Educational robotics: instructional technology to unify diversity of computing topics into a single cohesive unit

Andrew Chiou, Faculty of Informatics and Communication, Central Queensland University
a.chiou@cqu.edu.au

Introduction

In computing studies at tertiary level, students are often taught different computing subjects as individual and distinct units. Often than not, these subjects are taught without any relationship to each other. This paper introduces a curriculum that shows how educational robotics, employing construction kits such as *Lego Mindstorms* (Lego, 2002) and *Fischertechnik* (2002), can be used as instructional technology to unify diverse computing and technical topics into a single comprehensive and cohesive unit. The advantages of this approach are many. In practical terms, students are directly involved in the design and development of hardware and software devices using the knowledge they have acquired as prerequisite. The very essence of computing using robotics appeals to a large student audience and teachers alike. In addition, by playing a reversed role, educational robotics can be employed as a platform for problem-based learning. Here, the flexibility of an educational robotics curriculum becomes evident – not only does it serve as a unifying factor for computing and technical topics, but the course itself serves as an instructional medium for the student's incomplete prerequisites.

Curriculum

This section briefly describes a curriculum for a 12-week term course on educational robotics. The curriculum is suitable as a special topic offered to third year or Honours level undergraduates. The prerequisites required are a working knowledge of the following areas: programming languages (i.e. C/C++, *Visual Basic*, Forth, ADA and Java), programming methodology (i.e. linear programming, concurrent programming, data structures), intelligent systems methodology, Boolean algebra, image processing, electronics and mechanical engineering. However, these prerequisites are not compulsory and serve only as a guideline. In such cases where students do not possess the proper prerequisites, this course should be undertaken with the understanding that the curriculum is to be approached as a problem-based learning project. That is, the student is required to take the initiative to obtain the necessary knowledge throughout the duration of the course. The recommended text for this course is by Ferrari, Ferrari and Hempel (2002).

Week 1 – Introduction
Introduces the student to the art and science of constructing and programming robots. This includes a description of the necessary software and hardware resources available for development of robots required for this course. Students are allowed to familiarise themselves with the construction kits.

Weeks 2 to 3 – Mechatronics Part 1: Engineering
Students are taught the fundamentals of building components required to construct robots. Specific mechanical design principles specific to robotics are taught, such as articulated joints, gear trains and pneumatics.

Weeks 4 to 5 – Mechatronics Part 2: Electronics
Electronics specific to robotics are taught to the students. Special focus is placed on different types of motors, sensors and micro controllers. However, generic electronic skills are also taught so third party components can be used to supplement parts that are unavailable from the recommended construction kits.

Weeks 6 to 7 – Software

Students are taught how to interface, control and communicate with robots via a computer. They will be introduced to software resources and tools such as, *NQC*, *LeJOS*, *legOS*, *pbForth*, *ADA for Mindstorms*, *RoboLab*, *LLWin*, *Spirit*, *Mindstorms SDK*, standard programming languages and other proprietary software tools.

Week 8 – Types of robots

Different categories of robots are introduced. This includes autonomous robotics, industrial robotics, mobile robotics, competition robotics, and entertainment robotics. Students are directed to appreciate the advantages and the functions of each type of design.

Weeks 9 to 10 – Advanced programming techniques

Advanced programming techniques are incorporated into the design of each type of robot. Intelligent systems techniques such as fuzzy logic, neural networks and genetic algorithms are embedded into the controller software to make the robots 'smarter'. Concurrent programming is also introduced to multitask the robot's functions.

Weeks 11 to 12 – Project

Towards the end of the course, three weeks (as there are no examinations for this course, the extra examination week is used for project activity) are allocated to the students to work on a project assigned to them or of their own choosing. The project work should include hardware, software and programming methodology of robotic design and construction. The project should be undertaken by groups of two to four students each.

Assessment

Assessment is based entirely on practicals and project work. The total achievable score for this course is 100. A score of up to 5 is allocated to each student per week from week 2 to week 10. This gives a possible 45 for nine weeks. