

# Redesigning the Calculus Sequence for Engineering Students

Pierre Colson\* and Katherine Roegner\*

\* Institut für Mathematik, Technische Universität Berlin, Str. des 17. Juni 136, 10623 Berlin, Germany.

## Abstract

Modeling, programming and computing are core activities of modern day engineering and require a certain fluency with mathematics. In striking contrast, many engineering students lack familiarity with elementary mathematical notions and their difficulties only accumulate in mathematical disciplines. This article presents some efforts to deal with this situation at the Technische Universität Berlin (TU Berlin), where the project “Innovations in Mathematics Education for the Engineering Sciences” is currently in the initial phase of developing and testing material for the calculus sequence. This project is embedded in the Matheon Research Center and funded by the German Research Foundation, with the overall aim to completely redesign the first-year calculus education for engineering students at the TU Berlin. A first emphasis of the project is the development of applets containing background information on and visualizations of key notions, as well as electronically corrected exercises. A second emphasis is the early introduction to computer software used by practicing engineers.

## Background

All engineering students at the TU Berlin (approximately 4000 first-year students from over 20 different fields) attend a two-semester calculus course and a one-semester linear algebra course. Both courses are taught by members of the mathematics department and the syllabuses, as well as the learning goals and grading issues, rely on a consensus reached with the engineering departments. This curriculum acknowledges the need for modern day engineers to possess a sound understanding of mathematical ideas and techniques, much in accordance with different studies (see Nguyen (1998) and Mustoe (2002)).

On the other hand, feedback from lectures and tutorials confirms that our incoming students have very heterogeneous mathematics ability and background. Since there are no entrance examinations, it remains difficult to obtain precise data on this important point. The rapid pace imposed upon the lecture (due to the amount of material to be taught) and the impossibility of individual follow-up in tutorials are clearly a disadvantage for students with a weaker background in mathematics, for whom difficulties often keep accumulating. Even when performing well in engineering disciplines, they may fail in mathematics and thus seriously compromise their chance of obtaining a degree.

## Aim of the Project

A primary goal of the project is to develop new and efficient material for the first-year calculus sequence so that students reach a deeper understanding of mathematical concepts. This should happen without lowering standards nor changing the emphasis of existing courses.

Attending lectures and solving tutorial problems are presently the main possibilities for our students to become familiar with calculus notions. It is therefore appropriate to introduce complementary

material the students can refer to at any given time and without supervision. At this stage, applets are the ideal medium, since they embody the possibility of visualization, coupled with interactivity and immediate feedback. To intensify student engagement and promote reflective learning, we shall further introduce our students to computer software used by practicing engineers. This should give students the possibility of generating their own visualizations of calculus notions and seeing calculus notions in a computational context.

This is of course not the first effort in this direction, and we refer the reader to Edwards (2003), Schott (2005), Tiedt (2001), Wlodkowski (2006), where interesting accounts of implementing graphical applets or computing software can be found.

## Graphical Applets

Each applet contains a theory part explaining in a concise and self-contained way the notion at hand, using the notations introduced in the lecture and providing a memorization help for the student. A demonstration part contains a visualization of the notion at hand, requiring the student to type in a limited amount of data. The emphasis is on the graphical illustration of the notion to make it meaningful and lively. In the training part, the student is expected to solve a problem centered on the mathematical point and enter the solution data. The solution is automatically corrected and an immediate feedback is provided. If the student solution is found to be incorrect, a step by step check of the solution process is proposed.

A typical example would be the graphical applet devoted to the Taylor formula. The Taylor approximation is not only at the core of the calculus sequence but the very justification of discretization procedures widely used in engineering sciences. The theory part introduces the constituents of the Taylor formula: the function we want to approximate at a given point, the Taylor polynomial formed with the successive derivatives of the function at this point, and the remainder, which assesses the approximation.

