

Gender-Aware Course Reform in Scientific Computing¹

This report will be published in the proceedings from the conference NU2008
(<http://www.nu2008.se/>)

Elisabeth Larsson (bette@it.uu.se), Stefan Pålsson
(stefan@it.uu.se), Jarmo Rantakokko (jarmo@it.uu.se), Lina von
Sydow² (lina@it.uu.se), Michael Thuné (michael@it.uu.se)
Department of Information Technology, Uppsala University,
P. O. Box 337, SE-751 05 Uppsala, Sweden,

September 2008

¹ The project is supported by NSHU and a pre-study was funded by the Faculty of Science and Technology.

² Corresponding author. URL: <http://www.it.uu.se/katalog/lina>

Abstract

The objective of the course reform project presented here was to perform a top-to-bottom reform of the introductory courses in Scientific Computing at Uppsala University. The reform was made gender-aware to make the courses more attractive to women students and in that way broaden the base of recruitment for the courses, the subject and the programs where the courses are present. In the long run we see this reform as a part of a larger process of change of the Master of Science in Engineering programs.

The emphasis for the course reform is on constructive alignment and students' sense-making. The underlying hypothesis is that women students' pay particular attention to sense-making, and courses well-aligned with course goals contribute to sense-making.

The traditional way of teaching courses in Scientific Computing does not align with the overall motivation and goal of the subject. The courses lack constructive alignment. In this report we describe a way to re-design the course activities in a way such that constructive alignment is achieved. Results that support the underlying hypothesis are presented. Excerpts from students' answers to course evaluations and questionnaires provide empirical evidence that the course reform has been successful.

Key words: Scientific Computing, gender, student oriented pedagogic, constructive alignment, student-centred pedagogy

Introduction

Background

The M.Sc. programs in Engineering are among the most prestigious ones in the Swedish university system. Many of their graduates get leading positions in Swedish industry. This implies that it is particularly troublesome from a gender equality perspective that women are under-represented among the students in those programs. This is the more striking since women are slightly over-represented among Swedish university students in general.

There is a variation between the M.Sc. Engineering programs, so that the under-representation of women is largest in programs with many mathematically oriented courses. It would thus be particularly relevant to consider what could be done to make such courses more appealing to women.

Scientific computing is one of the mathematically oriented subjects in the M.Sc. Engineering curriculum. It is about using computers and mathematical models to simulate phenomena related to nature, technical artefacts, etc.

M.Sc. Engineering programs typically contain at least one mandatory course in Scientific Computing. At Uppsala University, students who find the subject interesting can also take additional, optional courses. We found that among the students enrolled in the M.Sc. Engineering programs a larger proportion of the men than of the women chose to take optional, advanced courses in Scientific Computing.

Course reform project

Against this background, we set up a course reform project at Uppsala University, to change the initial, mandatory courses in Scientific Computing. The objective was that a larger proportion of the women students would find the subject interesting enough to also continue with an optional course.

Success of the project would have at least two beneficial effects:

- Better gender balance within the area of Scientific Computing
- Better chance that women would choose to enrol in the M Sc Engineering programs if the courses are attractive to women already enrolled in those programs

A third important effect would be that new categories of male students would find the courses appealing.

A project group was set up, including both staff members and students. In addition, a reference group gave advice on strategic directions and provided a broader perspective on the work. The reference group included the Faculty Dean of Engineering Education, the Director of Uppsala Learning Lab and a representative from the Office for Development of Teaching and Learning.

The basic hypothesis underlying the project was that female students pay more attention to sense-making. This hypothesis is based on results from a Swedish study (Albinsson et al. 2002) where Swedish M Sc. Engineering women students in computer science gave the following suggestions for changes that would make the education more interesting in their opinion:

- Increased possibility to

- See how different parts of their knowledge connect.
- Understand the usefulness of their knowledge in the study environment and professional life.
- See relations between subject area knowledge and reality.
- Technical discussions should have a social perspective.
- More project and group work.
- More oral presentations and writing of reports.
- Better contact and communication with teachers and fellow students.

