INTEGRATION OF A COMPUTATIONAL MATHEMATICS EDUCATION IN THE MECHANICAL ENGINEERING CURRICULUM

Mikael Enelund
Department of Applied Mechanics

Stig Larsson
Department of Mathematical Sciences

Johan Malmqvist
Department of Product and Production Development

Chalmers University of Technology, Gothenburg, SWEDEN

ABSTRACT

The rapid development of computers and the internet has given new opportunities for engineering work as wells as for teaching and learning. The use of advanced modern mathematics is becoming increasingly more popular in the engineering community and and most problem solutions and developments incorporate high precision digital models, numerical analyses and simulations. However, this kind of mathematics has not been fully implemented into current engineering education programs. Students spend too much time on solving oversimplified problems that can be expressed analytically and with solutions that are already known in advance. Instead, we should be using computers to solve more general, real-world problems. Here we present the integration of a computationally oriented mathematics education into the CDIO-based MSc program in mechanical engineering at Chalmers. We found that the CDIO-approach was beneficial when designing a reformed mathematics education and integrating the mathematics in the curriculum. In the reform of the mathematics education, traditional symbolic mathematics is integrated with numerical calculations and the computer is used as a tool. Furthermore, the computer exercises and homework assignments are taken from applications of mechanical engineering and solutions are analyzed and discussed by means of simulations. The experience is very positive. The students' interest for computation and simulation has increased. The students consider the the computer to be an important tool for learning and understanding of mathematics. Students spend more time training mathematics and solve more problems.

KEYWORDS

Engineering mathematics, Integrated, computational oriented mathematics education, Full view of problem solving, Virtual learning environments

INTRODUCTION

The development of computers, hardware and software, has led to new possibilities for engineering work in which mathematically complex problems solved in the computer, visualization and simulation play a central role. The development has also lead to to better conditions for teaching as well as learning of mathematics and other engineering disciplines. It is possible to solve not only special, simplified problems but also complex realistic problems and to visualize solutions, phenomena and theoretical aspects. Visualization can be used to facilitate understanding and learning and promote interest in mathematics. In general, the basic courses and textbooks in mathematics do not take advantage of this development.

The CDIO model for engineering education stresses engineering fundamentals set in the context of Conceiving – Designing– Implementing – Operating real-world systems and products. Further, the model emphasizes a holistic view on problem solving and the ability to translate skills into practice [1]. This has in particular affected the design courses, project courses and more applied courses. Today facilities suitable to build physical models and prototypes exist in many engineering programs. An important part of problem solving chain is simulations and the need for a virtual prototype lab (that is, an engineering tool for calculation and simulation) is obvious. The introduction of this requires a more computation and simulation oriented mathematics education. Furthermore, we are convinced that the computational aspects, construction of numerical algorithms and programming of mathematics should be included from the beginning in all mathematics courses. This it makes it possible to study more realistic problems and to use simulations, which promotes understanding and motivates the study of mathematics.

In the academic year 2007–2008 a reformed mathematics education for the Master of Science in program in Mechanical Engineering at Chalmers was launched.

MATHEMATICS IN CDIO

Before we elaborate on our approach to teaching mathematics, it is appropriate to review the mathematics-related elements of the CDIO framework – the syllabus and the standards.

What mathematics should be taught in a CDIO program? On one level, this is a question that is not considered in the CDIO syllabus. Mathematics primarily belongs to the discipline-dependent section 1 – Disciplinary knowledge and reasoning. Indeed, mathematics is discipline-dependent and programs in mechanical engineering and computer engineering, for example, emphasize different mathematical topics. However, also the sections 2–4 of the CDIO syllabus [2] include mathematical topics. CDIO skills topics with a strong foundation in mathematics include 2.1.2 Modeling, 2.3.4 Trade-offs, Judgment and Balance in Resolution, and 4.4.4 Disciplinary Design. Broadly speaking, it can be argued that the mathematics in a CDIO program should, in the context of the specific discipline of the program, emphasize modeling, decision-making and design.

How should mathematics be taught in a CDIO program? Again considering the CDIO framework, we observe that several CDIO standards [3] are relevant for the mathematics curriculum in a program, and propose strategies for designing the mathematics teaching, learning and assessment in the program. Standard 6 – Engineering workspaces – suggests that a CDIO program should be supported by student workspaces that supports hands-on, concrete learning of product realization. Most published examples of CDIO workspaces (*e.g.,* [4]) has described physical prototyping workspaces. However, it is equally important that a CDIO program is supported by a virtual learning environment designed with the same underlying principles in mind including interactivity, use of

state-of-the-art modeling and simulation tools, and computer-based experimentation and assessment. In a CDIO program, the mathematics curriculum is enriched with experiential, industry-related tasks: students solve real problems using computer tools ([5]). Standards 7 and 8 – Integrated and active learning experiences – interpreted in the perspective of mathematic – point to the need to integrate mathematics learning with other disciplines and to adapt active learning techniques in the mathematics curriculum. The benefit of integrating mathematics education with computational-intensive disciplines such as strength of materials might be evident but there is also potential in integrating mathematics and the training of communication skills, see [7]. Such an approach will bring the mathematics learning closer to learning in design-oriented courses, strengthen student motivation and attainment [9].

