

Reviews and Findings on Implementing Active Learning in a Large Class Environment

for Mechatronics and Computer Science students

M.Sc. Tze Ying Sim

Department of Embedded System
Faculty of Electrical Engineering and Computer Science
University of Kassel (Germany)

Prof. Dr.-Ing Birgit Vogel-Heuser

Chair of Information Technology in Mechanical Engineering
Faculty of Mechanical Engineering
Technical University of Munich (Germany)

Abstract— This paper tackled the issues of different learning preferences for Computer Science and Mechatronics students in a large class environment by using different active learning methodologies. Apart from implementing active learning methods in the class, another idea present here is coupling with another course. The efforts required and the effectiveness for these methods are being presented, with this the suitable methods can be selected depending on the participants and the resources available.

Keywords—Computer Science education, Education, Engineering education, Mechatronics

I. INTRODUCTION

The class size in the universities starts to grow when higher education ceases to be “only for the elites” but also “for the masses” [1]. The most common pedagogy form for large class setting is the lecture setting. A traditional lecture takes place when a lecturer recites or pass on the knowledge in verbal form while the students take note [2]. Among the challenges in a large class settings are it is more difficult to control the class behavior [3], to follow up on students’ understanding, and to encourage students’ participation in the class. In order to overcome these problems, different methods under the term “Active Learning” were introduced. Active learning means the students not only listen and take notes in the class but they also have the opportunity to participate actively in the class [4]. Among the methods implemented in active learning include informal group learning, formal group learning, problem based learning, group work, team-teaching, cold calling, in-class demonstration, muddiest card, flash card, concept test, evaluation form, pause method, laboratory work, and learn management system [2,5,6,7,8]. The different methods mentioned above can be divided into two categories, those that can be implemented in the class and those to be implemented outside the class (see Table I).

Implementation of active learning methodologies requires more effort and resources. As described by Bonwell and Sutherland [5], among the challenges include the decrease of total covered contents, the increase of preparation effort, and the lack of materials and resources. Active learning can be

implemented in both large and small class. The size definition used in this paper is as described by Kuo [9]. A small class has less than 25 students, whereas a large class has more than 75 students.

This paper focuses on the introduction of different active

TABLE I. IN CLASS AND OUT OF CLASS ACTIVE LEARNING METHODS

Methods	In Class Implementation	Out of Class Implementation
Informal / Formal group learning	X	
Problem based learning	X	X
Group work	X	X
Team-Teaching	X	
Cold Calling	X	
In Class Demonstration	X	
Muddiest card	X	
Flash card	X	
Concept test	X	
Evaluation form	X	
Pause method	X	
Laboratory work		X
Learn management system / Internet		X

learning methods implemented in the course for Mechatronics and Computer Science students in a large class environment with limited resources. Section 2 introduces the course content, the settings of the course and the challenges faced. Section 3 to 5 present the methods implemented in the class outside the class but within the frame of the course, and outside the class through coupling with another course. Section 6 to 8 discuss on the findings for the different methods implemented. The last section concludes this paper by comparing the different methods to its impacts and effort needed for implementation.

II. COURSE INTRODUCTION

A. Course Content

The course Embedded System 1 with course code FB16-6951 (ES1) and Embedded System 2 with course code FB16-6952 (ES2) were introduced at the Faculty of Computer Science and Engineering, University Kassel [26] in summer

TABLE II. COMPARISON OF RESEARCH WITH DISCIPLINES, CLASS SIZE AND METHODS IMPLEMENTED

Item	Source	Disciplines	Disciplines	Contributing Dept.	# of Students	Lab work	Remote Lab/ LMS	Traditional Lecture	Group work	Project with External	Active Lecture	Tutorial	Muddiest card	Flash Card	
1	[10] (Murata & Stern, 1993)	Mechatronics	1	1	M	X	X	-	-	-	-	-	-	-	
2	[11] (Cowan, Ewell, & McConnell, 1995)	Religion, English	2	2	M	X	-	X	-	-	-	-	-	-	
3	[2] (Mills-Jones, 1999)	Computer Science, Business	2	3	L	-	-	X	X	X	X	S	-	-	
4	[12] (MacGregor, 2000)	English, Philosophy, Sociology	2	3	L	-	-	-	X	-	-	-	-	-	
5	[12] (MacGregor, 2000)	English, US History	2	2	L	-	-	-	X	-	-	-	-	-	
6	[13] (Grimheden & Mats Hanson, 2001)	Mechatronics	1	1	-	X	-	-	-	-	-	-	-	-	
7	[14] (Wikander et al., 2001)	Mechatronics	1	1	-	-	-	-	X	-	-	-	-	-	
8	[15] (Tomizuka, 2002)	Mechatronics	1	1	-	X	-	-	-	-	-	-	-	-	
9	[5] (Hall, Waitz, Brodeur, Soderholm, & Nasr, 2002)	Engineering	1	1	M	-	-	-	-	-	X	S	X	X	
10	[16] (Cooper, Carlisle, Gibbs, & Watkins, 2002)	Medical	1	3	-	-	-	-	X	X	X	S	-	-	
11	[17] (Sargison, Bullen & McCulloch, 2002)	Engineering	4	4	L	-	-	-	X	-	-	-	-	-	
12	[18] (Shooter & McNeill, 2002)	Mechatronics	1	2	-	X	-	-	X	-	X	-	-	-	
13	[19] (Erbe & Burns, 2003)	Mechatronics	1	2	-	X	X	-	-	-	-	-	-	-	
14	[20] (Amey & Brown, 2005)	Sociology	5	5	-	-	-	-	-	X	-	-	-	-	
15	[7] (Smith, Sheppard & Johnson, 2005)	General	-	-	S	-	-	-	X	-	-	-	-	-	
16	[21] (Graber & Pionke, 2006)	Engineering	1	2	M	-	-	-	X	-	X	-	-	-	
17	[22] (James, 2007)	General	-	-	M	-	X	-	X	-	-	X	-	-	
18	[23] (Pears & Daniels, 2007)	Engineering	1	1	S	-	-	-	-	-	-	-	-	-	
19	[24] (Sommers & Engbretson, 2007)	Engineering	1	1	S	-	-	-	X	-	-	-	-	-	
20	[25] (Mitusi, Kambe, Satoshi Endo & Koizumi, 2008)	Engineering	1	1	S	X	-	-	X	-	-	-	-	-	
21	University of Kassel	Mechatronics, Computer Science	2	1	L	X	-	X	X	-	X	M	X	-	
	Total for Implemented Methods						8	3	3	13	3	6	5	2	1