The items in the upper part of this list are clearly about sense-making. In addition, the items about group work, reports, and communication also have a relation to sense-making. They concern forms of education with a large degree of student activity. When appropriately aligned with course goals and forms of examination such activities have the potential to contribute to sense-making (Biggs 2003).

Traditional education in Scientific Computing

The challenge for our project was to re-design the introductory courses in Scientific Computing to make it easier for students to perceive the meaningfulness of the subject. In order to explain the changes we have made, there is need for a very brief introduction for readers not familiar with the subject.

Scientific Computing is concerned with large scale simulations of phenomena in physics, biology, finance, etc. The simulations are based on mathematical models of the phenomena of interest. A typical everyday example is weather forecasts, where future weather is simulated on the basis of a mathematical description of the weather physics. The simulation consists in solving the equations in the mathematical model and presenting the results in an understandable form, for example as a weather map. The equations are ‘unsolvable’ in the sense that they can not be solved by methods familiar from mathematics. The complexity and the size of the problem preclude such a solution. The computations to solve the problem rather require the use of so called “numerical methods”, i.e., computer-based algorithms to solve mathematical problems. In real-life applications, such as weather forecasting, there are huge amounts of computational operations involved. It is thus necessary to write an efficient computer program to carry out the simulation in reasonable time. The solutions achieved in these numerical computations are always approximations, and one part of the subject is to interpret and validate results.

In summary, to understand what Scientific Computing is one has to be able to take in the whole picture, the context, and see the relation between many different aspects:

- Application areas (i.e., areas where various phenomena are studied via computer simulations)
- Numerical methods
- Programming techniques

The challenge for the teacher in a beginners’ course is to organize learning activities that open possibilities for the students to grasp these aspects and see the connection between them. Only then can the students make sense from the material.

The traditional way of teaching Scientific Computing does not achieve this. A partial explanation is that the traditional Scientific Computing beginners’ course is largely teacher-centred. Teachers lecture about theory without providing sufficient context for the students to

relate the theory to. Teachers demonstrate how to solve model problems, without providing the students with a rationale for why these problems are relevant.

Another part of the explanation is that the traditional Scientific Computing beginners' course lacks constructive alignment. There is too much emphasis on abstractly formulated, small model problems to solve by hand. This does not align with the overall goal to make students aware of real-life applications where computers are required to carry out the computations. A traditional course in Scientific Computing consists of

- Lectures
- Problem solving classes
- Computer labs and assignments
- Final exam

In all these activities, except for computer labs and assignments, the emphasis is completely on theory and small size model problems. The lab and project assignments are intended to give the "big picture" of the subject, but students fail to see the connection between these assignments and the matter covered in lectures and problem solving classes. In fact, if Scientific Computing would be to solve small problems by hand, then the subject would truly lack purpose. Sadly enough, this is the impression that many students have obtained after a traditional first course in the subject.

The course activities listed above are typically organized in the order in which they are listed, i.e. theory comes prior to practical work and applications. This might seem perfectly natural from a teacher point of view, first explain the theory and then illustrate the theory on the computer screen. However, if a student fails to understand the theory, the computer lab will not make any sense whatsoever. It illustrates and gives answers to questions never raised by the student. This is one of the factors making the traditional introductory courses in Scientific Computing teacher-centred.

The approach taken in our course reform project to improve the situation was to re-design the courses according to principles of constructive alignment (Biggs 2003). As a side-effect this made the courses largely student-centred.

It should be noted that although this reform was primarily intended to make more women continue with optional courses in Scientific Computing, the re-design was also expected to give better learning outcomes for all students, female and male.

Method

Action research

In the implementation phase of the course reform project we applied an action research model with cycles of *planning, action, observation, reflection*, followed by *revised planning, action*, etc. In this way, we iteratively re-designed the two compulsory introductory courses in Scientific Computing in M.Sc. Engineering programs. The iterative process involved a total of 15 course instances over a two year period (4 instances Scientific Computing II and 11 instances Scientific Computing I).

