numerischen Algorithmus umgesetzt werden. In vielen Fällen ist auch die Komplexität der Auswertung einer analytischen Lösung zu hoch, so daß man sich aus Zeit- oder Speicherbedarfsgründen, d.h. Kostengründen, mit einem approximativen Lösungsverfahren begnügen muß.

Einige der ersten numerischen Verfahren, die auch heute noch die Grundlage vieler moderner Algorithmen liefern, wurden Anfang des 19. Jahrhunderts, motiviert aus der

Landvermessung (Kartierung) und der Astronomie (Berechnung der Planetenbahnen), entwickelt. Es mußten erstmals lineare Gleichungssysteme mit "vielen" Unbekannten (das hieß damals, 5–20 Unbekannte, heute löst man Gleichungssysteme mit 10^9 Unbekannten!) gelöst werden. Dazu entwickelten Gauß[1] und Jacobi[2] erste Berechnungsverfahren. Die moderne Numerik entwickelte sich rasant mit dem Aufkommen der Computer in den 50er und 60er Jahren des letzten Jahrhunderts.

Grob gesprochen, unterteilt sich die Numerische Mathematik in die folgenden beiden Teilbereiche:

- Approximation mathematischer Größen, deren exakte Berechnung nicht möglich oder zu aufwendig ist, z.B.
  - Funktionsapproximation,
  - Lösung von Differentialgleichungen.
- Entwicklung und Analyse numerischer Algorithmen; Untersuchung der durch die Umsetzung auf dem Computer hervorgerufenen Effekte.

Dabei bedient sich die Numerische Mathematik in vielen anderen mathematischen Teildisziplinen, insbesondere sind hier zu nennen:

- Lineare Algebra,
- reelle und komplexe Analysis,
- Theorie der (gewöhnlichen und partiellen) Differentialgleichungen,
- Funktionalanalysis,

aber auch einzelne Aspekte vieler anderer Bereiche, wie z.B. Statistik, Graphentheorie, Differentialgeometrie spielen oft eine wichtige Rolle.

Die Aufgaben der Numerik bei der Computersimulation realer Prozesse wie in Abbildung 1.1 dargestellt liegt vornehmlich in den Bereichen der Analyse des mathematisierten Problems, um ein geeignetes Verfahren auszuwählen oder zu entwickeln, entsprechende mathematische Software zur Verfügung zu stellen, die Computersimulation durchzuführen und die Güte der berechneten Resultate abzuschätzen. Dies erfordert gerade an den Schnittstellen in hohem Maße die Interaktion mit Anwendern (Ingenieuren, Naturwissenschaftlern,...). Die Simulationsergebnisse selbst führen dann oft zu einer Änderung/Verbesserung des Modells, wodurch der gesamte Prozeß erneut durchlaufen werden muß, ggf. auch unter Änderung/Anpassung des numerischen Verfahrens.

Wenn möglich, sollte man mathematische Standardsoftware verwenden, da die Entwicklung von Verfahren und Software oft sehr zeitaufwendig und fehleranfällig ist. Dabei

[1] Carl Friedrich Gauß (1777–1855), deutscher Mathematiker
[2] Carl Gustav Jacob Jacobi (1804-1851), deutscher Mathematiker

