

**Pierre Colson**

## **Einsatz von Scilab in der Analysis für Ingenieure**

### **1. Einführung**

Der Einsatz eines Softwaresystems in einer Analysis-Vorlesung für Ingenieure ist an sich keine Neuigkeit. Er wird aber zu einer besonderen Herausforderung, wenn man sich im Rahmen einer Service-Veranstaltung an sehr viele Studenten mit einer inhomogenen und häufig ungenügenden Vorbildung in Mathematik wendet.

In diesem Beitrag beschreiben wir ein passendes Konzept für den Einsatz von Scilab (ein frei erhältliches Softwaresystem, das im Aufbau, in der Bedienung und in der Leistung mit MATLAB vergleichbar ist) im Rahmen der Lehrveranstaltungen der Analysis im ersten Semester der Ingenieurstudiengänge an der Technischen Universität Berlin.

### **2. Die Software Scilab**

Die Software Scilab wird seit Anfang der 90er Jahre an verschiedenen Forschungseinrichtungen in Frankreich, zuletzt am INRIA (Institut National de Recherche en Informatique et en Automatique), weiterentwickelt. Auf der Scilab-Webseite [4] ist sowohl die eigentliche Software zu finden, als auch eine Online-Hilfe zu allen Scilab-Kommandos und viel Dokumentation. Dem interessierten Leser empfehlen wir die Benutzereinführungen von A. Stoffel [5] und J.-M. Zogg [6].

Studenten können die frei erhältliche Software Scilab auf ihrem eigenen Rechner installieren und Programmieraufgaben nicht nur in den Computerräumen der Universität bearbeiten, sondern auch zu Hause. Aufgrund der Ähnlichkeit ist ein späterer Umstieg von Scilab auf MATLAB (z.B. im Rahmen von Projektarbeiten oder der Diplomarbeit) problemlos zu bewältigen.

Der Schwerpunkt unserer Scilab-Einführung für Studienanfänger ist es nicht, möglichst effiziente Algorithmen zu beschreiben, oder die volle Leistung der Scilab-Umgebung (wie z.B. komponentenweise Operationen) zu nutzen. Der fortgeschrittene Umgang mit Ingenieursoftware bleibt Gegenstand von weiterführenden Lehrveranstaltungen auf dem Gebiet der Numerik. An der TU Berlin findet z.B. das „Projekt Praktische Mathematik“ statt, in dem kleinere

Gruppen von Studenten ein Semester lang an ingenieurwissenschaftlichen Aufgaben arbeiten und in FORTRAN, C oder Matlab programmieren.

### 3. Einsatz von Scilab an der TU Berlin

An den Mathematikveranstaltungen für Ingenieure der TU Berlin nehmen weit über 2000 Studienanfänger teil. Dies ist insbesondere der Fall bei der Vorlesung „Analysis 1 für Ingenieure“, die im üblichen Umfang (4 SWS Vorlesung und 2 SWS Tutorium) als Service-Veranstaltung von dem Institut für Mathematik angeboten wird.

Die Lehrinhalte und Lernziele dieser Vorlesung beruhen auf einem langjährigen Kompromiss zwischen der Fakultät für Mathematik und den Service-nehmenden Instituten. Diese Institute plädieren andererseits für eine Einführung in eine ingenieurrelevante Software ab dem ersten Semester. Dies umzusetzen ist aber schwierig: der Umgang mit einer Programmiersprache und die Interpretation von Computer-Lösungen setzen genau die solide mathematische Vorbildung voraus, die vielen Studienanfängern fehlt [1], [2], [3]. Des Weiteren scheitert der Einsatz von neuem Stoff oft an der zeitlichen Überlastung der Studenten in den meisten Ingenieurstudiengängen.

Aus diesen Gründen soll die Scilab-Einführung für die Studenten lediglich ein erster Einblick in eine Programmiersprache und in numerische Methoden darstellen, sowie eine Illustration der mathematischen Ergebnisse aus der Vorlesung. Vor allem soll sie aber ein Anreiz sein, sich intensiver mit der eigentlichen Mathematik auseinanderzusetzen.

Die Motivation und Einführung der grundlegenden Scilab-Kommandos kann durchaus ohne grossen zeitlichen Aufwand in der Analysisvorlesung erfolgen. Da wird auch auf mögliche Trugschlüsse und Fehlerquellen hingewiesen. Pro Woche bearbeiten dann die Studenten eine Scilab-Hausaufgabe, wobei die Erfahrung zeigt, dass sie sich spontan und sehr schnell mit der Scilab-Umgebung (Editor, Online-Hilfe, usw.) vertraut machen.