Moreover, action research is also used as a model for the re-designed course structure. The intention is to organize the course matter into thematic modules, where learning activities in

each module are meant to support cycles of *action, observation, reflection, tentative understanding*, followed by new *action, observation, reflection, revised understanding*, etc.

In more detail, each thematic module includes the following learning activities, in chronological order (each activity and the relations between activities will be explained in more detail below):

- Computer lab
- Lecture
- Workout
- Problem solving
- Mini-project

These activities serve different purposes and are aligned with the goal that students should be able to make sense of the subject matter and get a relevant overall understanding of what Scientific Computing is.

Computer labs and lectures.

Computer labs are standard ingredients in Scientific Computing courses. The novel aspect of our course design is that the computer labs come *before* the lectures and are intended to generate questions rather than answers. During the lab sessions students work in groups of 2-3 students. The rationale for this is that discussions with fellow students about the lab assignments will support reflection. In addition, a teacher is present during the lab sessions to give advice when needed.

In this new design, the role of the computer lab is to give students some experience of the kinds of computations and applications that the particular course module is focused on. The aim with the computer lab is to provide a context to relate to in the subsequent lectures. Each lab contains at least one realistic application example. In addition, the labs demonstrate numerical methods and contain experiments that high-light their central properties.

During the subsequent *lectures*, the teacher explicitly refers to the lab, so that new theoretical aspects can be motivated by observations that students have made during the lab session. Questions raised by students during the lab are also used by the teacher to structure the lecture.

In summary, the close interaction between lab and lecture is intended to support students' reflection and lead to a revised, more coherent understanding of the subject matter. From the point of view of constructive alignment it is also very important that each computer lab should contain at least one example of a realistic case where the computations would not be feasible to carry out by hand, but really require a computer program.

Workout.

The lectures are followed by a *workout* session. The name "workout" was deliberately chosen to give associations to gym workout sessions. This should give the signal that the students are expected to be active themselves. Moreover, all students know that a workout in the gym is hard work but makes you feel in better shape afterwards. We wanted the workout sessions in the "scientific computing gym" to have the same effect.

Our workout sessions are student-centred. Students work in groups of 2-3 persons, solving problems designed to give a basic understanding of numerical methods and related theory. Discussions with fellow students about the problems help to give better understanding.

The students are encouraged to primarily use the course literature and their own lecture and lab notes to understand how to solve the problems. In addition, a teacher is present during the workout session, to provide further explanations when needed. In this way, the workout session provides an opportunity for individual student-teacher interaction and feed-back.

To ensure alignment with the overall goals of the course it is strongly emphasised that the problems solved during the workout sessions are meant to help the students understand the details of algorithms and theory, rather than solving a mathematical problem and produce the (right) answer. “Real” problems that the algorithms are intended for are impossible to solve by hand.

Problem solving and mini projects.

The next link in the chain of learning activities during a thematic module is the *problem solving session*. This is a scheduled two hour session that begins with a teacher-led brief demonstration of a realistic case study, starting with the statement of a problem and leading to a solution in the form of a computer program. The case study is formulated such that the students reflect on the whole chain: Mathematical model; Numerical method; Programming; Presentation of result.

After the teacher’s presentation of the case study, the students get a similar problem to work on by themselves for the rest of the session, in groups of 2-3 students. The teacher is available to give advice when needed. The goal is that when leaving the session the student groups should understand the problem and have a reasonable solution outline or algorithm formulated on paper.

The work that remains for a complete solution of the problem is to be carried out as a *mini-project*. Then, each student group has to elaborate on its solution, implement it on the computer, run some experiments to demonstrate that the solution works, and write a report. During the mini-project the student groups are expected to work more independently but still with the possibility to get help from the teacher when needed.