Legend: S- Small; M- Medium; L- Large; X-Implemented

semester 2006 and winter semester 2006/07 respectively. These courses are compulsory for Mechatronics students and Computer Science students who are in their third and fourth semesters. The objective of this course is to train the students

to think independently and be able to work in team when developing an embedded system. An embedded system is an embedded hardware/software system that regulates a physical device by sending control signals to actuators in reaction to

input signals provided by its users and by sensors capturing the relevant state parameters of the system [27]. The ES1 subjects focus on the development of microprocessors. Among the subjects covered in ES1 are Logic and Gates, Computer Architecture, Scheduling, Programming Languages – PEARL, Assembly Language and VHDL, and Bus System. The ES2 subjects focus on the development lifecycle of an embedded system [28]. The subjects in this course include Requirements Engineering, Modeling Methods, Automation Technology (with focus on hardware and architecture), Programming Language – IEC 61131-3, and Verification, Validation and Test.

Both these courses are courses with two semester hours a week. One semester hour is 45 minutes. The lecture and the exercise sessions take turn in almost alternate weeks. All sessions last for 90 minutes. There are no prepared lecture notes but the students are able to download the presentation slides from the professor a week before the respective lecture. The week before the exercise session, the exercise questions are uploaded to the internet. This enables the students to work on the questions prior to the exercise sessions.

B. Course Challenges

The first challenge is the discipline diversity of the participants in the courses. Averagely, 66% of the course participants are Computer Science students and another 34% are Mechatronics students. The participants from Electrical Engineering students are negligible. They are less than 1 % if any. These students have different cognitive minds and response to the different methods differently. The students understand the term modelling differently. During a question answer in the class, the Mechatronic students relate modelling to the modelling of a physical system, whereas the Computer Science students relate it to the design of a software system. According to Grimhenden and Hanson [13], Mechatronics students prefer practical work. As shown in Table II, the most implemented active learning method for Mechatronics students are group work, where the students develop a physical system. They are conducted in groups of two or three.

Secondly, the class size for both the courses are large. The average participants for ES1 is 100 students, whereas ES2 is 80 students. The implementation of active learning in a large class environment will be more challenging. Arias and Walker [29] argued that the small class students tend to fare better than those from large classes do. The research undertaken to implement active learning in large class environment [2,5,21,25] showed that much resources are required. This leads to the third challenge, the limited human resources.

The third challenge is the limited human resources. Interdisciplinary courses are courses where more than one disciplines are taken into considerations [30]. The course discussed in this paper is considered as interdisciplinary course [31]. From the research conducted (see Table II) the number of the contributing departments for interdisciplinary course is usually more than the number of disciplines. This also means that at least a professor and teaching staff from each department are available to conduct the course. As shown in item 21 in Table II, the course consisted of students from two

different disciplines – the Mechatronics students and the Computer Science students. However, unlike most of the research done, where the number of contributing departments is more or same as the disciplines involved, here, the opposite happens. The number of disciplines is two, whereas the contributing department is only one. A professor and a teaching assistant are responsible for this course. The ratio between teaching resources to students is 1 to 80.

The three challenges faced in this course will be discussed overall. Different methods implemented in the course of 5 semesters to cater to the diversity of the class, to overcome the barriers of large class environment and to make full use of the limited resources available will be presented. The implemented methods will be addressed in three categories, namely the methods implemented

- in the class,
- partially outside the class but within the frame of the course, and
- outside the class as coupled with another course.

III. METHODS – IN THE CLASS

A. Methods implemented during the Lecture

In order to maintain the students' attention, the lecturer very often includes students' participations in the lecture. Among the methods used in the lecture are

- in class demonstration with participations from students,
- asking questions to the students without embarrassing them, and
- presenting the idea visually on the board even though the class also uses slides presentation.

An example of an in class demonstration with students' participation is the join process in a Petri Net. In order to explain the tokens, places and transitions, four students participate in this demonstration. One student will hold the pen cover, and the other the pen. Both these students are the places with the pen and the cover as the tokens. Next, the pen and the cover will be passed to a student who plays the "transition" role. The student will put the pen and the cover together, and then pass the covered pen as a single token to the next student. Different role plays were conducted to explain the different process that can take place in a Petri Net.

As to asking questions in the class, the round robin method was implemented. If the student is not able to answer the question, the person next to this student will be next in row to help him/her out. Not much time is spent waiting to avoid embarrassment. This method not only keeps the students involved but also awake, as the class starts at 8:15 in the morning.

Even though the information is available in the handouts for the students, it helps to draw the students' attention when this information is once more presented on the board. For

example drawing a microprocessor part by part, and explain the functions of each component as they are drawn on the board.

B. Methods implemented during the Exercise

During the exercise session, which takes place almost alternate weeks, students are requested to work out the exercise problems on the board. However, the response had not always been positive. A few minutes will be given for the students to discuss the problem with their neighbor if most of the students did not work out the questions before hand. They would then be requested to solve the problem on the black board in front of the class.