In conclusion, a CDIO program should include a mathematics curriculum where

- Modeling and decision-making are brought forward in the context of the mathematics underlying the specific discipline of the program.

- Mathematics teaching and learning is integrated with disciplinary courses.

- Active learning techniques such as simulations and parameter studies are emphasized.

- An interactive virtual mathematics learning environments which supports teaching, learning and assessment.

In the paper, we will return to these basic characteristics and explain and exemplify how they are addressed in our program, the mechanical engineering program at Chalmers University of Technology.

THE MECHANICAL ENGINEERING PROGRAM AT CHALMERS

The Master of Science in Mechaical Engineering program is a five year program divided into two cycles in accordance with the Bologna structure. The first cycle consists of three years of full time studies and corresponds to 180 ECTS and ends with the degree of Bachelor of Science. The second cycle is a two year's (120 ECTS) master program. After completing both cycles the student is awarded the Swedish degree "Civilingenjör" as well as the degree of Master of Science.

In 2000 the Mechanical Engineering program at Chalmers University of Technology teamed up with programs from The Royal Institute of Technology (KTH), Linköping University and Massachusetts Institute of Technology to form the Wallenberg CDIO project, which later become the CDIO Initiative. This was the starting point for an education development process with focus on engineering as a profession that involved many changes including a framework for curriculum design, the integration of general skills, pedagogic and learning environment innovations [8]. Recent developments using CDIO-model at the Mechanical engineering program at Chalmers are the integrated computational mathematics education and sustainability.

The program has organized the education and curriculum focusing on the engineer's professional role, the integration of non-technical skills and contexts. The base of the program is the fundamentals of mathematics and mechanical engineering with emphasis on common principles. This by having joint projects and assignments between mathematics and the basic courses in mechanics and strength of materials. These projects include the full view of problem solving, from selecting a model and setting up equations, describing the model to solve equations and simulate and to assess quality of the choice of model and the accuracy of the solution. The purpose of working with the full view, joint projects and the sequence of courses is that education and learning of a topic shall not be isolated in a specific course. Applications from the Fundamental topics are introduced early in the

learning plan to prepare for the design-implement-(design-build) projects where real and relevant products and systems to be created. There are at least one project in each grade.

The basis for program development and program monitoring is the CDIO-based program description with which decompose in course objectives through a program design matrix. The program description can be found at the Program website [6]. Design and integration of the mathematics education are based on program level goals and in which courses they are fulfilled are displayed in the program design matrix. Below we cite the program goals related to mathematics

The Master of Science in Mechanical Engineering graduate shall

1 Be able to put into practice (apply) mathematics and fundamental science within applied mechanics and have an insight into basic principles of classical physics with focus on

   1.1 being able to solve linear and nonlinear systems of algebraic equations by numerical methods,

   1.2 being able to solve ordinary differential equations of the following types; separable, inhomogeneous with constant coefficients and Euler's,

   1.3 being able to solve by numerical methods linear and nonlinear ordinary differential equations inclusive reformulating to a first order system,

   1.4 being able to solve the eigenvalue problem for continuous and discretized systems

   1.5 being able to use the Finite element method to solve partial differential equations,

  1.10 based on given models and mathematical formulas, being able to program solutions, including graphic presentations of engineering problems in Matlab.

4 Be able to formulate theoretical models and set up equations to describe the models. Solve equations in order to simulate reality and assess the reasonableness of the choice of model along and the solution's level of accuracy.

5 Be able to analyze, solve and simulate advanced mechanical engineering problems within the selected specialization area/master's program by using modern, computer-based tools and from these, selecting the most appropriate ones

The reformed courses in mathematics have been developed to meet these goals. Cornerstones in the reformed mathematics education are

- To highlight and clarify modeling, computations and simulations.

- Full integration of computational aspects (including programming) and symbolic aspects of mathematics,

- Emphasis on the full view of problem solving, *i.e.*, set up the mathematical model, formulate the equations, solve the equations and visualize the solution to asses the correctness of the model and the solution,

- Computer exercises where students solve problems including visualization by developing their own codes in Matlab.

- The finite element method, which is taught and used already in first year courses in mathematics and mechanics.

- Assignments, exercises and applications are taken from the parallel courses in mechanics and strength of materials, and other fundamental mechanical engineering courses. Technical relevant problems are solved.

- Close cooperation with courses in mechanics and strength of materials, including joint project work, computer exercises and assignments.

- Teaching and learning of mathematics in applied courses such as mechanics, strength of materials and control theory.

- Teaching and learning of mechanics and physics in mathematics courses.

In the work of the reformed mathematics education, we have specifically developed:

- A basic course in programming in Matlab.

- Compendium and lecture notes in computational mathematics.

- Computer-oriented exercises, assignments and team projects that are used simultaneously in the mathematics courses and in courses of mechanics and solid mechanics.