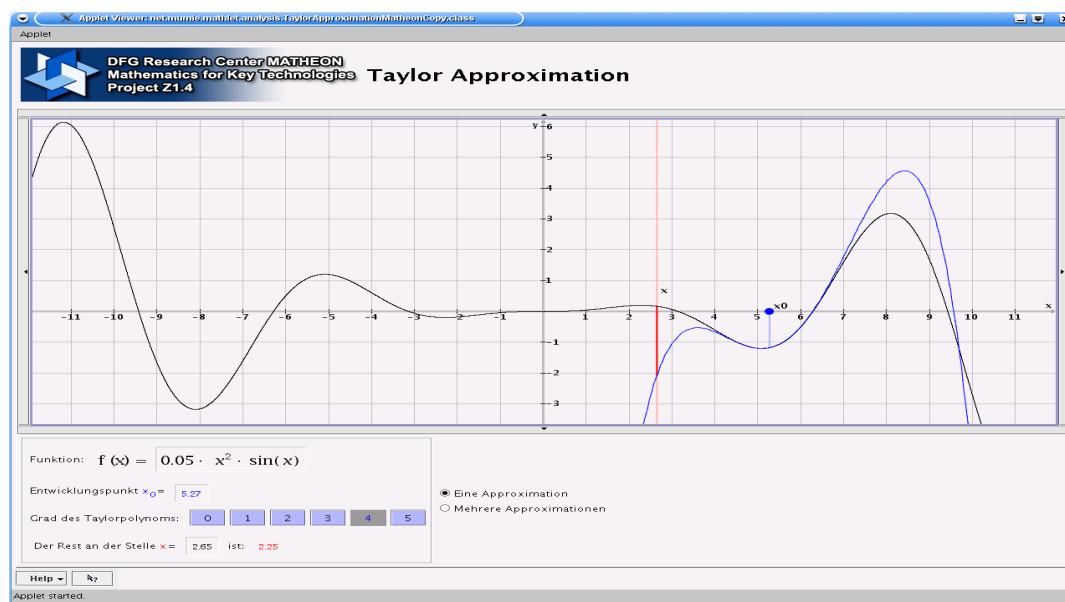


Figure 1: A Java applet on the Taylor expansion (demonstration part).

The demonstration part (Figure 1) is designed in such a way that the student enters a function (using the syntax familiar from pocket calculators), the expansion point and the degree of the Taylor polynomial. The data is processed and the applet displays a visualization of the function in the vicinity of the expansion point, as well as of the required Taylor polynomial. It is moreover possible to compute the remainder term at any given point or to display successive Taylor expansions at the same point.

The main point of the applet is to show that the Taylor expansion only holds locally and that the approximation at a given point improves with the degree of the Taylor polynomial. In the tutorial part, the student is given a function and an expansion point and is prompted to compute the corresponding first Taylor coefficients. The student solution is then displayed and graphically compared with the actual Taylor polynomial.

## Programming Exercises

Two reasons dictated our choice of the computer software Scilab: it is designed much like Matlab, making it easy for the students to switch later on to the software used by most practicing engineers; Scilab is free, making it possible for the students to download the software on their own computer and to work on the programming exercises at home rather than in a crowded university computer facility.

We do not expect programming knowledge from our students but rather introduce a small set of carefully chosen programming concepts during the first weeks of the lecture term. Bearing in mind that most of the students will be introduced to numerical methods in subsequent courses, we keep the knowledge of numerical processes at a very elementary level.

An important point is that computational software makes it possible to provide numerical evidence for many results in the calculus course. We believe this to be a satisfying approach for engineering students when many calculus results explained in the lecture are motivated but left without mathematical proof. Of course we should warn the students that numerical evidence merely provides an insight, does not constitute a proof in the mathematical sense and can even be misleading.