In order to encourage the class participations, the teaching assistant tried to remember as many names as possible and to call these names in the class. However, the number of names remembered is at most 15. It was also not exactly “fair” to repeat the few same names weeks after weeks. The second method used to encourage students participation was to have a name list and to pick a name from this list. However, some students will pretend as if they were absent, choose to remain silent and not proceed to work out the problem even though they are actually in the class. In summer semester 2009, the teaching assistant tried a new approach. The tool was a packet of facial tissue paper. The teaching assistant posed a question and threw the packet to the first student who was supposed to answer the question. The student then in turn can throw the packet of tissue to another student for the next question. This method had been the most welcomed method as compared to the first two. This also brought some fun factors into the class.

C. Bench Scale Equipment as Example

Throughout the lecture of ES2, bench scale equipment was introduced. The purpose is to use a consistent example during the lecture and exercise, with the idea that this would help the students’ understanding on the course content. The bench scale equipment is a stamping and sorting machine. The equipment can be divided to four modules, namely the storage module, the

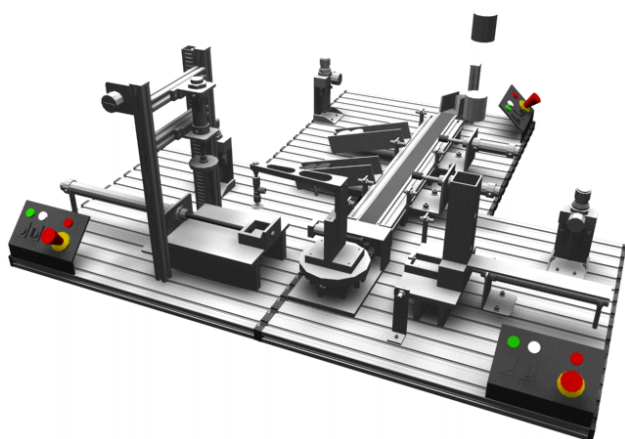


Figure 1. Bench scale equipment with four modules

crane module, the stamping module, and the sorting module. The students were taught how the sensors and the actuators function. The storage module is the simplest module where a

tactile sensor will sense if there is a work-piece that needs to be pushed out. The crane will pick the work-piece and send it to the stamping module and than later to the sorting module. Examples for requirements engineering, system modeling, program development, and testing was based on this equipment when possible.

D. Pop Quizzes

Pop quizzes are impromptu quiz conducted at the beginning of the class. The purpose of pop quizzes is to encourage students’ attendance and to have an in between overview on how much did the students understood in the previous lectures. Two or three pop quizzes will be conducted through out the semesters. Fifteen minutes were given to the students to finish up the quizzes. The motivation for participating in the pop quizzes is the possibility to include pop quizzes’ results in the final result of the course. The students who participated in pop quizzes can at most increase their marks by 15%.

IV. METHODS – PARTIALLY OUTSIDE THE CLASS

A. Group Work with Partial Life Cycle Development

As the content of ES2 focuses on the development of an Embedded System, the idea is to allow the students to go through the different life cycle phases using group work. In summer semester 2007, the students were divided into groups of 10. Each group was responsible for a chapter covered in the course. As observed in Fig. 2, the students were divided into nine groups. In order to encourage communications from the different disciplines, it was a pre-requisite that the group should be 70% Computer science students and 30% Mechatronics students.

No specific tool was implemented here. The students can

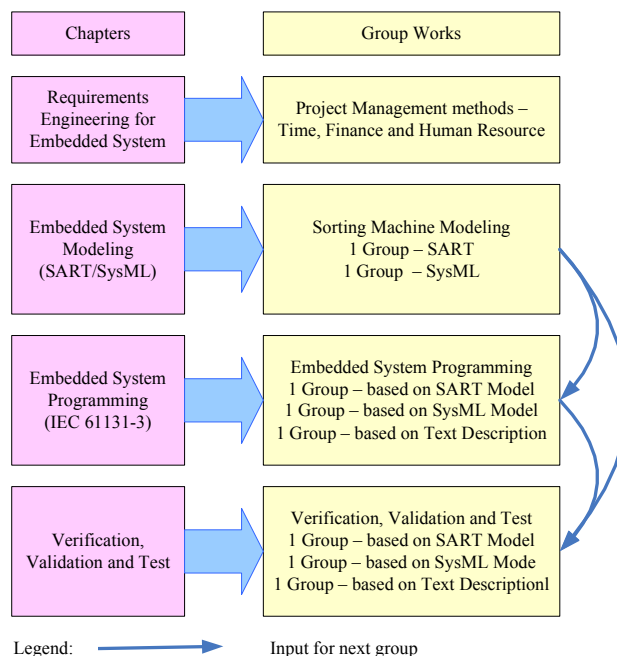


Figure 2. Bench scale equipment with four modules

select tools that they know or ordinary drawing tools to model the system in “Structured Analysis with Real Time Extension” (SART) [32] or “System Modeling Language” (SysML) [33]. The programming was conducted in the exercise session by writing the program either in structured text or in function block. The main objective here is to observe what the students can understand from the modeling diagram. The validation and test group will then compare if the functions mentioned in the modeling diagram were fulfilled in the program.

Each group was expected to deliver a written report and a 15 minutes presentation. Each group had a fix appointment with the teaching assistant two weeks before the presentation session. This ensured that the students are on the right track and well-prepared for the presentation. They can make extra appointments when necessary. The presentation was conducted during the exercise session. Therefore, during summer semester 2007, thirty minutes from the one and a half hour exercise was used for group works’ presentation.

B. Group Work with Whole Life Cycle Development

In summer semester 2009, a different concept of group work was implemented. The students were again divided into groups of tens. This time instead of making it a pre-requisite, the students were encouraged to have a group with members from both disciplines. However, the Computer Science students grouped automatically with the Computer Science and the Mechatronics with the Mechatronics. This shows that students are still more comfortable to work with the people they are familiar. In order to encourage interdisciplinary communication and group work, the group distribution with students from different disciplines needs to be intentional.