- Interactive learning environments in mathematics and supported the development of a Virtual learning environment in the statistics courses

The new mathematics education covers all the mathematics courses in the first year and the mathematical statistics course in the third year. Mathematics courses and simultaneously taught courses are listed below (the academic year is divided into four study periods of eight weeks).
Year 1

- Study period 1.*Introductory course in mathematics* (7.5 ECTS), *Programming in Matlab* (4.5 ECTS) and *Introduction to mechanical engineering* (continued in Study period 2).

- Study period 2.*Calculus in a single variable* (7.5 ECTS), *Computer aided design* (4.5 ECTS) and *Introduction to mechanical engineering* (7.5 ECTS).

- Study period 3. *Linear algebra* (7.5 ECTS) and *Statics & strength of materials* (7.5 ECTS).

- Study period 4. *Calculus in several variables* (7.5 ECTS) and *Strength of materials* (7.5 ECTS).

Year 3

- Study period 4.*Mathematical statistics* (7.5 ECTS) and *Bachelor's thesis project* (15 ECTS, launched in Study period 3).

Year 3 also includes the elective course *Transforms and differential equations*. This course is not included in the new mathematics education and is therefore not discussed here. Further, year 3 also includes the elective course *The finite element method*. This course is a continuation of the courses *Calculus in several variables* and *Strength of materials* and uses in a natural manner the same pedagogics.

Numerical calculations, Matlab programming and simulations are included in most of of the applied courses in years 2 and 3, for example, *Machine design*, *Mechatronics*, *Thermodynamics & energy technology*, *Integrated design and manufacturing*, *Materials and manufacturing engineering*, *Fluid mechanics*, *Control engineering*, *Heat transfer*, etc. The final two years, the specialization, is made in a masters program. Numerical calculations, Matlab programming and simulations are also included in most masters programs.

THE REFORMED MATHEMATICS EDUCATION

The main idea of the reformed mathematics education is a complete integration of computation (numerical analysis) and analytical (symbolic) mathematics. This requires computers and a programming environment. We chose Matlab because it is a suitable environment, it is easy to use, there are many built-in functions and "Tool Boxes"(packages) for applications, and Matlab is used both in subsequent courses and in research at departments and companies linked to the Mechanical Engineering program. It is also relatively easy to create graphs, simulations and animations.

Matlab is used by students for calculations and illustrations of mathematical concepts and phenomena using advanced built-in functions and packages, students write their own program for implementation of numerical algorithms for solving technical problems from applications and to present results. We think it is very important that students write their own programs. This provides students with programming skills and knowledge, and it increases their understanding of mathematics and algorithm construction. Finally, it gives students the confidence to solve all kind of problems, provide training in abstract and logical thinking and problem-solving, which is very valuable in future courses and the working life.

The use of computing in mathematics courses increases the motivation for the study of difficult mathematical concepts. When we generate approximative sequences with the bisection method it becomes necessary and meaningful to study the convergence of sequences and we can carry out proof of the Intermediate Value Theorem. Integral theorems of multivariable analysis may in a traditionally structured course only be used to rewrite difficult integrals, which the students (and the professor) probably consider to be pretty useless. We can now justify the Gauss divergence theorem by deriving the heat equation with mixed boundary conditions (as well as pointing out the similarity with boundary value problems of elasticity theory) and to derive the finite element method. Our approach was inspired by [10] and first developed in the Chemical Engineering Program at Chalmers [11, 12]. For further discussion, see [13].

The course literature consists of two traditional mathematics textbooks, [14], [15], supplemented with a compendium of Computational Mathematics and computer exercises, etc. In the "Mathematical statistics" course the literature is linked as PDF-documents in the Virtual Learning Environment.

Teaching consists of lectures (2–6 hours/week); exercises (0–4 hours/week), computer exercises (2–4 hours/week) and project team work. The examination is mainly in the form of a written final exam that may be located in a computer lab together with quizzes, computer exercises, and project reports on a scale that varies between the courses.

The following is a short description of the courses with emphasis on the computational topics.

INTRODUCTORY COURSE IN MATHEMATICS

The course covers functions of one variable, continuity and derivative. Vector geometry and the Gauss elimination method for linear systems of equations are also included.

We put a special emphasis on the proof of the intermediate value theorem via the bisection algorithm, which allows us to introduce concepts like Lipschitz condition, convergent sequence, decimal expansion (= real number) in a computational context.

Computer exercises:

1. *Function gallery.* To plot the graphs of a large number of functions that are important to know.

2. *Geometry.* To write Matlab functions for the scalar product, orthogonal projection, cross product and so on.

3. *The bisection algorithm.* To implement the bisection algorithm in order to deepen the understanding of the intermediate value theorem, convergent sequence, and decimal expansion. The program solves equations of the form $f(x) = 0$ with arbitrary function f.

4. *Fixed point iteration.* To write a Matlab function based on the the fixed point theorem for contraction mappings. This solves equations of the form $x = g(x)$.

5. *Numerical derivative.* To write a Matlab function for approximative computation of the derivative of an arbitrary function.

6. *Newton's method.* To write a Matlab function for Newton's method with numerical computation of the derivative. This solves equations of the form $f(x) = 0$. This is repeated in study period 2 and 4 in more general contexts.