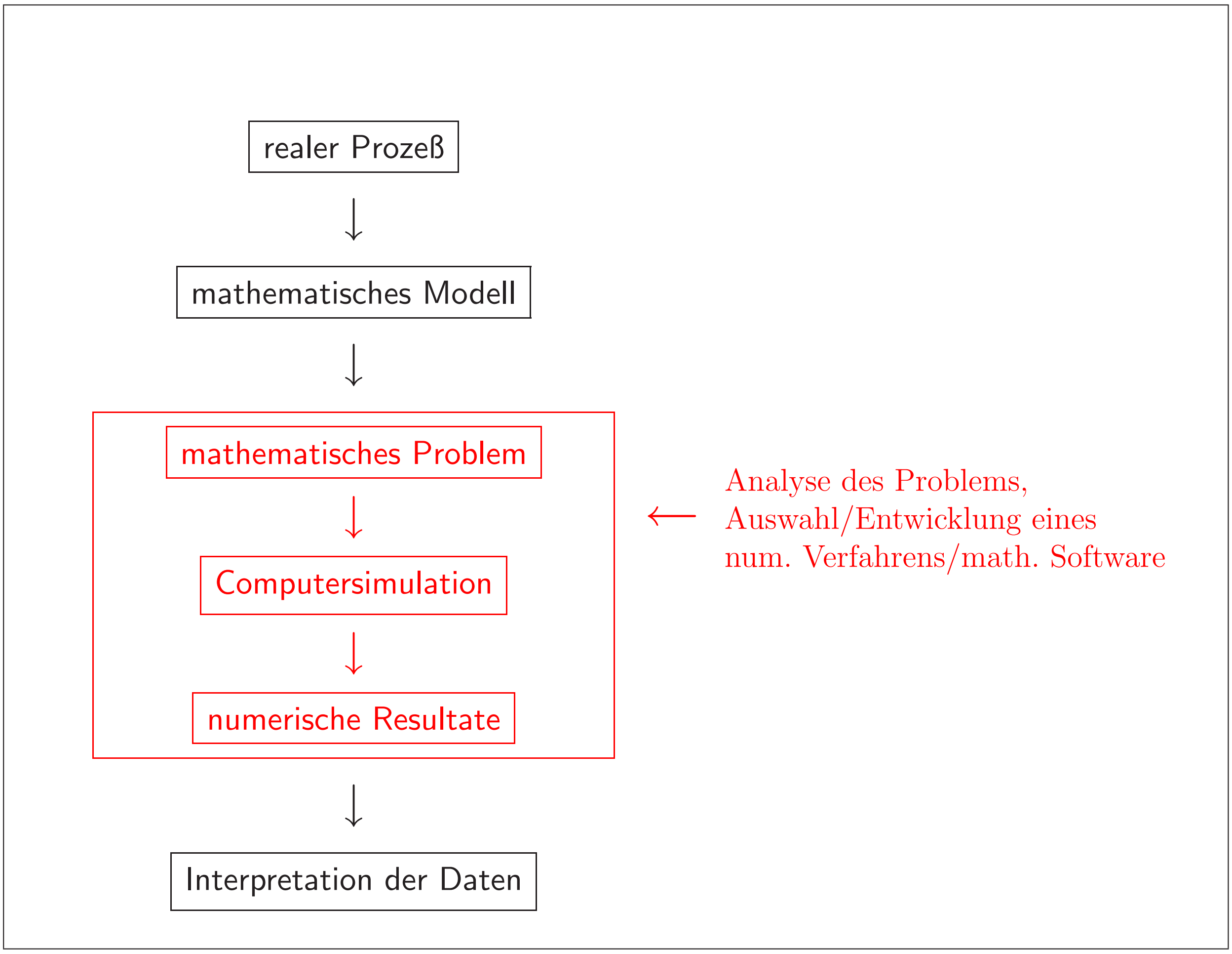

Abbildung 1.1: Aufgaben der Numerik bei der Simulation.

ist jedoch ein reflektierter Umgang mit der Software entscheidend für den Erfolg der Computersimulation, wie man z.B. beim Elch-Test für die Mercedes A-Klasse, der Explosion der ersten Ariane V Rakete, oder dem Sinken der Sleipner Ölplattform gesehen hat[3]. Bei mathematischer Software unterscheidet man grob zwischen numerischer und symbolischer Software (siehe Abbildung 1.2), wobei aber in den letzten Jahren die Grenzen verschwimmen, da z.B. MATLAB inzwischen symbolische Rechenfunktionen beinhaltet, Mathematica ohnehin ein "Zwitter" ist, und auch die anderen Formelmanipulationspakete wie Maple numerische Rechnungen durchführen können.