TABLE I DIFFERENT ASSIGNMENTS IN GROUP WORK

Project Assignment	Description
Part A: Requirements Engineering	1) Decide on the job description of each member 2) Requirement document for the system
Part B: System Modeling (SART)	1) System modeling document
Part C: System Modeling (SysML)	1) Report on modules that can be reuse
Part D: System Development (Own Group)	1) System simulation 2) Report on document evaluation from part A to C.
Part E: System Development (Another Group)	1) System simulation 2) Report on document evaluation from part A to C.
Part F: Test and Validation	1) Test cases and test result for own system
Part G: Group Work Evaluation	1) Evaluation report

The same bench scale equipment was used as the foundation for the group work. Instead of working on one of the lifecycle phases, the same group had to start from requirements gathering and ends at verification, validation and test. Using the different actuators and sensors introduced in the class, the students were requested to design a system on their own. They needed to produce a requirement document,

explaining the system description. Next, they would need to design the system using SART, and observe on any possible reusable modules and present it using block definition diagram and internal block diagrams from SysML. After that, they would develop a simulation for the system using TrySim [34]. Apart from developing their own system, the group received a copy of system description and system modeling from another group, and they should implement the system from another group as well. Table III describes the different assignments in this group work.

For assignment parts A to C, the students were free to use any drawing tools that they are familiar. However, the students were also encouraged to attend an additional related course, which will be described in the following chapter, to learn the tools that they can use for the whole lifecycle process. The simulation software needed for part D and E was presented during the exercise and was also taught in this coupled course. Unlike the former group work, no in-class presentation was arranged. The students only need to do one presentation at the end of the group work

V. METHOD – COUPLING WITH ANOTHER COURSE

A. Loosely coupled Software Tools

In summer semester 2007, the course Software Tools with course code FB16-6959 (SWT) was introduced. This was a two semester hours week course, meaning the students needed to attend a 90 minutes class/lab session each week. The objective of this course is to introduce different software tools that can assist the engineering process. These different software tools can be mapped into the different development life cycle of an embedded system. Therefore, the coupling is done by first teaching the theory in ES2 lecture and then practical in SWT. In summer semester 2007, the tools introduced in SWT were IBM RequisitePro Software [35] for Requirements Engineering, Visual Object Net++ [36] for Petri net, Enterprise Architect [37] for UML [33], and TrySim [34] for IEC 61131-3 programming. The students worked in pairs to solve the problem using the different tools.

Before the semester began, the schedule for SWT and ES2 were coordinated in such a way that the practical work in SWT should only be conducted after the subjects were taught in ES2. SWT also used the same bench scale equipment as case study for the class. However, the functions of the equipment were modified. The students were requested to sort the work-pieces differently as those in ES2. The students were graded base on the practical work delivered throughout the whole course. There is no written test for SWT.

B. Closely coupled Software Tools

Through the SWT course evaluation from summer semester 2007, we learnt a few lessons. The course arrangement scheduled during the beginning of the course was not optimal as there were times when ES2 lectures took longer than expected to finish or the subjects were rescheduled due to unforeseen circumstances. ES2 taught the students SysML but in SWT, UML was implemented. As a result, a number of students were confused as both mark-up languages share a

number of similar points though they are different. The students who participated in both ES2 and SWT claimed that the same lab bench equipment with different software functions confused them.

For summer semester 2008, a few different methods were implemented. Firstly, the first three weeks of SWT were used for two lectures on lifecycle process and the selection of tools, as well as group organisation. This gave SWT a three weeks' buffer if ES2 schedule is delayed due to unforeseen circumstances. Secondly, instead of teaching UML, the students were taught SysML. This would avoid any confusion that might occur. SysML was chosen as SysML includes modelling of system architecture and behaviour.. Thirdly, throughout the whole ES2 and SWT classes, the students were reminded that the same hardware can have different software functions. The same lab bench equipment was used for ES2 and SWT but their software functions are different.

The tool IBM Requisite Pro Software excluded from the course Software Tools for summer semester 2008. This is due to two reasons. Firstly, from past experience, the students needed at least three weeks (three practical sessions) to be familiarized with the tool. Secondly, we received six sets of network kit where the students can configure the connections between computer and controller and test download the programs for the controllers. The network controller kit was coupled with the Bus System chapter that was brought forward from ES1.

VI. BASICS FOR ANALYSIS

A. Data Source

There are three batches of students involved in this study. As shown in Table IVA, they are batch 0607, 0708, and 0809. They each stands for the semesters where ES1 takes place. Batch 0607 includes ES1 in winter semester 2006/07 (ES1 WS0607) and ES2 in summer semester 2007 (ES2 SS2007). The same applies to batch 0708 and 0809.

TABLE IIA METHODS IMPLEMENTED IN THE DIFFERENT SEMESTERS

Batch	0607		0708		0809	
Implemented Methods	WS0607	SS2007	WS0708	SS2008	WS0809	SS2009
Pop Quiz	--	--	X	X	X	--
Group Work	--	X	--	--	--	X
Software Tools	--	X	--	X	--	X

TABLE IVB DATA SOURCE FOR ANALYSIS

Data Source	WS0607	SS2007	WS0708	SS2008	WS0809	SS2009
Course Exam Result	X	X	X	X	X	X
School Exam Result	X	X	X	X	--	--
Course Evaluation	--	X	--	X	X	X
Group work Evaluation	--	--	--	--	--	X

Three main data source were used to analyze the effectiveness of the different methods (see Table IVB). They are the course evaluation conducted at the end of the semesters, the group work evaluation for summer semester 2008, the group work, pop quizzes and exam results from the students. The data were collected across five semesters, from summer semester 2007 to summer semester 2009.

When comparing the effectiveness of different methods in a particular semester, no extra point grade will be used. The students who participated in the pop quizzes or group work had the opportunity to collect extra 15% points. A student who did not participate in pop quizzes nor group work can have 100% full score but a student who did can have 115% for full score. For example, a student earned 10% extra point from the pop quizzes or group work and in the final exam this student scored 70%. The no extra point (NEP) percentage is 70%, whereas the extra point percentage (WEP) is 80% (10% + 70%). "No extra point grades" are the grades according to the NEP percentage and "extra point grades" are the grades according to the WEP percentage.