MATHEMATICAL ANALYSIS IN ONE VARIABLE

We continue the study of functions of one variable and the computational topics include computation of the integral and solution of differential equations.

In the computer exercises we write our own Matlab functions for the following:

1. ODE1: primitive function.

2. ODE2: Euler's method for systems of ODE (initial value problems).

3. ODE3: implicit methods for systems av ODE (initial value problems).

4. ODE4: a boundary value problem (a shooting method combining a solver for initial value problems with with the Newton solver from the previous course). The program is used to solve a heat conduction problem.

LINEAR ALGEBRA

This is a rather traditional course in linear algebra but with computer exercises from the simultaneously taught course *Statics and strength of materials*:

1. *Matrix algebra.* Analysis of a statically determined large plane truss. In the course "Statics and strength of materials" the truss is made indeterminate and the truss is analyzed by the displacement based matrix method. This also serves as an introduction to the finite element method.

2. *Geometry.*

3. *Error analysis* for the solution of linear systems of equations based on the condition number of a matrix.

4. *The least squares method.* Calibration of Norton's law for creep of copper at elevated temperature.

CALCULUS IN SEVERAL VARIABLES

In addition to the traditional topics of multivariable calculus, we introduce *boundary value problems* for partial differential equations and *the finite element method*. We begin with boundary value problems in one variable of the general form:

$$-D(a(x)Du(x)) + c(x)u(x) = f(x) \qquad \text{for } x \in I = (K, L),$$
$$a(x)D_n u(x) + k(x)(u(x) - u_A) = g(x) \qquad \text{for } x = K, x = L,$$

and we derive the finite element method in one variable. The derivation is based on the fundamental theorem of calculus and integration by parts. After this we can do the same thing in several variables:

$$\begin{cases} -\nabla \cdot (a\nabla u) + cu = f & \text{in } D, \\ \hat{\mathbf{N}} \cdot (a\nabla u) + k(u - u_\mathsf{A}) = g & \text{on } S_2 \text{ (Robin boundary condition)}, \\ u = u_\mathsf{A} & \text{on } S_1 \text{ (Dirichlet boundary condition)}. \end{cases}$$

Here we use the Gauss divergence theorem (a multivariable version of the fundamental theorem of calculus) and the integration by parts formula

$$\iiint_D \nabla \cdot \mathbf{F}\phi \, dV = \iint_S \hat{\mathbf{N}} \cdot \mathbf{F}\phi \, dS - \iiint_D \mathbf{F} \cdot \nabla\phi \, dV.$$

The boundary value problems are interpreted in terms of heat conduction, but the analogy with the equations of strength of materials is also displayed. We also think that it is important to derive and understand the meaning of the boundary conditions in this general form so that the students will be able to use professional finite element programs where all these terms are present.

Computer exercises:

1. *Visualization* of multivariable functions.

2. *Jacobi and Newton*. The students implement Newton's method for systems of nonlinear equations with numerical computation of the Jacobi matrix.

3. *Extreme value problems*. Critical points are computed by means of the Newton solver from the previous exercise.

4. *The finite element method in 1–D*. We use a Matlab program for boundary value problems in one variable with the same datastructures as the "PDE Toolbox" as a preparation for the following exercise.

5. *The finite element method in 2–D*. We use the "PDE Toolbox" of Matlab.

MATHEMATICAL STATISTICS

The course covers basic probability theory and statistics with emphasis on concepts and computational methods of importance for mechanical engineering applications. The course also introduces elements of experimental design. The course is built-up by lectures and web-based exercises with the use of a Virtual Learning Environment (VLE) aided with statistical routines of Matlab. The virtual learning environment includes about 700 numerical examples and a compulsory team project. The team project considers prediction of probability for crack growth and failure of a railway structure (the Iron Ore-line between Kiruna and Luleå in the north of Sweden). The input data are the contact forces between rail and wheel and the temperature measured in the field. The project includes a report that is assessed by the teachers.

EXAMPLES OF PROJECT WORK AND COMPUTER ASSIGNMENTS

COMPUTER ASSIGNMENT 4 — FINITE ELEMENT METHOD IN 1D

From the course *Calculus in Several Variables* we the following assignemnts.

GOALS

To learn how to solve boundary value problems in one variable by the finite element method in Matlab. This is also a preparation for the "PDE Toolbox" for finite elements in two variables.

LITERATURE

FEM1.

Matlab programs

MyPoissonSolver.m. BdryData1.m. EqData1.m.

Copy the files by clicking on the links. Type `help MyPoissonSolver` on the command line and read the documentation.

INTRODUCTION

The program `MyPoissonSolver` solves boundary value problems of the form

$$- D(a(x)Du(x)) + d(x)Du(x) + c(x)u(x) = f(x), \quad \text{for } x \in I = (K, L),$$
$$a(x)D_n u(x) + k(x)(u(x) - u_A) = g(x), \quad \text{for } x = K, x = L.$$

Here $D = \frac{\mathrm{d}}{\mathrm{d}x}$ and D_n is the directional derivative in the outward direction, i.e., $D_n = -\frac{\mathrm{d}}{\mathrm{d}x}$ at $x = K$ and $D_n = \frac{\mathrm{d}}{\mathrm{d}x}$ at $x = L$. The program is based on the finite element method with piecewise linear functions.