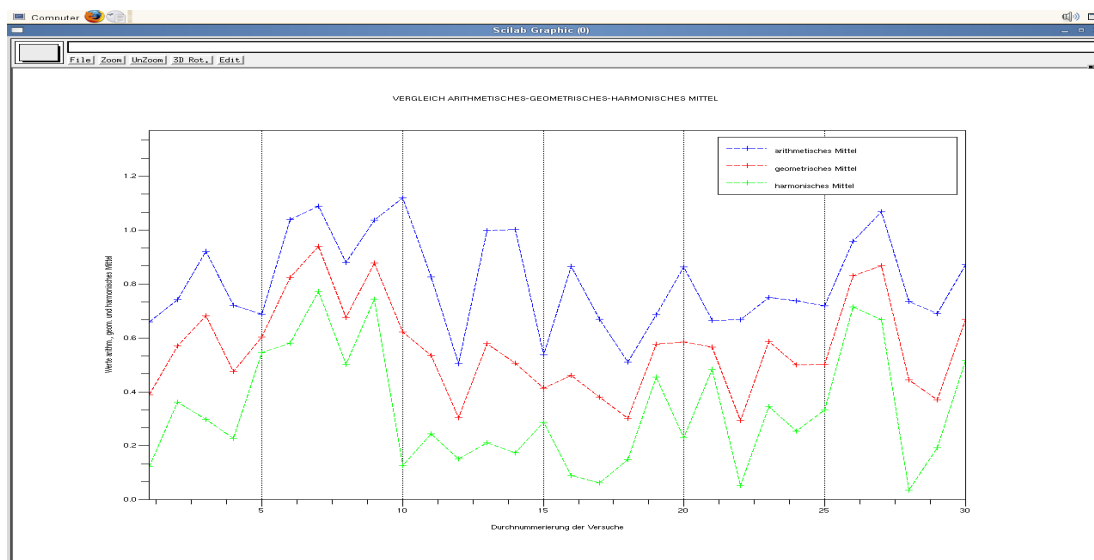


Figure 2: Visualizing the arithmetic, geometric and harmonic means of 10 random positive numbers ( $n=30$  runs).

This point of view may be illustrated by the following exercise aimed at introducing the arithmetic-geometric-harmonic inequality. The student is given the mathematical expression for the arithmetic, the geometric and the harmonic mean of positive numbers and asked to write a Scilab-function computing the means of  $n$  given positive numbers. The student is asked to test the function to ensure that it works and does generate the desired quantities. The function is then run using  $n$ -tuples of randomly generated positive numbers and their means are visualized (Figure 2).

This is done in a such a way that the student cannot fail to notice that the arithmetic, geometric and harmonic means occur in a certain order and that this situation persists when modifying the number  $n$  of positive numbers and the number of runs of the function. The student is then asked to infer a mathematical result from the visualization and to state it using sound mathematical notations.

### Implementation considerations

As a conclusion we shall give a brief outline of the state of implementation of our material. For the design of the graphical applets, we rely in part on expertise and existing Java code by project MUMIE (Multimediale Mathematikausbildung für Ingenieure, teaching engineers mathematics using multimedia) at the TU Berlin. This project has been implementing applets for the linear-algebra course in the past years to much appraisal and positive feedback. On the other hand, we now have Scilab-exercises addressing all the notions taught in single variable calculus and many notions from multivariate calculus. We plan to test this material in an experimental tutorial during the summer term of 2008.

### References

Edwards, P. and Edwards, P.M. (2003) Tackling the Mathematics Problem with *MathinSite*, Global J. of Engng. Educ., Vol. 7 (No.1): 95-102.

Mustoe, Leslie and Lawson, Duncan (2002) "Mathematics for the European Engineer. A curriculum for the twenty-first century. A Report by the SEFI Working Group on Mathematics". Accessed via <https://learn.lboro.ac.uk/mwg/core/latest/sefimarch2002.pdf> (11 February 2008).

Nguyen, D. Q. (1998) The Essential Skills and Attributes of an Engineer: A Comparative Study of Academics, Industry Personnel and Engineering Students, Global J. of Engng. Educ., Vol. 2 (No.1): 65-75.

Schott, D. (2005) Challenges in Computer Mathematics in Engineering Education. Global J. of Engng. Educ., Vol. 9 (No.1): 27-34.

Tiedt, R.-P. (2001) Unterstützung der Mathematikausbildung für Ingenieure durch Nutzung von Matlab, Global J. of Engng. Educ., Vol. 5 (No.3): 283-287.

Wlodkowski, P. (2006) Teaching numerical methods in Engineering with Mathcad, Annual Conference of the American Society for Engineering Education "Advancing Scholarship in Engineering Education", Chicago, IL. Accessed under <http://www.asee.org/acPapers/code/getPaper.cfm?paperID=10334&pdf=2006Full1549.pdf> (11 February 2008).