Peer Learning with Lego Mindstorms

Ciarán Mc Goldrick[1], Meriel Huggard[2]

Abstract - **Reciprocal peer learning involves students learning from, and with, each other. This paper details a peer learning centred course where small teams of students design and develop a multifunctional robot using LEGO Mindstorms™. In particular, it describes how students were introduced to the concept of peer learning and outlines how the learning environment was managed and sustained. Particular emphasis is placed on acknowledging and rewarding peer collaboration as part of the assessment procedures, thus encouraging active student engagement with the peer learning process.**

Index Terms - Collaboration, Lego Mindstorms, Peer learning, Robotics.

INTRODUCTION

Peer learning is ubiquitous: most of the information and guidance we use to conduct our daily lives is obtained from our peers. For example, if someone is unfamiliar with a computer application they often seek basic information on the application's capabilities from a friend or colleague. This assistance may include a demonstration of how the application works. The peer is expected to be capable of answering any queries that may arise regarding the application or to suggest where answers to such queries may be obtained e.g. by a referral to another peer.

Peer learning is not only an informal process; for most students it is an integral part of their formal academic learning. They discuss the material they are learning, the learning style they are adopting and any problems they are encountering with their fellow students outside of the classroom setting [4]. Boud argues that `there is considerable benefit in taking what we know of the value of informal peer learning, making it explicit and using it more directly in the design and conduct of higher education courses' [5]. The course described herein fosters peer learning as a necessary and important part of the learning process inside the classroom.

Haller, Gallagher, Weldon and Felder observed [e7] that peer education typically occurs via two types of teaching-learning interactions: through *transfer-of-knowledge* sequences where one student acts as teacher and the other as pupil or through *collaborative sequences* where students work together with no clear role differentiation. They argue that peer learning can be facilitated by `introducing students to the two modes of teaching interaction so group members can effectively manage exchanges of knowledge in their work'. This approach is utilised in the course outlined below.

A key principle underpinning the introduction of peer learning is that the teamwork, leadership and decision making skills it cultivates are fundamental to the educational formation of engineers and computer scientists [1]. Over the past decade, many undergraduate degree programmes have been restructured to incorporate the fostering of student skills in these, and related, areas [2]. This paper describes how this approach has shaped the development of a second year undergraduate course on Computer Technology and details how a peer learning environment was created and assessed as part of this course.

To formalize peer learning, students are required to plan their learning activities, work together as a team and manage the assigned tasks within a given timeframe. Moreover, students have to share resources and engage in debate and critical reflection. The methods used to introduce students to the concepts of peer learning are described, together with the strategies used to manage and sustain the learning environment. The assessment procedures used place particular emphasis on acknowledging and rewarding peer collaboration, thus encouraging active student engagement in the peer learning process.

The module in question requires small teams of students to work on the design and development of a number of small robots using LEGO Mindstorms™. LEGO™ robots have been successfully incorporated into the undergraduate computer science curriculum across a wide variety of courses; from introductory programming [e1] and software design [e2] to artificial intelligence [e2] and neural networks [e4]. The laboratory experience described below is distinct because formalized peer learning is made an explicit, key part of the module. The students are evaluated on both the quality of the robots they produce and on their performance as a team. The following sections describe the design philosophy, experimental setup and evaluation methodology adopted for delivery of this module.

MODULE STRUCTURE

`Computer Technology' is a required course for second year undergraduate Information and Communication Technology (ICT) degree students. Class size has varied from 40 to 118 in the five years since the inception of the course. The course aims to provide the students with a strong foundation in a diverse range of Computer Science topics. In doing so it empowers students to make informed, considered decisions on technology related issues and, indeed, on ICT course options in future years. Lecture material is supplemented by tutorials