An important part of the mini-project report is a “discussion section”. The students are required to formulate at least three Scientific Computing issues in relation to the mini-project problem. Moreover, they should discuss these issues in a “scientific” way, providing solid arguments based on their theoretical and practical insights in Scientific Computing.

The purpose of the discussion section is to encourage the students to see the relations between the problem setting addressed in the mini-project and the more abstract theoretical and practical skills they have acquired from other learning activities in the course. This will help to reach the course goal of understanding the “big picture”, as discussed above.

Assessment.

In order to achieve constructive alignment the learning activities need to be aligned with the course goal. We have now described how the different learning activities included in a typical course module in our course design are intended to contribute to the overarching course goal that the students should perceive Scientific Computing as meaningful and useful.

In addition to this, it is crucial for constructive alignment that the *assessment* of students' progress is aligned with the learning goals. Following Biggs' recommendation, we distinguish between formative and summative assessment. It is mandatory for students to hand in workout solutions and mini-project reports as a basis for formative assessment during the course. This kind of assessment is standard in Scientific Computing courses. The difference in our case is that we make a conscious effort to make it explicit to the students how these assignments relate to different learning goals.

The summative assessment is in the form of a written exam. Compared to a "traditional" written exam in Scientific Computing, we have introduced new kinds of exam problems that relate to real-life situations where the methods studied in the introductory course become meaningful. To test the students' ability to see the "big picture" we have also used essay questions where students were required to write short essays in the same style as the "discussion section" of the mini-project reports.

Result

To quantitatively measure results and changes and relate them to courses structured the conventional way is difficult at this stage. The courses have gradually been changed and we have tried different ways to implement our ideas. In many cases a certain amount of fine tuning has been needed, i.e. to suit a specific student group. A number of different student groups have taken the course, and they all differ in attitudes, "study culture" and traditions. It is too early to follow a particular study programme and see how results and course evaluations change. It is also too early to see any effect on the number of women students on higher courses in scientific computing. Thus, it is difficult at this stage to measure the result of the course reform by means of figures and numbers.

However, it is possible to see changes in students' comments in the ordinary course evaluations and to relate these comments to evaluations on previous (conventional) courses. We have also carried out a short one-page questionnaire in addition to the course evaluations. The emphasis on this questionnaire is on the students' attitudes and feelings about Scientific Computing as a topic. The result presented here is based on these evaluation comments and questionnaires.

It is obvious from the questionnaires and course evaluations that the students view on the courses has been changed. We would like to point out the following:

- students got a clearer view on scientific computing as a topic,
- students consider the courses as well-structured and with well-connected activities,
- female students are significantly more positive to the changes,
- in some study programmes the student attitudes and study culture significantly affects the way they look upon the course reform

Each item is described more in detail below. The first three items are clearly connected to the goals in this project, i.e. alignment, sense-making and the hypothesis that women students pay more attention to sense-making.

Students got a clearer view on scientific computing as a topic.

When teaching Scientific computing the traditional way, it was common with comments in

course evaluations that indicated that the student did not understand what the course and the topic was all about. These comments were there even if the teacher and the course in general got high grades in the evaluation. The quotes below are all taken from questionnaires in Scientific computing I, spring term 2007 (this course was taught in a traditional way) and can serve as examples. All quotes are translated from Swedish. The original text is presented in square brackets after the translation.

“What was the course all about??? Clearer explanation of where and how the course can be of use later. Was it a course about learning matlab and how demanding it is for a computer program to do calculations? Or was it about something completely different?”

“Difficult course to grasp.”

“It was hard to get what we were doing not until the end of the course did I begin to get what it was about.”

[Swedish original:

”Vad gick kursen ut på??? Tydligare förklaring på vart och hur kursen kan komma till användning senare. Var det en kurs i att lära sig matlab och hur krävande det är för ett dataprogram att gör uträkningar? Eller hade det med något helt annat att göra?”

”Svår kurs att greppa.”

”Det var svårt att fatta vad det var man höll på med inte förrän i slutet av kursen började man fatta vad det handlade om.”]