The function `MyPoissonSolver` with the declaration

```
function [U, A, b] = MyPoissonSolver(p, t, e, EqData, BdryData)
```

assembles and solves the system of equations $AU = b$, where A is the the stiffness matrix, b is the load vector, and the vector U contains the node values $U_i = U(x_i)$ to the finite element solution $U(x) = \sum_{i=1}^{n} U_i \phi_i(x)$.

Information about the computational mesh is stored in the matrices p,t,e with the same structure as in Matlab's "PDE Toolbox" that we will use later (also in the mechanics course). The data of the problem, a, d, c, f, u_A, g, are defined in the function files `EqData.m` and `BdryData.m`.

The matrix p. The coordinates of the nodes

$$K = x_1 < x_2 < \cdots < x_{i-1} < x_i < \cdots < x_{n-1} < x_n = L$$

are stored in the vector p of type $1 \times n$.

The matrix t of type $3 \times (n-1)$ contains information about the $n-1$ intervals

$$I_i = (x_i, x_{i+1}), \quad i = 1, \ldots, n-1.$$

More precisely, column number i contains indices (pointers) pointing to the endpoints of interval number i, that is,

$$\begin{bmatrix} i \\ i+1 \\ 1 \end{bmatrix}$$

The third digit is a tag ("subdomain reference tag"), which I have set to 1 here, and which can be used to mark which subdomain the interval I_i belongs to, if you have divided the interval $I = (K, L)$ into subdomains. This could be practical if the coefficients are given by different expressione in different parts of I.

The matrix e contains information about the boundary points,

$$e = \begin{bmatrix} 1 & n \\ 1 & 2 \end{bmatrix}.$$

Here the first row contains pointers to the two boundary points, here x_1 och x_n. The second row contains tags ("reference tags") marking which boundary point it is, here 1 is the left endpoint and 2 the right endpoint. Since we use pointers it does not matter in which order they are entered. The following matrix gives the same result:

$$e = \begin{bmatrix} n & 1 \\ 2 & 1 \end{bmatrix}.$$

The same is true for the matrix t.

This may seem unnecessarlily complicated but it is a preparation for the "PDE Toolbox", where this structure is needed to describe a mesh of triangles in the plane. In two dimension there is no natural way to number the points, triangles, and boundary points, so we need to use pointers. In the "PDE Toolbox" p, t, e means "points", "triangles", "edges".

PROBLEMS

PROBLEM 1. PIECEWISE LINEAR FUNCTION.

Create a mesh in the interval $I = (0, 1)$ with only $n = 9$ points (so that you can clearly see all of them):

```
>> n=9
>> p=linspace(0,1,n)
>> t=[1:n-1; 2:n; ones(1,n-1)]
>> e=[1 n; 1 2]
```

Create and plot a piecewise linear function:

```
>> V=sin(7*p)
>> plot(p,V,'.-')
```

(See Figure 3 in FEM1.)

PROBLEM 2. MYPOISSONSOLVER

Run the program with the same mesh and the given function files `EqData1.m` and `BdryData1.m`. Read the documenation (`help MyPoissonSolver`) and the files to see what boundary problem it is. (It is one of Problem 1.1–1.5 in FEM1.)

```
>> [U, A, b] = MyPoissonSolver(p, t, e, @EqData1, @BdryData1);
```

Look at the stiffness matrix A and see that it is tri-diagonal.

Plot the approximate solution U and the exact solution u in the same figure.

Refine the mesh to $n = 101$ points and compute again.

FURTHER PROBLEMS

The remaining problems cover heat conduction in an inhomogeneous material, traction of a bar, elasticity in rotational symmetry.

PROJECT WORK 2 — STRESS INTENSITY FACTOR COMPUTED WITH FEM

This project is from the course *Strength of materials*.

The project considers a two dimensional stress analysis of a thin plate with three holes subjected to uniform stress at the vertical boundaries, see Figure 1. Plane stress conditions are assumed. beginfigure[hhh]

The analysis is carried using the finite element method and the PDE-toolbox in Matlab. The task is to calculate the stress concentration factor K_{t} which is defined as

$$K_{\mathrm{t}} = \sigma_{\max}/\sigma_{\mathrm{non}}$$

By varying the distance b in the figure, the students should be able to decide whether the stress concentrations near the holes are correlated or not. Further, the calculated Kt should be compared with tabulated values from handbooks. To reduce the number of elements, symmetries should be used and only a quarter of the plate needs to be considered. This means that special attention needs to be put on the boundary conditions. There are several aims with this assignment, for example: (1) By visualizing the stress distribution, the students can develop an intuition about stress distributions and how the stress is increased due to abrupt changes in geometry. (2) Motivate the need to study the governing equations of elasticity. (3) It serves as an introduction to the finite element method. (4) It provides an introduction to error estimation and adaptive mesh refinement in the finite element method.
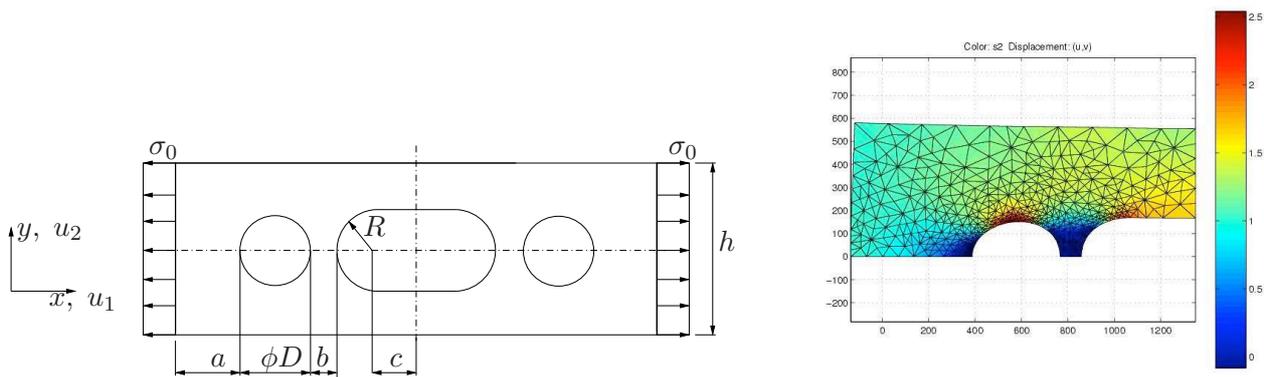
Figure 1: Left: plate with three holes. Right: the color represents the stress distribution in a quarter of the plate. The largest principal stress is presented. The applied stress is $\sigma_0 = 1$. Note the higher stress near the holes and the mesh refinement around the holes. Note also the deformed geometry.

INTERACTIVE/VIRTUAL LEARNING ENVIRONMENTS

Young people learn much differently than they used to just a few decades ago. Researchers in the area of pedagogy refer to this as the "Nintendo Syndrome", see [17]. The video-games generation does not read manuals. Instead, they jump right in and give it a try. If you get "killed", you just try something else until you get to the next level. If that does no work, just go to the internet to get a hint. We have noted that this attitude seems to prevail when it comes to studying. Convincing students to read a book has become increasingly difficult. But students do find it educationally rewarding to actually try something before reading about it. Naturally we need to take this new learning style into account and take benefits. We have noticed that the new learning style can be very effective. For example, students learn to use software (*e.g.*, for CAD, finite elements, material selection, virtual production documentation, *etc.*) faster and more efficient today compared to some years ago and this without reading the manuals. In order to exploit this and to enhance teaching and student learning, we have introduced virtual or interactive learning environments.

In the course *Mathematical statistics* we provide the students with a Virtual Learning Environment (VLE) containing randomly generated problems and questions (quizzes) on every subject of the course syllabus, extensive system of hints and answers, links to the class text and other support materials: statistical tables, demonstrations, help files, see [18]. VLE is developed at the Department of Statistics and Modelling Science of the University of Strathclyde, UK, and at Chalmers University. The reformed course *Mathematical statistics* is entirely ported to VLE. The course was given for the first time the academic year 2009/2010 and had about 200 students. The novel approach consists in shifting the weight from formal lectures and towards self-practise and self-studying with the help of the VLE also aided with statistical routines of Matlab computer package. This makes it also possible to study large complex open ended problems connected to applications of mechanical engineering. Obviously, the VLE fits the CDIO-paradigm providing the students with a web-based framework for probability and statistics. Since VLE is web-based, the VLE is accessible from anywhere anytime, it does not depend on the operating system used, it has infinite number of variations of the study questions, so it provides students with valuable resource to practice beyond the assisted computer lab times. A snapshot of a web-browser running VLE is shown in Figure 2.

Figure 2: Student VLE training sesion.

Another feature of VLE is that it provides teachers with an administrative tool for communication, documentation, timetabling, diagnostics, surveys and monitoring each student's activity in the VLE. Each problem tried can be examined for errors which allows for individual support, see example of the activity log in Figure 3.

Figure 3: VLE adminstrator view.

The VLE is also used for assessment, tests and examination. By the end of the course the students become very familiar with its environment and this helps to ease the examination stress. The final exam is carried out in computer lab with using VLE and with access to Matlab. Since marking is done automatically, VLE saves a huge amount of teachers' time. This time can preferably be used to discuss with and tutoring the students in scheduled classes in computer lab.

The VLE meets the challenges with new learning attitudes by providing the students with a tool for learning by practices. The VLE also trains and prepares the students for modern way of engineering work based on computer calculations and simulations. The VLE provides the student with a way to practice as much as they want to become confident with the topic. On average the students answered 150-200 problems during the course. We do not think this would have been possible in a statistics course given in a traditional manner. The results shown by the students were excellent, over 95% of registered students passed to course which is much higher compared to previous years 70-80% passed. The students sincerely enjoyed working in the VLE and meant that it is a great way of learning statistics.

In this academic year an interactive learning environment based on Maple TA [19] has been introduced the courses "Linear algebra" and "Calculus in several variables". So far this consists mainly of automatically graded exercises and tests, but it will be successively developed into a more advanced interactive learning environment for all the mathematics courses.

EVALUATION AND RESULTS

All course are evaluated using the standard Chalmers system. The evaluation is carried out in cooperation between the program management, the students and the teachers. For each course five to seven student representatives are selected as a reference group. The reference group meets the teachers three times . The final course meeting takes place after the course has been completed, the head of the program /and the program coordinator are also present. Before the final meeting, a web-based questionnaire is distributed (via email) to all students, to be answered anonymously. The results are collected and are discussed at the final meeting and presented in the minutes from this meeting. The minutes from the final meeting are published on the Chalmers web site. In addition, at Mechanical engineering we have annual class feedback meetings with all students that have been appointed course representatives during the. At those meeting, the full view of the curriculum is discussed, *e.g.*, how courses connect to each other and how the program aims are meet. For the particular evaluation of the mathematics education we have put specific questions in the web-based questionnaires, interviewed a group of ten third year students as well as interviewed teachers of following design-build-test projects and master level courses in structural dynamics and employers at a consulting company and at a large global company.

The main goal that each student should gain knowledge, skills and ability to effectively use computational mathematical modeling and simulations in applications has been reached to a large extent. Employers claim that the mechanical engineering students have became significantly better prepared for the managing and solving of open-ended problem, carrying out numerical simulations, programming and using modern industrial softwares. Teachers of structural dynamics courses verify that the students' ability to solve large complex problems have improved and that the computational skills in are much better. They also claim that Mechanical engineering students in average are much better prepared for the courses and can handle computations and projects involving finite element simulation more efficiently and at higher level compared to engineering students from other engineering programs at Chalmers and outside Chalmers. Furthermore, the teachers of the subsequent Design-Built-Test coursers have noticed that the students carry out more adequate analyses and simulations using the computational tools, e.g., the Finite element method, from the mathematical

courses. This has improved the quality of the products developed. Teachers in parallel and following courses have noticed a marked improvement the ability to program Matlab, which is not surprisingly given the programming course and all computer labs.

The students appreciate the ability to work with realistic models and the possibility to gain insight and understanding of the of the behavior of the systems studied. They mean that it is natural to use the computer in mathematics courses and regard the computer as an important for calculations as well as for understanding mathematics. A change in thinking (and reasoning) is that very few students question why mechanical engineering students have to study mathematics. The proposed approach has strengthen the connection between the applications and mathematics. This is very important for the engineering education, having in mind that mathematics is the fundamental tool for most of the engineering students. The students believe that computer exercises and cooperation with the mechanics courses taught in parallel increases motivation to study both courses. The motivation for studying the mechanics courses appear to have increased slightly more than the motivation to study mathematics. Although the difference is small, this is a somewhat unexpected result.

One concern is that students give priority to the computational mathematics and do not learn the traditional analysis required for example to solve specific integrals and differential equations that are important in applications. From interviews with students, it is clear that the motivation for the more traditional analysis is high and it becomes more understandable with the help of the computer exercises. Teachers in parallel courses mean that the ability to carry out traditional analysis has not deteriorated but neither improved. The results of the applied courses verifies this. The students' ability and willingness to perform traditional analysis need to be monitored continuously, not least of the concerns of other teachers. Teachers have also expressed a concern that the computer and Matlab will only be a "black box" that deliver results without the know how. We believe that there are no grounds for this concern because students write their own code to solve systems of equations, integrals and differential equations, *etc*. and many problems and exercises are taken from applications where it is important to assess the results plausibility.

The number of students that passes the courses has increased, in particular for the courses *Calculus in several variables* and *Mathematical statistics*, where the proportion of passed at first attempt increased from 40-60% to 70-90% and 60-70% to over 90%, respectively. Further in course evaluations the the students' general impressions of the courses have been increased from adequate to good or excellent. The attendance at lectures, exercises and assisted computer lab sessions are high and significantly higher compared to comparable traditional courses. The teacher responsible for the mathematics courses the first year, Professor Stig Larsson, was 2008 awarded best lecture at the Mechanical Engineering program, Chalmers, by the students. In 2008 Stig also received the Chalmers Pedagogical Award for his efforts to integrate mathematics in engineering subject. The conclusion is that the students are very satisfied with the reformed courses as well as with the teaching in the courses.

CONCLUDING REMARKS

We strongly believe that the proposed education also has the potential to increase the interest for the underlying mathematics. Clearly, the proposed approach strengthens the connection between the applications and mathematics. This kind of mathematics makes it possible to solve the complete problem: from modeling and solution to simulation of the system and comparison with physical reality. This is one of the corner-stones in the CDIO-curriculum. However, our experience also shows that it is important to emphasize symbolic hand calculation and basic programming concepts in the teaching, so that these are not lost in the excitement over the possibility of doing simulations.

Let us return to the bullet list in section MATHEMATICS INCDIO. The main focus of the reformed mathematics education is modeling and simulations which are present and extensively trained in al courses. Here Matlab and finite element simulations are key components but also industrial softwares as ADAMS, ANSYS, CATIA and FLUENT are introduced and used in the applied courses. Decision making is brought forward in the sense that we consider real systems and structures and solve real problems. This means that the results can act as basis for making engineering decisions as well as reasoning at a much higher level compared to results of fictitious oversimplified problems with just an answer. Moreover, decision making is trained at a much higher level in the following design-built-test projects using the computational tools to analyze, investigate, evaluate and design different solutions.

Integration of mathematics into the curriculum and the applied courses are demonstrated and discussed. We conclude that mechanical engineering education offer very good opportunities for the integration of mathematics into courses such as, *e.g.*, Mechanics, Strength of materials and Control theory, but alos the other way around, *i.e.* integrating the applications into the mathematics courses, are very rewarding and important for the education.

Active learning is met largely in simulations, open-ended problems and in the virtual/interactive learning environments are used.

REFERENCES

[1] http://www.cdio.org.

[2] E. F. Crawley, The CDIO Syllabus v 2.3, http://www.cdio.org 2010.

[3] D. R. Brodeur, The CDIO Standards v 2.0, http://www.cdio.org 2010.

[4] P. W. Young, J. Malmqvist, S. Hallström, J. Kuttenkeuler, T. Svensson, and G. Cunningham, "Design and Development of CDIO Workspaces – Lessons Learned", *Proceedings of the 2005 ASEE Conference*, Portland, Oregon, 2005.

[5] L.-E. H. Wuan and W. K. Yoong, "Students' Perceptions of Ways to Learning Engineering Mathematics: Are these ways CDIO-related", *Proceedings of the 5th International CDIO Conference* Singapore, 2009.

[6] https://student.gate.chalmers.se/sv/studier.

[7] J.-Å. Larsson, "Communication of Mathematics' as a Tool to Improve Students? General Communicative Skill", *Proceedings of the 3rd International CDIO Conference*, Cambridge, Massachusetts, USA, 2007.

[8] J. Malmqvist, J. Bankel, M. Enelund, G. Gustafsson and M. Knutson-Wedel, "Ten years of CDIO ? Experiences from a long-term education development process??, *Proceedings of the 6th International CDIO Conference*, Montreal, Canada, 2010.

[9] C. McCartan, P. J. Hermon, and G. Cunningham, "A Model to Sustain Engineering Mathematics in a CDIO Environment", *Proceedings of the 6th International CDIO Conference*, Montreal, Canada, 2010.

[10] K. Eriksson, D. Estep, and C. Johnson, *Applied Mathematics: Body and Soul*, vol. 1–3, Springer, 2004.

[11] S. Larsson, "A reformed mathematics education at Chalmers", Högskoleverkets kvalitetskonferens, Norrköping, September 25–27 2001, http://www.math.chalmers.se/ stig/presentations/norrkoping/norrkoping4.pdf

[12] L. Öhrström, G. Svensson, S. Larsson, M. Christie, and C. Niklasson, "The pedagogical implications of using Matlab in integrated chemistry and mathematics courses", *Int. J. Engrg. Education*, vol. 21, no. 4, pp. 683–691, 2005.

[13] M. Enelund and S. Larsson, "A computational mathematics education for students of mechanical engineering", *World Transactions on Engineering and Technology Education*, vol. 5, no. 2, pp. 329–332, 2006.

[14] R. A. Adams and C. Essex, *Calculus: A Complete Course*, Seventh Edition, Pearson Canada, 2010.

[15] D. C. Lay, *Linear Algebra and Its Applications*, Third Edition Update, Addison Wesley, 2006.

[16] http://www.student.chalmers.se/.

[17] G. Sinclair, M. McClaren, and M. J. Griffin, "E-learning and beyond. A discussion paper." Campus 2020 process for the British Columbia Ministry of Advanced Education, 2006.

[18] S. Zuyev and M. Enelund, "VLE: Virtual Learning Environment for Statistics", *Proceeding of 15th SEFI MWG SEMINAR AND 8th WORKSHOP GFC*, Wismar, Germany, 2010.

[19] Maple TA, http://www.maplesoft.com/products/mapleta/

Biographical Information

Mikael Enelund is an Associate professor in Applied Mechanics and Head of the Mechanical Engineering program at Chalmers University of Technology, Gothenburg, Sweden. His current research includes modeling and optimization of damping and mechanical engineering education development.

Johan Malmqvist is a Professor in Product Development and Dean of Education at Chalmers University of Technology, Gothenburg, Sweden. His current research focuses on information management in the product development process (PLM) and on curriculum development methodology.

Stig Larsson is a Professor of Applied Mathematics at Chalmers University of Technology, Gothenburg, Sweden. His research focuses on the theory of partial differential equations and the finite element method.

Corresponding author

Mikael Enelund
Chalmers University of Technology
Department of Applied Mechanics
SE-412 96 Gothenburg, SWEDEN
+46 31 772 5115
mikael.enelund@chalmers.se