[1] Ciarán Mc Goldrick, Department of Computer Science, O'Reilly Institute, Trinity College, Dublin 2. Ireland, Ciaran.McGoldrick@cs.tcd.ie
[2] Meriel Huggard, Department of Computer Science, O'Reilly Institute, Trinity College, Dublin 2. Ireland., Meriel.Huggard@cs.tcd.ie

and by a highly integrated laboratory programme. This programme is aligned with the weekly lecture material to ensure students are benefiting from practical reinforcement of, and exposure to, the approaches, concepts and techniques addressed in formal lectures. To ensure maximum benefit for the students, and to facilitate small group interaction and learning, the class is subdivided into groups of 22 - 25 students for laboratory exercises. Each student attends one two-hour laboratory session per week. This requires considerable teaching assistant and demonstrator support: A dedicated teaching assistant (TA) and two demonstrators are present at every laboratory session. The Instructor attends each session to ensure that the required learning and developmental objectives are being attained. Moreover, he monitors group progress during each laboratory session and is available to meet students outside timetabled contact hours.

MODULE DESIGN PHILOSOPHY

The Computer Technology course is divided into a number of modules involving a mix of formal lectures and laboratory work. This paper describes one module taken by students on this course; this module seeks to reinforce learning in an enjoyable way whilst encouraging students to become independent, critical thinkers. Moreover, it provides small groups of students with the opportunity to learn how to develop and function as a coherent, effective group, cooperating with others to attain mutually beneficial goals.

Students are encouraged to view peer learning as both an essential and beneficial aspect of their future professional lives. By highlighting the relevance of cooperative, peer-learning skills, students view the use of peer learning within the module as a valuable experience. The individual effort required from each student is reduced if each group makes efficient use of their collective knowledge and skills.

On completion of the module each group of students is required to reflect on their performance as a group and how they have developed their intra- and inter-group collaboration skills. They should be more self-confident and have an enhanced ability to self and peer assess. Furthermore they should able to reason knowledgeably and logically about technical problems within their realm of expertise.

The learning objectives for this module are to:

- Refine problem identification and solution planning skills
- Plan, implement and manage a technical project
- Test and validate that a product satisfies given design criteria
- Develop an understanding of the peer learning process
- Practice and reflect on the nature of group collaboration
- Value and judge knowledge gained from their peers.

The module was designed using the methodology of `constructive alignment' proposed by Biggs [6]. The presentation structure and assessment criteria were specifically chosen to encourage students to engage in tasks which help them to attain the desired learning objectives. Hence, the activities carried out by the students are directly linked to the curriculum objectives for the course.

An initial, one hour session is used to establish a simple didatic contract between the students and the instructor. During this session the module objectives, technical tasks to be undertaken and assessment criteria are discussed and agreed. Peer learning processes are described and students are encouraged to be aware of, initiate, engage in and practice peer learning during the course of the module. The students are introduced to the available support structures; while the need to actively reflect on their progress towards achieving the agreed objectives at the end of each laboratory session is highlighted.

To encourage the use of logical discussion and reasoning, the set assignments are outlined concisely, using a minimal parameter set. Initially, groups are encouraged to work independently, seeking clarification from the instructor or the teaching assistant. This process of critical enquiry, and reflection on the tasks to be completed, should help students draw on their collective group experience when considering novel suggestions from their peers.

Once all the groups have developed a strategy and schedule for completing the assigned tasks, the groups are encouraged to interact with each other. The assessment methods used mean that any competition between groups is friendly in nature and inter-group collaboration is actively fostered as part of the peer learning process.

Using the nomenclature of Biggs [6], the course may be considered as comprising of both teacher-directed and peer-directed activities.

EQUIPMENT

Lego Mindstorms products comprise a programmable microcontroller (the RCX), an infrared transmitter used to program the RCX, a collection of Lego elements and software that can be used to program the RCX. Current products form part of the Lego Robotics Invention System (RIS) family [7].