Zur Bearbeitung der Hausaufgaben wird den Studenten eine Vorlage zur Verfügung gestellt. Diese Vorgabe enthält eine gewisse Struktur, aus der hervorgeht, was und an welcher Stelle zu tun ist. Als Test für die entstandenen Scilab-Programme sind Grafiken oder Tabellen zu erzeugen und es sind schließlich diese Ausgaben, die von den Tutoren korrigiert und benotet werden (s. Abbildung 1).

Abbildung 1: Vorlage, Ausgabetabelle und Lösung einer Programmieraufgabe zur Heron-Formel. Anhand dieses elementaren Programms kann auf Rundungsfehler und numerisch instabile Formeln eingegangen werden.

```

DIE FORMEL VON HERON FUER DEN FLAECHEINHALT EINES DREIECKS
Berechnung der Flaechе mit der Formel von Heron
Daten: a,b,c  Seitenlaengen des Dreiecks
        F      Flaechеinhalte des Dreiecks
function [F]=heron(a,b,c)
endfunction

// Berechnung des Fehlers
function [fehler]=herontest(eps)
a=1
b=eps
c=0.5*eps
// Echte Flaechе des Dreiecks
F=0.5*a*b
// Flaechе des Dreiecks nach Heron
Fheron=heron(a,b,c)
// Fehler in Prozent
fehler=100*abs((Fheron/F)-1)
endfunction

mit eps=10^(-n)
n      eps      Fehler in %
===      =====
1      1.0e-01      0.0
2      1.0e-02      0.0
3      1.0e-03      0.0
4      1.0e-04      0.0
5      1.0e-05      0.0
6      1.0e-06      0.0
7      1.0e-07      0.0
8      1.0e-08      0.0
9      1.0e-09      0.0
10     1.0e-10      0.0
11     1.0e-11      0.0
12     1.0e-12      0.0
13     1.0e-13      0.1
14     1.0e-14      2.3
15     1.0e-15      11.2
16     1.0e-16      100.0
17     1.0e-17      100.0
18     1.0e-18      100.0
19     1.0e-19      100.0
20     1.0e-20      100.0

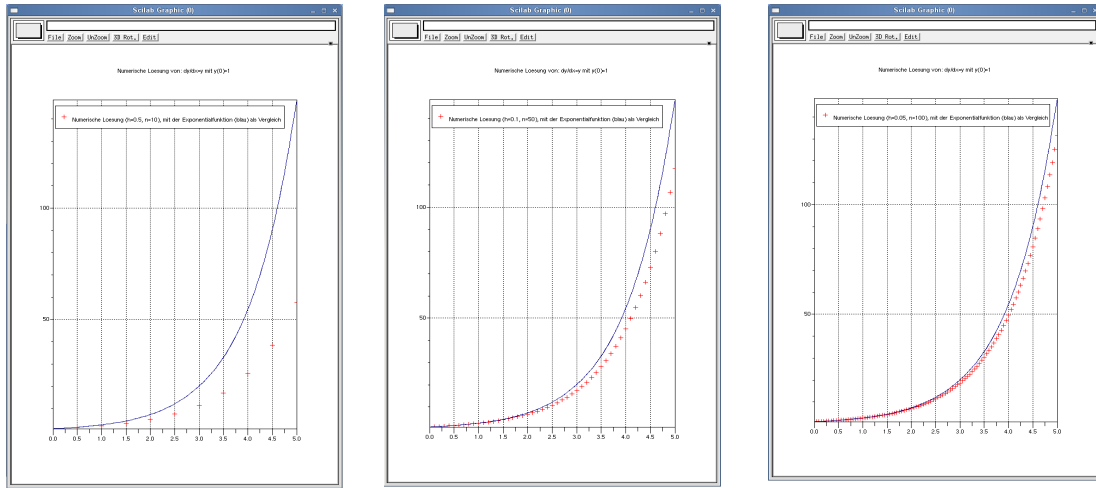
```

#### 4. Material am Beispiel der Exponentialfunktion

Die Exponentialfunktion ist in der Vorlesung „Analysis 1 für Ingenieure“ eine der wichtigen elementaren Funktionen, auf die immer wieder zurückgegriffen wird. In den ersten Wochen werden die Eigenschaften dieser Funktion als Schulstoff vorausgesetzt, dann wird sie als Lösung einer Differentialgleichung definiert, und gegen Ende des Semesters als Summe einer Reihe. Jedesmal werden die vertrauten Eigenschaften (Additionsformel, Ableitung, usw.) abgeleitet und die Äquivalenz der verschiedenen Zugänge bewiesen.