[3]Beschreibungen dieser und anderer Beispiele für Software- und Berechnungsfehler mit fatalen Folgen findet man z.B. unter `http://www5.in.tum.de/~huckle/bugse.html`

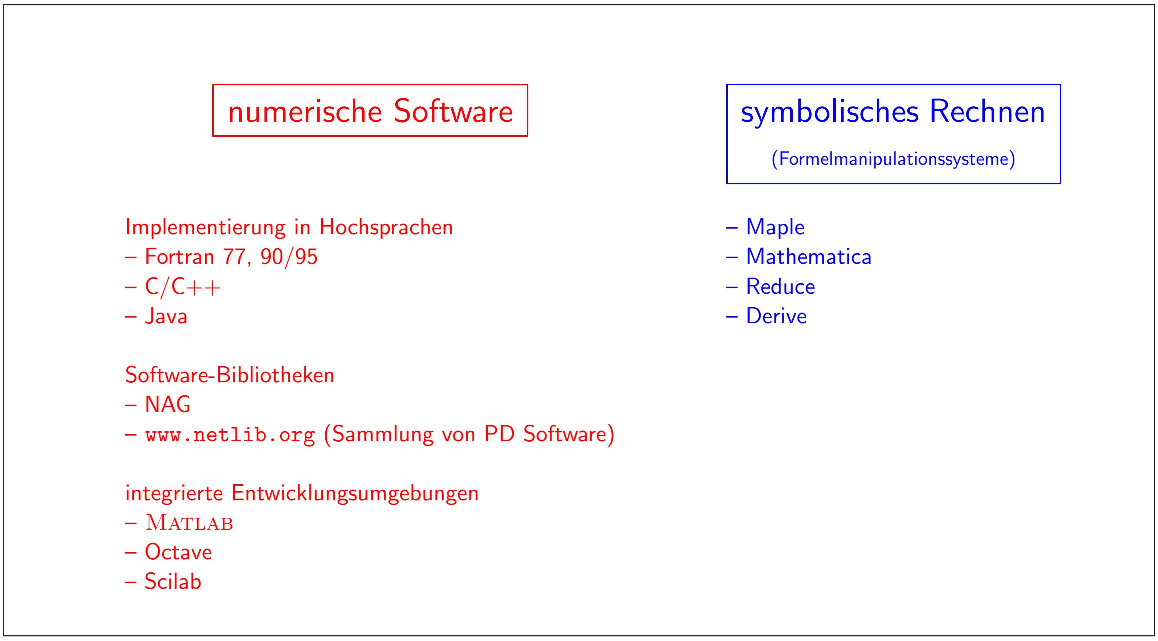

Abbildung 1.2: Mathematische Software.

Die folgenden drei Beispiele demonstrieren einige der Schwierigkeiten bei der Lösung mathematischer Aufgaben mit numerischer Rechnung.

Beispiel 1.1 *Dieses Beispiel demonstriert, daß die Lösung von linearen Gleichungssystemen der Form*

$$Ax = b, \qquad A \in \mathbb{R}^{n\times n} \textit{ invertierbar}, \; b \in \mathbb{R}^n,$$

mit Hilfe der Gauß-Elimination numerisch nicht immer befriedigende Lösungen liefert. In MATLAB *wird ein solches Gleichungssystem für eine invertierbare Matrix A durch Verwendung des Befehls*

```
x = A\b
```

mit Gauß-Elimination gelöst.

Als Koeffizientenmatrix verwenden wir hier die Hilbertmatrix $A = [a_{ij}]$ mit

$$a_{ij} = \frac{1}{i+j-1}, \qquad i, j = 1, \dots, n.$$

Die rechte Seite b wird so gewählt, daß die exakte Lösung jeweils der Vektor ist, der nur aus Einsen besteht. Für wachsendes n wird das Gleichungssystem zunehmend schlechter *gelöst, z.B. erhält man für* $n = 12$

$$\tilde{x} = \begin{bmatrix} 1.0000 \\ 1.0000 \\ 0.9999 \\ 1.0013 \\ 0.9907 \\ 1.0408 \\ 0.8867 \\ 1.2046 \\ 0.7605 \\ 1.1752 \\ 0.9272 \\ 1.0131 \end{bmatrix},$$

wobei $\tilde{x}$ *die numerisch berechnete Lösung ist. Man sieht also, daß die berechneten Werte bereits für kleine Werte von* n *kaum noch etwas mit den exakten Werten zu tun haben.*