The RCX is a programmable microcontroller housed in a yellow brick-like structure. It provides three inputs, three outputs and an infra-red serial communications interface. All can operate simultaneously. Additional input/output components combine with the RCX to allow the creation of autonomous, responsive robotic devices. These components include motors, a variety of sensors and more traditional Lego elements.

Why Lego is Used

Lego Mindstorms is a practical, fun approach for introducing students to the rudiments of microcontrollers, motors and sensors. It does so in an environment that encourages and promotes problem solving and small group learning and interaction. The assigned tasks are structured to be fun, to encourage an effective group dynamic and to promote healthy intergroup rivalry. Students are encouraged and assisted in dividing the principal task into modular subtasks - establishing clear requirements and targets for each subtask.

Microcontrollers

Lego Mindstorms introduce the students to the limitations of a simple microcontroller. The key limitations may be characterised as: a restricted number of inputs and outputs, limited computational power and limited memory requiring efficient programming and use of resources. In addition to the limitations imposed by the RCX, the students must also consider the limitations of individual components. The motors, sensors and other active elements exhibit slight variations from component to component. Students must factor these considerations into their design and programming process.

Development Environment

brickOs is used as the development environment for this module. It is an open source operating system for the Lego Mindstorms RCX controller. brickOs is programmed in C or C++. brickOs provides more complete thread functionality, more comprehensive data structure support and greater memory and process control than other Lego development environments.

Its key benefit for the Lego module is that it removes many of the constraints and limitations imposed by the `official' Lego firmware and software. In doing so it provides freedom for the students to implement reasonably complex solutions to the assigned tasks.

TECHNICAL TASKS

Students taking the Lego module are set a number of technical tasks to complete within a given time window. The class was divided into self-selecting groups of three with each group assembling and programming at least one robot.

The module assignment in the 2002/2003 academic year required the students to create a single robot capable of completing two prescribed tasks. Firstly, the robot should be capable of taking part in a sumo wrestling like event. As the main objective is to compete effectively, this activity gives rise to a spirit of friendly competition amongst the students. Secondly, the same robot must travel in a perfectly straight line for an arbitrary distance. A robot that is specifically purposed for one task will be non optimal for the other. These conflicting objectives provide students with the stimulus to engage in reflective discussion on robot design and implementation.

Task 1

The first, and more significant challenge, was to create a robot capable of taking part in a Lego Sumo basho. Competition occurs on a black elevated platform. A 2.5cm thick white circular band is painted at a 36cm radius to form the dohyo (ring). The competing robots must satisfy certain basic dimensional criteria e.g. fit in a 10cm square. Each group was provided with a Lego Mindstorms RoboLab kit and a collection of motors, light, touch and angle/rotation sensors. The final robots were divided into three categories – heavyweight, mediumweight and lightweight. Robots must then compete in at least three bouts against robots of a similar class.

Task 2

The second task requires the robot to travel in a perfectly straight line for an arbitrary distance. A narrow white line is provided and the robot should stop at the point where this line terminates. The robot cannot use a light sensor to `track' the line and accordingly adjust its position. It can only use the light sensor to determine when to stop.

Technical Challenges

The prescribed tasks encompass a variety of challenges for the students. Firstly, the students must analyse the task in hand, decompose it into subtasks and assign responsibility for specific subtasks to specific group members. Students are encouraged to draw on their collective skills to assign tasks to specific group members while maintaining group responsibility for time schedules and milestones. Time management and critical path issues impinge on the process e.g. it's not possible to test code for the robot if the robot is not already assembled. Similarly, it's not possible to produce code for the robot if the strategy for the basho has not been agreed.

The programming team must contend with the restrictions imposed by both the Lego kits and the programming language. The RCX has only 3 inputs and 3 outputs so no more than 3 sensors can be connected as inputs and no more than 3 motors can be connected as outputs. One of these sensors must be a light sensor to detect the dohyo boundary and also the line in task 2. Thus only 2 inputs are available for other sensors. The programming and build teams must liaise closely to determine their optimum configuration. The feedback process between programmers and robot designers also mirrors the professional practice of engineers who rely on feedback to optimize concepts during production.

Use of one or more angle/rotation sensors will be advantageous for task 2 but will constrain collision detection which is essential for task 1. Thus a robot that is specifically purposed for one task will be non-optimal for the other task. In task 2, students must consider both the limitations of the Lego elements e.g. component tolerances, in addition to the physical environment in which the task is to be performed. Different motors, actuated for identical periods of time, do not, necessarily, produce the same number of revolutions. Similarly, dual `belt' drive systems may experience asymmetrical slippage. It is thus incumbent on the build and programming teams to factor these component properties into their activities.

STUDENT ASSESSMENT

The assessment methods used are carefully chosen to reflect the course objectives and to provide positive feedback and encouragement to students during the course of the module. The diverse nature of the course objectives means that students will be assessed on their technical knowledge, group collaboration and peer learning skills. To evaluate these successfully, a mixture of formal and informal assessment techniques were employed. The assessment methods and criteria used were discussed, and agreed, with the students at the start of the module. The groups were assessed in the following ways:

The Initial Intra-Group Collaboration

During the first three laboratory sessions students become familiar with the programming environment, experiment with some basic motive robots and establish an initial group strategy to ensure successful completion of the assignment. They also establish the competencies of individual group members and identify how peer learning processes will be of benefit to them in satisfying the project goals.

The Instructor and teaching assistant attend all laboratory sessions, providing immediate feedback, encouragement and guidance as required. At the end of each laboratory session, students reflect on and discuss the nature of the collaborative dynamic established. They may then make brief notes on the group learning experience, highlighting where peer learning has proved advantageous. These notes may be incorporated into the group's final report on the module.

At the end of this design phase the Instructor and TA informally assess the group dynamic, the division of responsibility and progress towards the achievement of developmental goals for each group. Groups are awarded a mark and given feedback on their performance.

Technical Work and Inter-Group Collaboration

During the middle section of the module, each group will carry out the necessary developmental work to achieve their design goals. The sub-tasks to be undertaken typically include robot design, programming and building. These sub-tasks are carried out in parallel with much redesign, experimentation and evolution taking place.

It has been established that using formal assessment in this middle period of the module is counterproductive: Student's work strategically to satisfy the interim assessment criterion thus stifling their innovative and creative tendencies. Moreover, assessment at this stage actively discourages the inter-group collaboration that the course seeks to promote. Instead, each group is encouraged to seek peer evaluation and feedback from other groups. This peer assessment is informal in nature, but is undertaken very seriously by the students.

Ongoing critical analysis, as well as quantitative and qualitative feedback, is provided to each group. Groups are again informally assessed on their technical and collaborative skills towards the end of this phase of the module.

Technical Demonstration

In the closing stages of the module a week of laboratory sessions is set aside for technical demonstration. As almost half the marks for the module are awarded at the demonstration this session can prove very frenetic.

Each robot must compete in at least three sumo bouts. In actuality, most groups enjoy the process and compete in many more bouts. Marks are awarded to each group for their robot based on its technical proficiency, style of competition, victory and destructiveness. The latter criterion was chosen by the students. These marks are weighted by category and the marks for victory are sufficiently low so as to encourage a friendly spirit of competition between groups. The robots must also complete the straight line challenge where marks are awarded for accuracy, speed, and stopping at the required location.

Final Report

Each group was required to submit a final report on their laboratory experience. This report was to include the following information:

- the group work plan and detail of individual roles within the group
- technical details on the work carried out (e.g. programs written, robot design)
- the role of peer learning throughout the project
- difficulties encountered and how they were overcome
- information on both their individual and group learning experience

This report was assessed based on its content (technical knowledge, comprehension and analysis) as well as on the evidence of reflection on the peer learning process and its outcomes.

EVALUATION

When assessing the quality of student learning it is essential to consider the information gathered from individual students as well as the global statistical data gathered from the class. As the module described above takes a student-centred approach to learning, the evaluation carried out focused on the quality of the learning environment created.

Assessing the impact of peer learning is considered particularly challenging [2]. The use of cluster analysis has recently been proposed for multidisciplinary team settings and, while results are not conclusive, they `do suggest that cluster analysis may be a valid tool for evaluating peer interaction' [e6]. The use of cluster analysis for this study is questionable as the teams cannot be classified as multidisciplinary. However, a longitudinal study is currently exploring the use of related techniques to assess the impact of peer learning for student groups drawn from within the same discipline.

The effect of peer learning in this study was evaluated through student interviews. Student feedback was also obtained through independent class surveys and questionnaires.

In Trinity College, Dublin an independent survey of each course taken by students is carried out each term. These surveys are administered centrally by the University Quality Office. Each questionnaire includes almost thirty closed response questions where students must indicate the extent of their agreement/disagreement with a number of statements regarding the course. These questions cover a wide range of topics: Some are instructor specific e.g. `The Instructor communicated clearly and effectively', while some are more general in nature: `Classroom facilities were adequate for this course'. Students are asked if there are any reasons why they are unable to attend class regularly (e.g. the lecture was scheduled too early or too late). The questionnaire also asks the student to list two positive and two negative aspects of anything to do with the course. The Quality Office then provides the instructor with a graphical summary of the statistics gathered together with any associated student comments.

The instructor also carried out a survey to assess how student's viewed the relationship between the course objectives, learning outcomes and assessment methods used. This helped to establish whether the desired alignment of the course delivery style with the course objectives had been achieved. The results of these separate evaluations are given below; these indicate clearly that the students feel that the environment created significantly enhances their learning experience.

Quantitative

The independent quantitative evaluation criterion used may be grouped into the following classes (using a similar methodology to that of Ramsden [13]): Good Teaching, Clear Goals, Appropriate Workload, Appropriate Assessment and Emphasis on Independence. The results are summarised in Figure 1. This includes a separate class for non-course specific questions that do not fit the above criteria.

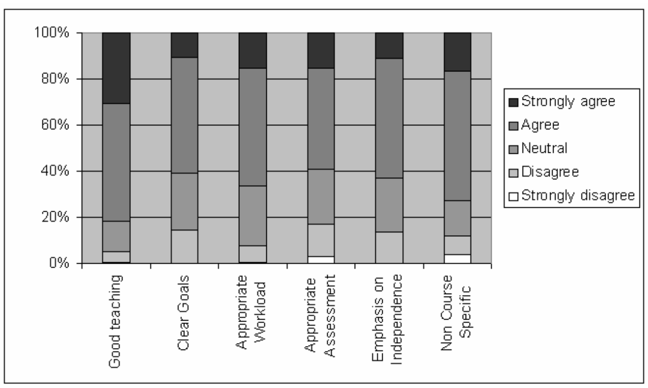

FIGURE 1
SUMMARY OF INDEPENDENT QUANTITATIVE EVALUATION

The students were also given a questionnaire relating to the course objectives. This showed that 82% of students either agreed or strongly agreed that `Building and demonstrating a robot satisfying the design criterion' and `Developing and functioning as a coherent, effective group' were features of both the course and its assessment criterion. Over 77% of the students also agreed with the statement that the course `Encouraged them to think critically' and to `Cooperate with their peers to attain mutual goals'. However only 22% of the students felt these skills were explicitly assessed as part of the course.

Qualitative

A number of interviews were held with current and former students who had taken the Lego module. These sought to assess the skills they had acquired, their impressions of the course and their experience of peer-learning. Comments made by the students include

- `It was the only course where I committed more time and effort than was required to just get through the exam'.
- `Since I was the best programmer, I seemed to only write code and didn't get to benefit much from the peer process'.
- `The freedom to design your own robots (within the specified rules) was excellent'.
- `My basic programming skills improved a lot while I was taking the course'.
- `The rules (e.g. robot dimensions) were difficult to adhere to and some of the controllers didn't work which was very frustrating'.
- `Peer learning is odd. It's really just talking to your friends about what you're doing'.
- `Our group didn't work. One of the lads just took over and we couldn't get him to tell us what he was at. We only got to do something once he'd got a robot working'.
- `It was different to other group projects. Everyone got involved and you, like, had to listen to each other and talk about where things were going and who was going to do what'.

Strategies Adopted

In evaluating the module a number of common strategies became evident. To satisfy the project goals, most groups initially focused solely on the Sumo task and largely ignored the line task. The rationale offered for this approach was that the Sumo task was more challenging, more enjoyable and had a larger competitive element. A variety of ingenious and unique robots were assembled within the constraints of the guidelines specified.

This suggests that in future presentations the tasks assigned should be more balanced, to encourage the students to divide their time more evenly between them.

CONCLUSION

A novel laboratory environment where students not only improve their problem solving skills but also practice and reflect on the nature of group collaboration and peer learning has been described. Student's work in groups to design and create Lego Mindstorms robots to carry out specified tasks. Students are encouraged to learn how to value and judge knowledge gained from their peers.

The impact of peer learning has been assessed through the use of interviews; work on using statistical methods, such as cluster analysis, to provide a quantitative evaluation of the benefits of peer learning is ongoing.

This module has been developed and refined to provide undergraduates with the necessary skills to plan a technical project logically and carry it through to completion while simultaneously developing collaboration, team-work and leadership skills.

REFERENCES

[1] Hagler, Marion O., and Marcy, William M., `Strategies for Designing Engineering Courses', *J. Engr. Educ.* Vol. 88, 1999, pp. 11-13

[2] Boud, D., Cohen, R., Samson, J., `Peer Learning in Higher Education', Kogan Page, London, 2001

[3] Haller, C. R., Gallagher, V.J., Weldon, T.L., Felder, R.M., `Dynamics of Peer Education in Cooperative Learning Workgroups', *J. Engr. Educ.*, Vol 89, No. 3, 2000, pp. 285-293.

[4] Accreditation Board for Engineering and Technology(ABET), `Criteria for Accrediting Engineering Programmes', *[Online], http://www.abet.org/criteria.html*

[5] National Science Foundation(NSF), `Division of Engineering Education and Centers', *[Online], http://www.eng.nsf.gov/eec/*

[6] Fagin, B.S., Merkle, L.D., and Eggers, T., `Teaching Computer Science With Robotics Using Ada/Mindstorms 2.0', in Proc. ACM SIGAda, 2001, pp73-78.Information and Communication Technology, *[Online] http://www.cs.tcd.ie/courses/baict/*

[7] Wolz, U, `Teaching Design and Project Management with Lego RCX Robots', in Proc. 32nd ACM SIGCSE Technical Symposium on Computer Science Education, 2001.pp95-99.

[8] Kumar, D., and Meeden, L., `A Robot Laboratory for Teaching Artificial Intelligence', in Proc. 29th SIGCSE Technical Symposium on Computer Science Education, 1998, pp341-344.

[9] Imberman, S, P, ``Teaching Neural Networks Using LEGO Handy Board Robots in an Artificial Intelligence Course' in Proc. 34th ACM SIGCSE Technical Symposium on Computer Science Education, 2003, pp312-316.

[10] Biggs, J., `Teaching for Quality Learning at University', Open University Press and Society for Research into Higher Education, Buckingham. 1999

[11] Lego.com, `Robotics Invention System', *[Online], http://mindstorms.lego.com,*

[12] Miskimins, J., `Peer Learning: Observation of the Cluster Effect in Multidisciplinary Team Settings', in Proc. ASEE Annual Conference & Exposition, 2003.

[13] Ramsden, P., `Learning to Teach in Higher Education', Routledge, London. 1992.