The same underlying problem has been noticed by the teacher on lectures and other activities in the course, resulting in frustrated teachers. The students usually understand details in the course, but not the underlying motivation and overall meaning. We believe that a likely explanation is that the course content was non-aligned and disconnected. With the new course, this type of comments has basically disappeared, indicating that the course has become well-aligned.

Well-structured courses with well-connected activities.

One important part of the project was to change the course activities to make them well-aligned and supportive to the students' sense-making and overall understanding of the subject matter. As described in previous sections the courses are divided into modules and each module contains, in chronological order, computer lab, lecture, workout, problem solving and mini-project.

It is clear, both from course evaluations and the questionnaires, that the different course activities are considered to be well-structured and well-connected. A couple of comments from course evaluations in autumn term 2007 and spring term 2008 exemplifies this:

“The teaching has been well structured and the problems are carefully explained step by step, and it has been possible for everybody to ask without feeling stupid, if you did not understand. To use the knowledge acquired in workout sessions, computer labs, and mini projects was very good. It leads to consolidated knowledge and reduces the difficulties to prepare for the final exam, since you have used most of the elements of subject matter included in the course.”

(Scientific Computing I, spring term 2008)

“The course setup has been super. To schedule the computer labs at the beginning of the course modules, so that they do not require any prior knowledge and results to present was really good. It felt stimulating and totally right. The workouts were very good as well, I particularly appreciate the group work and the presentation of material (i.e., no written report to hand in, but discussions with the instructor during the workout session).”
(Scientific Computing II, autumn term 2007)

“VERY good set-up with computer lab - lecture - workout - assignment!!! Could have been more feedback to the computer labs on the lectures. If you attend a traditional lecture with the teacher solving problems on the blackboard you sometimes just spend the time doing nothing and don't understand the problems solved and maybe don't bother to ask about everything. With workouts you are forced to seriously dig into the problems.”
Scientific Computing II, spring term 2008.

“The mini-projects were really fun to work with. It was also a very smart setup with computer labs first and then theory on the following lecture, it was then easier to grasp the theory.”
Scientific Computing I, spring term 2008.

“It was good to have seen and tested the course content (theory) in the computer labs before the lecture, that way you could relate to what was said.”
Scientific computing I, spring term 2008.

[Swedish original:

“Undervisningen har varit välstrukturerad och problemen förklaras noggrant steg för steg så alla haft möjlighet att fråga om man inte förstått utan att känna sig dumförklarad. Att man använt sig av kunskapen man erhållit i workout, laborationer och miniprojekt har varit väldigt bra då detta leder till en mer befäst kunskap och mindre svårigheter att studera till tentamen då man använt sig av de flesta moment.

Kursupplägget har varit fantastiskt. Att lägga labbar innan segment, så att de inte kräver förkunskaper och resultat att presentera, var verkligen bra. Kändes stimulerande och helt rätt. Workouts var också väldigt bra, jag uppskattar särskilt grupparbetet och presentationen av material (dvs ingen inlämning utan diskussion med närvarande handledare.)

”MYCKET bra upplägg med lab-föreläsn-workout-inlupp!!! Kunde ha återkopplat mer till labbarna på föreläsningarna. Om man sitter på räkneövningar med traditionell tavelräkning sitter man ibland av tiden och förstår ej problemet helt och ids kanske inte fråga om allt. Med workouts tvingas man sätta sig in i problemet på allvar.”

Miniprojekten var jätteroliga att arbeta med. Det var också väldigt smart upplägg att man hade laborationerna först och sedan teorin på nästföljande föreläsning, för då hade man lättare att ta till sig teorin.

Bra att man har sett och provat på innehållet lite i laborationerna innan föreläsningen kommer så man kan relatera till vad som sägs.]

Also, the different course activities are seen as a whole to a larger extent than before the course reform. It is not one activity that supports the learning, it is rather the whole structure of activities.