Beispiel 1.2 *Die Eigenwerte* λ *einer Matrix* $A \in \mathbb{C}^{n\times n}$ *erfüllen die Gleichung*

$$Ax = \lambda x$$

für ein $x \in \mathbb{C}^n \setminus \{0\}$*. Aus der linearen Algebra wissen wir, daß die Eigenwerte gerade die Nullstellen des charakteristischen Polynoms von* A*, definiert durch*

$$p_A(\lambda) = \det(A - \lambda I_n),$$

sind. Das folgende Beispiel zeigt, daß das charakteristische Polynom aber nicht zur numerischen Berechnung der Eigenwerte taugt. Dazu wird eine diagonale Matrix A *mit Eigenwerten* $\{1, 2, \ldots, n\}$ *für verschiedene Werte von* n *erzeugt. Mit*

```
>> p =  poly(A)
```

erhält man in MATLAB *das charakteristische Polynom von* A *und mit*

```
>> roots(p)
```

dessen Nullstellen, also die berechneten Eigenwerte von A*. Die exakten und berechneten Eigenwerte für* $n = 20$ *sowie* $n = 25$ *sind in Abbildung 1.3 grafisch dargestellt. Man sieht, daß für* $n = 25$ *völlig unsinnige Werte ausgerechnet werden.*

Beispiel 1.3 *Bei diesem Beispiel handelt es sich um die analytische Berechnung des Integrals*

$$I(n) = \int_0^1 x^n e^{x-1}\, dx.$$

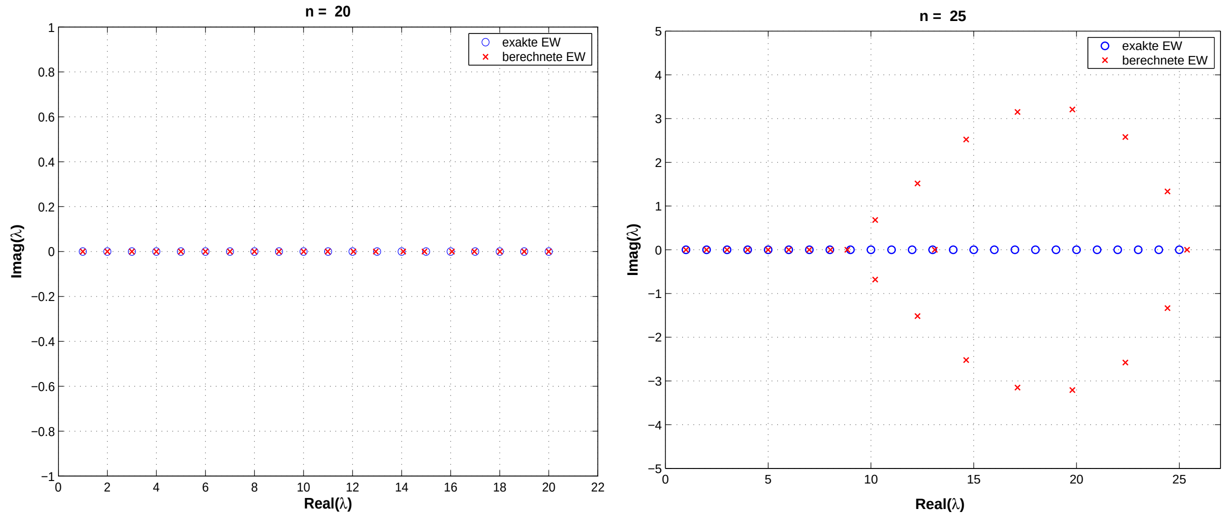

Abbildung 1.3: Exakte und mit Hilfe des charakteristischen Polynoms berechnete Eigenwerte von $A = \text{diag}(1, \dots, n)$.

Man zeigt leicht, daß $0 < I(n+1) < I(n) < 1$ *für alle* n *gilt. Abbildung 1.4 zeigt den Wert des Integrals als Flächeninhalt für* $n = 3$.

Mit Hilfe partieller Integration ergibt sich für die Berechnung der Werte von $I(n)$ *eine einfache Rekursionsformel:*

$$I(1) = \frac{1}{e}, \quad I(n) = 1 - nI(n-1) \text{ für } n > 1.$$

Die mit MATLAB *berechneten Ergebnisse für verschiedene Werte von* n *sind in folgender Tabelle gegeben:*

n	$I(n)$
1	0.3679
2	0.2642
3	0.2073
5	0.1455
10	0.0839
15	0.0590
16	0.0566
17	0.0374
18	0.3259
19	-5.1930
20	104.8608

Aufgrund der theoretischen Eigenschaften von $I(n)$ *ist sofort klar, daß die numerisch berechneten Werte für* $n > 17$ *falsch sein müssen.*

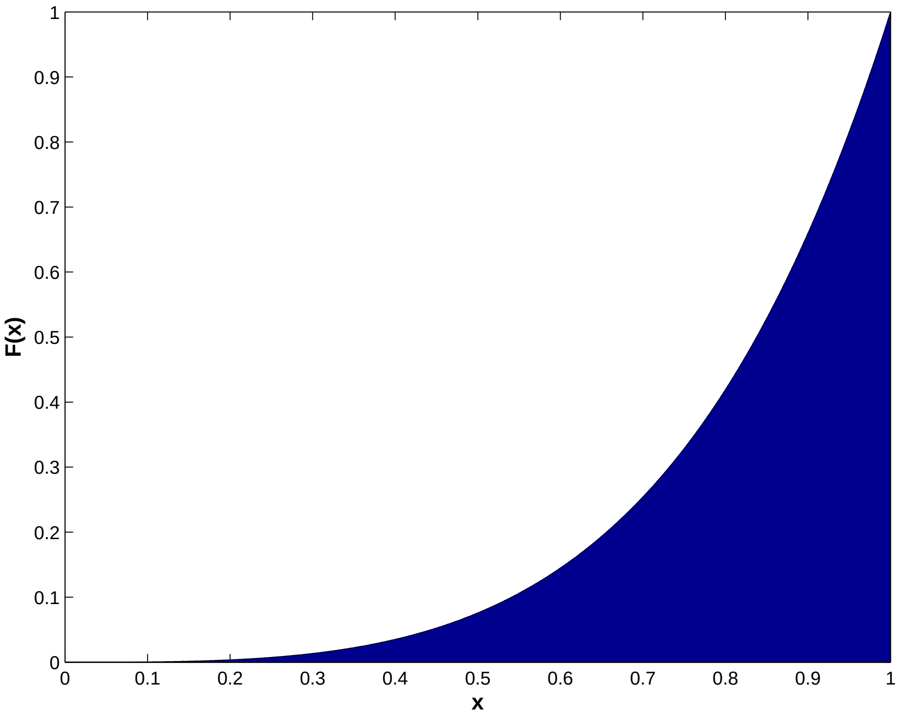

Abbildung 1.4: Darstellung von $I(3)$ als Flächeninhalt mit Hilfe des Graphen von $F(x) = x^n e^{x-1}$.

Ein solches Verhalten kann man bei der numerischen Umsetzung von Rekursionsformeln oft beobachten. Das Beispiel zeigt, daß man bei der Verwendung auch einfachster Rekursionen sehr vorsichtig sein muß.

Um die in den vorangegangenen Beispielen beobachteten "merkwürdigen" Ergebnisse verstehen und analysieren zu können, werden wir im folgenden Kapitel die Behandlung von reellen Zahlen und Rechenoperationen auf Computern näher untersuchen.