In order to have fair comparison, the exam results for other courses from the same batch of students for that particular semester were taken into consideration. This will function as the control data for comparison with ES1 and ES2. Even though it is not compulsory for the students to follow the semester plan, from the course evaluation, we discovered that almost 57% of the Computer Science students and 97% of the Mechatronics students do follow the semesters plan. Therefore, it is reasonable to use the overall grade from the semester as control data. The ES1 and ES2 grades used here are the final grades that include the extra points accumulated through pop quizzes or group work. As the control data for batch 0809 is still in process, only control data for batch 0607 and 0708 are available.

The grading used in Germany is from scale 1 to 5, with 1 as the best grade and 5 as the poorest grade. The grades used for comparison are the average grades, unless mentioned otherwise. In the following, control grade means the average grade for control data and course grade means the average for course grade. 2-tail tests at $\alpha = 0.05$ are mainly used in the analysis, unless mentioned otherwise. The students' results from the different semesters fulfills the normal distribution.

B. Comparison with Control Data

TABLE III COMPARISON OF STUDENTS IN ES1 AND ES2 TO CONTROL DATA

Computer Science	Control Data			Course Data		
	N	Average	Std. Dev.	N	Average	Std. Dev.
WS0607	298	3,436	1,2232	71	3,515	1,3920
SS2007	139	2,900	1,0476	50	2,884	1,0223
WS0708	258	3,322	1,2917	52	3,404	1,0066
SS2008	147	3,354	1,1637	45	2,984	1,4327
WS0809	--	--	--	50	2,946	1,3577
SS2009	--	--	--	56	2,850	1,2262

Mechatronics	Control Data			Course Data		
	N	Average	Std. Dev.	N	Average	Std. Dev.
WS0607	78	3,082	1,132	25	3,932	1,3247
SS2007	91	2,924	1,1261	24	3,054	0,9302
WS0708	140	2,800	1,1000	33	3,036	1,3583
SS2008	149	2,951	1,0771	27	2,419	1,4531
WS0809	--	--	--	29	3,007	1,5597
SS2009	--	--	--	24	1,662	0,7523

Computer Science

The average grade for the course and control grade improved significantly for batch 0607. The average grade for ES1 WS0708 was 3.404 and it improved significantly (with $p = 0.10$) in ES2 SS 2007, with 2.984 (see Table V). There was no significant improvement for the control grade for batch 0708.

The course grade for ES2 SS2008 is significantly better than control grade. Similarly, to batch 0708, there is also no significant improvement for the course grade in batch 0809. Therefore, the following two conclusions can be made for Computer Science students:

- The students performed better in ES2 as compared to ES1 for batch 0607, 0708, and 0809
- The methods implemented in ES2 SS2008 have more positive impacts than the methods implemented in ES2 SS2007 and ES2 SS2009.

Mechatronics

There is no significant improvement between the semesters for the control data. The average grade for the control data was 3.082 for winter semester 2006/07, 2.924 for summer semester 2007, 2.800 for winter semester 2007/08, and 2.951 for summer semester 2008. Unlike batch 0607, the control data's average grade for summer semester 2008 was poorer than the winter semester 2007/08. As for ES1 and ES2 average grade, the average grade for batch 0708 shows no significant difference but the grade for batch 0607 and 0809 shows significant improvement. The average grades for batch 0607, 0708 and 0809 are as follow: 3.932 and 3.054; 3.036 and 2.419; and 3.007 and 1.662. The following two conclusions can be made for Mechatronics students:

- Just like the Computer Science students, the Mechatronics performed better in ES2 as compared to ES1 for batch 0607, 0708, and 0809
- Unlike the Computer Science students, the methods implemented in ES2 SS2008 were least significant for the Mechatronics students.

VII. ANALYSIS – METHODS IN THE CLASS

A. Methods in the Lecture and Exercise

The observation for the methods implemented in the lecture and exercise are done through the feedbacks in the course evaluation form and the students' participation in the class. We conclude that the question and answer, demonstration in the class and students having the opportunity to work out the problem in the class have positive impact.

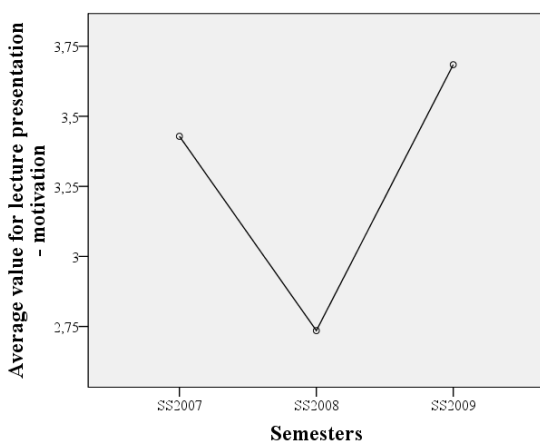


Figure 3. Average value for motivation in the lecture

The communication with the students in the class has more influences than the slides, in attracting students' attention and interest to the contents of the courses. The students who participated in ES1 and ES2 commented that there were too many slides. As comparison, other courses in the faculty provide lecture notes and as a result have simpler lecture slides. As there were no prepared lecture notes for ES1 and ES2, the slides had the function to provide the additional related information that the students need. Efforts to reorganize the slides took place in ES2 SS2008 but the students still find the slides and the course content overwhelming. In ES2 SS2009, another lecturer uses the same slide for lecture. However, unlike the previous lecturer, this lecturer had almost no communication with the students. Students were not questioned in the class, short pauses were made to ask if they have any question but the class remains silent most of the time. Using the evaluation feedback for ES2 SS2007, ES2 SS2008 and ES2 SS2009, there was a significant difference for the motivation in lecture for the different students. The motivation for SS2008 was the best, followed by SS2007 and SS2009 (see figure 3). Even though the understanding on the course lecture also follows the same pattern but the average did not differ significantly when using one-way ANOVA test. The attendance for ES2 SS2009 also dropped drastically when compared to other semesters. There were time when only 20 out of 80 students attended lecture.