Female students are significantly more positive to the changes.

All course evaluations and questionnaires can be divided into female and male categories. When studying these, an interesting observation can be made, namely that women students are significantly more positive to the new course structure and consider it to be more

important in the learning process compared with men students. The same trend can be noticed in all courses where we have used the new course structure.

Two figures taken from a course evaluation in Scientific computing I, spring term 2008 can serve as examples. The figures show the students' opinions on the course structure in general (Figure 1), and the construction computer lab before lecture (Figure 2). In both figures category A and B denote female and male students, respectively.

The course is divided into modules with computer lab, workout, problem solving in each module. How has this structure worked?

	#	Graph (%)	%	Σ
1 = Bad	A		0	0
	B	■	9	2
2	A		0	0
	B		0	0
3	A	■	13	2
	B	■	32	7
4	A	■	40	6
	B	■	32	7
5 = Very good	A	■	47	7
	B	■	27	6
No opinion	A		0	0
	B		0	0

Figure 1.

The computer labs come before the lectures. Has it worked well?

	#	Graph (%)	%	Σ
1 = No, not at all	A		0	0
	B		0	0
2	A		0	0
	B	■	9	2
3	A	■	27	4
	B	■	45	10
4	A	■	27	4
	B	■	18	4
5 = Yes, definitely	A	■	47	7
	B	■	23	5
No opinion	A		0	0
	B	■	5	1

Figure 2

A reasonable explanation to this result is that women students find it particularly important to connect the subject and reality and to see the connection between different parts in a course module. This result agrees with and gives support for the hypothesis described on page 3 (Albinsson et al.).

The significance of student attitudes, study culture and study programmes.

The quotes and results presented so far have all been positive to the changes. There are of course also students that are negative or less positive. An interesting observation is that the number of less positive students is strongly correlated to study programme. In most programmes there are a few negative students only, whereas we experienced a polarized opinion in one particular study programme. The following comments are all coming from this group of negative students in that study programme:

”Go back to the old structure. I believe that would make both teachers and students’ feeling better. It is indeed proved that you learn by watching somebody else solving problems.”

”More methods, less understanding in the final exam. There are too few credit points to motivate the time needed for understanding. Understanding is really a good thing and it is good in the long run, but if one should go for understanding ALL courses you wouldn’t manage to cope with the entire study program. You learn courses through methods and then you go deeper into things you believe are interesting. The guest lectures created an interest and I would like to get a deeper understanding of the course content. But I don’t have time for it if I am supposed to pass the other courses at the same time”

”This set-up is not at all good, as I see it. There is so much precious teacher-time wasted, without any chance to learn anything. The computer labs usually make you confused, as you don’t know what you see. The so-called feedback at the lectures seems, at best, forced. The workout is an interesting concept, but again, teacher-time is wasted when sitting and watching students that are trying to solve tasks without any clue how to solve them as they never seen any examples on how to do it. No, I would recommend going back to a more traditional model of teaching.”

“Change the setup lecture-workout-computer lab... During workout and computer lab you get relatively little help from the teacher and it becomes something you can just as well do another time, on your own or with study friends. Then, suddenly it's only lectures, that's only one third(!) of the "teaching" time that feels like teaching.... BAD!”

“Go back to traditional lessons where the teacher solves problems. Old-fashioned but efficient.”

“As I understand it, one has tried to make the course less of a mathematics course. Thanks to that one has completely lost the foundation and makes all later subject matter harder to understand, as you have to derive everything on your own to see that it works.”

“Computer labs before the lectures gives almost NOTHING to the student and is a waste of valuable scheduled time (and departmental money).”

[Swedish original:

“Gå tillbaka till det gamla systemet. Jag tror både lärarna och eleverna mår bättre av det. Det är ju bevisat att man lär sig av att se någon annan lösa problemet.”