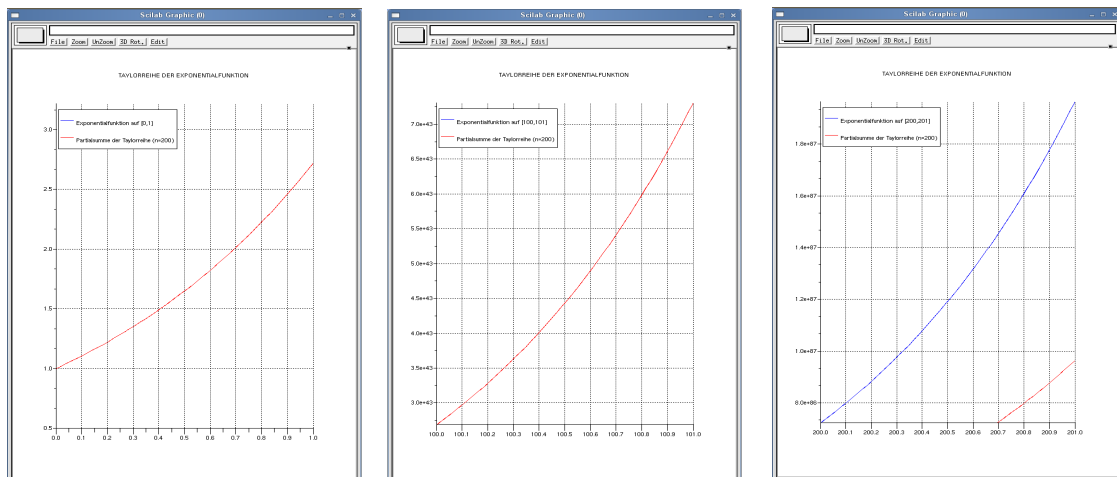
Die Definition der Exponentialfunktion als Lösung der Differentialgleichung  $y'=y$  mit  $y(0)=1$  ist für viele Studenten eine recht abstrakte. Sie bietet aber die Möglichkeit, ein Diskretisierungsverfahren an einem elementaren Beispiel zu implementieren. Als Test für ihr Programm sollen die Studenten die Parameter (Anzahl der Stützstellen und Schrittweite) variieren (s. Abbildung 2) und die entstandenen Grafiken interpretieren.

Abbildung 2. Ausgabegrafiken bei der Diskretisierung der Differentialgleichung  $y'=y$  mit  $y(0)=1$  (Schrittweite  $h=0,5; 0,1; 0,05$  bei  $n=10, 50, 100$  Stützstellen).



Nach der Einführung von Reihen und Funktionenreihen wird bewiesen, dass die Taylorreihe der Exponentialfunktion um 0 überall konvergiert. Es wird dann darauf hingewiesen, dass diese Reihe umgekehrt als Definition der Exponentialfunktion herangezogen werden kann. Zu dieser Definition schlagen wir als Scilab-Aufgabe vor, die Partialsummen der Reihe mit der Exponentialfunktion auf bestimmten Intervallen zu vergleichen (s. Abbildung 3). Das Scilab-Programm ist so zu schreiben, dass eine sehr grosse Anzahl von Summanden berechnet werden kann: die vorprogrammierte Fakultät kann man nicht verwenden und es muss eine rekursive Definition der Reihenglieder benutzt werden.

Abbildung 3. Ausgabegrafiken zum Vergleich der Exponentialfunktion mit der abgebrochenen Exponentialreihe (bei  $n=200$ ) auf den Intervallen  $[0,1]$ ,  $[100,101]$  und  $[200,201]$ .



## Literatur

1. Alpers, B., Discussion on integrating technology into the mathematical education of engineers, in: Demlova, M., Lawson, D. (Eds.): Proceedings of the 13<sup>th</sup> European Seminar on Mathematics in Engineering Education, Kongsberg 2006, 65-67.
2. Schott, D., Challenges in Computer Mathematics in Engineering Education, Global J. of Engng. Educ., Vol. 9, No. 1, 2005.
3. Schott, D., Fluch und Segen der Computermathematik, Global J. of Engng. Educ., Vol. 8, No. 3, 2004.
4. Scilab-Homepage: <http://www.scilab.org> (letzter Zugang am 10.11.08)
5. Stoffel, A., Kurzeinführung in Scilab, als .pdf Datei erhältlich unter: <http://alex.nt.fh-koeln.de/mapdf/scilabein.pdf> (letzter Zugang am 10.11.08)
6. Zogg, J.-M., Arbeiten mit Scilab und Scicos, als .pdf Datei erhältlich unter: [http://www.fh-htwchur.ch/uploads/media/Arbeiten\\_mit\\_Scilab\\_und\\_Scicos\\_v1.pdf](http://www.fh-htwchur.ch/uploads/media/Arbeiten_mit_Scilab_und_Scicos_v1.pdf) (letzter Zugang am 10.11.08)

## Autor

**Dr. Pierre Colson**

Technische Universität Berlin

Straße des 17. Juni 136

D-10963 Berlin

E-Mail: [colson@math.tu-berlin.de](mailto:colson@math.tu-berlin.de)