B. Implementation of Bench Scale Equipment

The same bench scale equipment was implemented for both ES2 and SWT. This saved time in explaining how the different hardware functions and the slides prepared can also be used for both courses. There were few comments in the evaluation form saying that the bench lab equipment should be introduced earlier. In the group work evaluation for ES2 SS2009, 65.9% of the students graded the bench lab equipment as useful or very useful in helping them to understand the course content. Another 20.9% was neutral and only 13.2% graded it as not useful and not useful at all.

From the teaching assistant point of view, using the same bench lab equipment saved preparation time in terms of preparation of lecture slides, exercise slides, and helped students to understand what can be implemented in the group work. This gave the students the time to gain needed understanding on the selected hardware structures and the respective software implementations. However, as the same hardware with different software might cause confusion to the students, it is important to stress this frequently in the class. No students complained about this issue in the latest evaluation.

C. Implementation of Pop Quizzes

Generally, students who participated in pop quizzes fared better than those who did not. Except for winter semester 2007/08, the students who participated in pop quizzes scored significantly better. The expected advantage of the students' participation in the pop quizzes is to help students familiarize with the exam questions' pattern, besides having a more consistent revision on important course contents. Another possible advantage might be students who sat for pop quizzes are able to estimate time needed per question better.

TABLE IV COMPARISON OF PARTICIPANTS AND NON PARTICIPANTS FOR POP QUIZZES

Pop Quizzes			Participants			Non-Participant			Significance
			N	Average	Std. Dev.	N	Average	Std. Dev.	
Computer Science	ES1	WS0708	52	3,506	0,9662	20	4,070	0,9251	0,000
	ES2	SS 2008	25	2,692	1,1622	20	3,371	1,1829	0,419
	ES1	WS0809	29	3,069	1,3696	21	3,371	1,1829	0,419
Mechatronics	ES1	WS0708	30	3,100	1,1858	3	4,333	0,5774	0,088
	ES2	SS 2008	20	2,400	1,2703	7	3,857	0,8772	0,010
	ES1	WS0809	15	2,320	1,4047	14	4,114	0,9281	0,000

The average grade for students who took part in pop quizzes is significantly better in ES2 SS2008 but not in ES1 WS0708. The improvement of average grade for Computer Science students in batch 0708 who took part in pop quizzes is from 3.506 to 2.692, whereas the improvement for Mechatronics students is from 3.100 to 2.400 (see Table VI). This average grade improvement is not observed in batch 0809 where pop quiz is implemented in ES1 and group work in ES2. One reason might be that after two semesters of pop quizzes, the students can understand and answer the exam questions better.

There is a significant difference when comparing the attendance for group works that were implemented in ES2 SS2008 and ES2 SS2009, with the attendance for pop quizzes that were implemented in ES2 SS2008 and ES1 WS0809. The attendance when pop quizzes were conducted is better than the group work.

The implementation of pop quizzes seems to be the most effective methods in preparing the students to answer the exam questions. The extra effort needed to implement this are the time for the pop quizzes preparation, quiz marking, and the discussions with the students after each pop quiz. Averagely, 4 hours were needed to prepare a 20 minutes pop quiz, 30 minutes to correct the forty quizzes, 30 minutes to discuss the quiz questions and answers with the students, and another 30 minutes for extra organization works. Each pop quizzes is estimated be an additional 5 to 6 hours work for the teaching assistant.

VIII. ANALYSIS – METHODS PARTIALLY OUTSIDE THE CLASS AND COUPLING WITH OTHER COURSE

A. Comparison of Partial Life Cycle with Whole Life Cycle Development Group Work

Out of the 74 students in summer semester 2007, 57 took part in the partial life cycle development group work. There is a significant difference between the average grades. Students who participated in the group work scored 2.746 as average grade, whereas those did not scored 3.588. The average grade for those who participated in whole life cycle development group work in summer semester 2009 is also significantly better than those did not. The students who participated in the group work scored average grade 2.310 while those did not only scored 3.055. 51 from the 80 students participated in the group work in summer semester 2009.

Table VII separates the Computer Science students from the Mechatronics. It is interesting to note that the Computer Science students and the Mechatronics students have almost

the opposite effects. The average grade for Computer Science students who participated in the partial group work was significantly better than those who did not, and the average grade for those who participated in the whole life cycle group project was significantly poorer than those who did not. The opposite is true for the Mechatronics students. One clear conclusion that can be drawn here is whole life cycle group work strongly influence the Mechatronics students, this supports the idea from Grimhenden and Hanson [13].

As compared to the partial group work in summer semester 2007, the Computer Science students' participation was lower in the whole life cycle group work in summer semester 2009. One reason given by the students who did not participate was the group work is too time-consuming. The partial life cycle group work requires the whole group work on one phase and the work can be completed in two to three meetings. The whole life cycle group work, on the other hand, will require more co-ordination effort and students might need to be involved in more than one phase of the life cycle. The Computer Science students commented that the extra 15% point is not worth the effort. They would prefer to use the time to revise on their own.

The effort needed to conduct partial life cycle group work was less than the whole life cycle group work. Only one to two appointments with the students were needed for the partial life cycle group work, where as the whole life cycle group work required at least four appointments, where each appointment had a duration of 30 minutes. The group work presentation for the partial life cycle group work was conducted in the class, where as extra appointment during the semester break was used for the group work presentation. Almost eight extra hours were required to conduct partial life cycle group work but at the expense of thirty minutes of the exercise. The whole life cycle group work required at least two to three hours extra time for each group. As there are eight groups, this amounted to an additional time of twenty four hours to implement the whole life cycle group work.