”Mer metod, mindre förståelse på tentan. Det är för få poäng för att det ska vara motiverat att lägga ner så mycket extratid för att förstå. Det är jättebra med förståelse och det är bra i längden men om man ska förstå ALLA kurser så skulle man aldrig klara av utbildningen. Man lär sig kurser genom metoder och sen fördjupar man sig i det man tycker är intressant. Gästföreläsningarna skapade ett intresse och jag vill gärna få en djupare förståelse för kursen. Men jag hinner inte det om jag samtidigt ska klara de andra kurser jag läser.”

”Detta upplägg är inte alls bra, är min uppfattning. Det är så mycket lärartid som går till spillo utan att man får chansen att lära sig något. Laborationerna gör en oftast mest förvirrad, då man faktiskt inte vet vad det är man ser. Den så kallade återkopplingen på föreläsningarna känns forcerad i bästa fall. Workout är ett intressant koncept, men återigen så slösar man bort lärarens tid på att sitta och titta på när elever försöker lösa uppgifter utan att ha en aning om hur då de aldrig sett något exempel på hur man gör detta. Nej, jag skulle rekommendera en återgång till en mer klassisk undervisningsmodell.”

”Ändra på upplägget med föreläsning-workout-lab... workout och lab har man egentligen relativt lite hjälp av lärare och det blir något man nästan lika kan göra på annan tid, själv eller med kurskamrater. Så helt plötsligt är det bara föreläsningar, som bara är en tredjedel(!) av "undervisnings"tiden som verkligen känns som undervisning... ILLA!”

”Börja med klassiska lektioner där läraren löser tal. gammalmodigt men effektivt.”

”Som jag förstår det har man försökt göra kursen till mindre av en matte-kurs. Tack vare detta har man tappat hela grunden och gör allt efterföljande svårare att förstå, pga man måste härleda allt själv för att förstå att det fungerar”

”Laborationer före föreläsningarna ger nästintill INGENTING för studenten och är slöseri med värdefull schemalagd tid (och institutionens pengar).” J

Most comments have to do with efficiency. It is more efficient to let the teacher present problems and solutions for the students. There is not enough time for understanding, as one student points out. Under that condition, and if a conventional final exam is used, a teacher-centred structure is probably a more efficient way to pass the course. This result started a discussion in our project group and in the reference group about student attitudes and “learning culture” in relation to different study programmes.

What are the results from a teacher’s perspective? The teacher experiences can be summarized in the following items:

- Better contact between teachers and students and more feedback and interaction between teacher-student,
- Much more student activity and discussions between students
- A general feeling that the students understand what scientific computing is about
- The students learn successively during the course
- The teaching is much more fun

A common question when working with pedagogical reforms is that it results in more work and a higher cost. This is not true in this case. Some parts of this reform imply time saving, or rather reallocation of teacher resources. Presenting solutions on the black board has more or less been replaced with individual discussions with student groups in workouts and problem solving sessions. Time consuming marking of assignments has been reduced and to some extent been replaced with the same type of discussions.

Discussion and conclusions

The course reform described in this paper has been going on for about two years. Important questions are if the students’ attitude about scientific computing as a topic changed? Does it

attract women student to a higher degree? One of the goals was to achieve constructive alignment, is that experienced by students and teachers? In view of the results reported in the previous section, we conclude that the answer to these questions is positive.

We have learnt and experienced a lot in the process, experiences that will be used in other courses. Hopefully the ideas and experiences will spread to other areas and disciplines.

Within Uppsala University and the Faculty of Science and Technology interesting discussions have started as a result of the project. One discussion is about the study culture and student attitudes towards learning in some study programmes. Is it something in the structure in some programmes that promotes a bad learning environment? If that is the case, how can it be changed?

The changes in scientific computing courses will be permanent and further development will be performed, also when the project is formally completed.

References

Albinsson S., Andersson H., Dervisevic D., Sjöstedt H., Genus och programmering, Institutionen för Datavetenskap, Linköpings universitet, Dec. 2002

Biggs J, Teaching for Quality Learning at University, 2nd ed., Open University Press, 2003.