B. Comparison of loosely coupled and closely coupled Software Tools

Generally, loosely coupled SWT did not have any significant impact on the students who participated in it. During summer semester 2007, 32 from 74 students participated in SWT. However, closely coupled SWT made significant impact on ES2 grade for the students who participated in it. The average grade for students who participated in SWT in summer semester 2008 was 2.28 as compared to 3.05 for those who did not. The result for summer semester 2009 also supported the observation where closely coupled SWT has more impact and the approach had significant influence on the participants. The students who

TABLE V COMPARISON OF PARTICIPANTS AND NON PARTICIPANTS FOR SOFTWARE TOOLS

Software Tools			Participants			Non-Participant			Significance
			N	Average	Std. Dev.	N	Average	Std. Dev.	
Computer Science	LC SWT	SS 2007	10	3,200	0,7409	40	3,238	1,0081	0,913
	CC SWT	SS 2008	2	3,150	2,6163	43	3,312	1,2236	0,862
	CC SWT	SS 2009	7	3,857	1,0830	49	3,286	1,2057	0,241
Mechatronics	LC SWT	SS 2007	22	3,350	1,0541	2	4,500	0,7071	0,149
	CC SWT	SS 2008	24	2,583	1,2761	3	4,333	0,5774	0,029
	CC SWT	SS 2009	22	2,532	1,2029	2	5,000	0,0000	0,009

participated in SWT had 2.021 average grades, whereas those did not, only have 2.763.

TABLE VII COMPARISON OF PARTICIPANTS AND NON PARTICIPANTS FOR GROUP WORK

Group Work			Participants			Non-Participant			Significance
			N	Average	Std. Dev.	N	Average	Std. Dev.	
Computer Science	GW-Partial	SS 2007	41	3.029	0.8878	9	4.144	0.6966	0.001
	GW-Whole	SS 2009	34	3.659	1.0252	22	2.891	1.3140	0.018
Mechatronics	GW-Partial	SS 2007	16	3.544	0.9906	8	3.250	1.2581	0.537
	GW-Whole	SS 2009	22	2.532	1.2029	2	5.000	0.0000	0.009

Table VIII separates the Computer Science students from the Mechatronics. A few trends can be observed here. Firstly, the participation from Mechatronics students was higher as it was a compulsory course for them. Secondly, there was no significant difference if the Computer Science students participated in SWT or not. Thirdly, closely coupled SWT positively influenced the Mechatronic students' result. The ES2 result between those who took part in closely coupled SWT was significantly better than those who did not.

Generally, students who participated in SWT had better average grades than those who did not. The average grades for loosely coupled SWT were poorer than the control data. However, tightly coupled SWT's average for both Computer Science and Mechatronics was better than the control data average.

IX. CONCLUSION

In this paper different active learning methods for a large interdisciplinary class were introduced analyzed. There were three challenges presented in section IIB, namely – students from different disciplines have different cognitive mindset, it is difficult to engage students in a large class setting, and the limited teaching resources.

To the point different cognitive mind set the following points are summarized:

- The pop quizzes benefitted both Computer Science students and Mechatronics students.
- The Computer Science students prefers partial life cycle group work as it requires less time and effort as compare to the whole life cycle group work.
- The Mechatronics students on the other hand are willing to put in the effort required by whole life cycle group work and benefitted more as compared to the Computer Science students.

In order to engage the students in a large class setting, it is important to involve the students. Simple methods like round robin questioning, in class demonstration and “throwing tissue pack” should be implemented. Preparing good nice slides or explaining the slides in details is not sufficient to draw the students' attention.

In order to reduce the effort required to conduct the course, implementation of the same bench scale equipment for the different subjects is recommended. Apart from this, coupling with another course is another possibility to win new resources for the course.

Table IX summarizes the effort needed and the effectiveness of the different methods. “Pop quizzes” is a winner, followed by closely coupled software tools. Pop quizzes prepare the students to answer the exam questions for the final exam. However, one weakness with pop quizzes is the students do not get to implement the theory. In SWT, the students are able to implement the theory they learnt in ES2 in a software tool. The effort required to closely couple with another course is not high in term of man-hours but the two co-

TABLE VI EFFORTS REQUIRED AND EFFECTIVENESS OF THE DIFFERENT METHODS

Method	Effort	Effect	
		CS	M
Round Robin questioning	Low	+	
In class demonstration with students' participation	Low	+	
Throwing tissue pack to students	Low	+	
Implementation of bench scale equipment	Low	+	
Pop quizzes	4 hrs + n(30 mins)	++	++
Partial life cycle group work	2 hrs + n(8 hrs)	++	-
Whole life cycle group work	2 hrs + n(24 hrs)	--	++
Loosely coupled software tools	2 hrs	-	+
Closely coupled software tools	3 hrs	o	++

++ significantly good

+ good

o no visible impact

- poor

-- significantly poor

n = number of group with 40 students

coordinators need to be in constantly exchange feedback concerning what was being taught and what difficulties faced by the students. Depending on the available resources and the participants in the class, one can select the most suitable method for the class using Table IX.

For teaching Embedded System in a large class environment, especially with participants from Computer Science and Mechatronics students, it is recommended to implement pop quizzes and closely coupled software tools. Pop quizzes significantly benefits both Computer Science and Mechatronics students in their exam. Closely coupled SWT provides the opportunity to implement the theory learnt in ES1, and has more impact than loosely coupled SWT.

REFERENCES

- [1] E.L. Boyer, *Scholarship Reconsidered: Priorities of the Professoriate*, 1990.
- [2] A. Mills-Jones, “Active Learning in IS Education: Choosing Effective Strategies for Teaching Large Classes in Higher Education,” 1999, pp. 622-634.
- [3] E. Carbone, “Students Behaving Badly in Large Classes,” *New Directions for Teaching and Learning*, vol. 1999, 1999, pp. 35-43.
- [4] C.C. Bonwell and J.A. Eison, *Active Learning: Creating Excitement in the Classroom*, Washington DC.: ERIC Clearinghouse on Higher Education; George Washington University, 1991.
- [5] S.R. Hall, I. Waitz, D.R. Brodeur, D.H. Soderholm, and R. Nasr, “Adoption Of Active Learning In A Lecture-Based Engineering Class,” Boston, MA: 2002, pp. T2A9-T2A15.
- [6] C.C. Bonwell and T.E. Sutherland, “The Active Learning Continuum: Choosing Activities to Engage Students in the Classroom,” *New Directions for Teaching and Learning*, vol. 1996, 1996, pp. 3-16.
- [7] K.A. Smith, S.D. Sheppard, D.W. Johnson, and R.T. Johnson, “Pedagogies of Engagement: Classroom-Based Practices,” *Journal of Engineering Education*, vol. 94, 2003, pp. 87-101.
- [8] J.R. Davis, *Interdisciplinary Courses and Team Teaching*, American Council on Education and The Oryx Press, 1995.
- [9] W. Kuo, “Editorial: How Reliable is Teaching Evaluation? The

- Relationship of Class Size to Teaching Evaluation Scores," *IEEE Transactions on Reliability*, vol. 56, 2007, pp. 178-181.
- [10] S. Murata and S. Stern, "Technology Education in Japan," vol. 5, 1993.
- [11] M.A. Cowan, B.C. Ewell, and P. McConnell, "Creating Conversations An Experiment in Interdisciplinary Team Teaching," 1995.
- [12] J. MacGregor, "Restructuring Large Classes to Create Communities of Learners," *New*, vol. 2000, 2000, pp. 25-46.
- [13] M. Grimheden and M. Hanson, "What is Mechatronics? Proposing a Didactical Approach to Mechatronics," 2001.
- [14] J. Wikander, M. Torngren, and M. Hanson, "The science and education of mechatronics engineering," *Robotics & Automation Magazine, IEEE*, vol. 8, 2001, pp. 20-26.
- [15] M. Tomizuka, "Mechatronics: from the 20th to 21st century," *Control Engineering Practice*, vol. 10, Aug. 2002, pp. 877-886.
- [16] H. Cooper, C. Carlisle, T. Gibbs, and C. Watkins, "Developing and evidence base for interdisciplinary learning: a systematic review.," *Blackwell Science Ltd, Journal of Advanced Nursing*, vol. 35, 2002, pp. 228-237.
- [17] J. Sargison, F. Bullen, and J. McCulloch, "Development of a cross discipline, experiential based, flexible delivery unit," *International Conference on Engineering Education*, 2002, pp. 1-4.
- [18] S. Shooter and M. McNeill, "Interdisciplinary Collaborative Learning in Mechatronics at Bucknell University," *Journal of Engineering Education*, 2002, pp. 339-344.
- [19] H. Erbe and F.W. Burns, "Didactical Aspects of Mechatronics Education," *SICICA 2003- Intelligent Components and Instruments for Control Application*, Oxford: Elsevier Ltd., 2003.
- [20] M.J. Amey and D.F. Brown, "Academic Work: A Case Study of a University-Community Partnership," *New Directions for Teaching and Learning*, vol. 2005, 2005, pp. 23-35.
- [21] G.C. Graber and C.D. Pionke, "A Team-Taught Interdisciplinary Approach to Engineering Ethics," *Science and Engineering Ethics*, vol. 12, 2006, pp. 313-320.
- [22] R. James, "Six educational issues facing universities and academic leaders," May. 2007.
- [23] A. Pears and M. Daniels, "A national center for pedagogical development in technology education from a student oriented perspective," *Meeting the Growing Demand for Engineers and Their Educators 2010-2020 International Summit, 2007 IEEE*, 2007, pp. 1-8.
- [24] D.R. Sommers and G. Engbretson, "Teaching "Real-World" Design in the Classroom," *Meeting the Growing Demand for Engineers and Their Educators 2010-2020 International Summit, 2007 IEEE*, 2007, pp. 1-8.
- [25] H. Mitusi, H. Kambe, S. Endo, and H. Koizumi, "A Student Experiment Method for Learning the Basics of Embedded Software Development Including HW/SW Co-design," *22nd International Conference on Advanced Information Networking and Applications - Workshops*, IEEE Computer Society, 2008, pp. 31-37.
- [26] FB16, University Kassel, "Course handbook for Computer Science Bachelor Students for the Faculty of Computer Science and Electrical, University Kassel (Lehrveranstaltungshandbuch für den Bachelorstudiengang Informatik des Fachbereiches Elektrotechnik/Informatik der Universität Kassel)," Apr. 2006.
- [27] M. Broy, "Requirements Engineering for Embedded Systems," *FemSys'97*, 1997.
- [28] B. Vogel-Heuser, *Systems Software Engineering. Angewandte Methoden des Systementwurfs für Ingenieure*, Oldenbourg Wissensch.Vlg, 2003.
- [29] J. Arias and D.M. Walker, "Additional Evidence on the Relationship between Class Size and Student Performance," *Journal of Economic Education*, vol. Fall 2004, 2004, pp. 311-329.
- [30] M. Davies and M. Devlin, "Interdisciplinary higher education: Implications for teaching and learning," 2007.
- [31] T.Y. Sim and B. Vogel-Heuser, "Lessons learnt from Interdisciplinary Class between Mechatronics and Computer Science students," *International of Arts and Sciences Conferences*, 2009.
- [32] D.J. Hatley, P. Hruschka, and I. Pirbhai, *Process for System Architecture and Requirements Engineering*, Dorset House Publishing, 2000.
- [33] T. Weilkens, *System Engineering with SysML/UML*, dpunkt Verlag, 2006.
- [34] "Machine Simulation for PLC Programmers - TrySim," *Machine Simulation for PLC Programmers - TrySim*, 2009.
- [35] "IBM - Rational RequisitePro - Software," *IBM-Rational RequisitePro - Software*.
- [36] "Visual Object Net Analysis," *VON: Visual Object Net++*.
- [37] "Enterprise Architect - UML Design Tools and UML CASE tools for software development," *Enterprise Architect*.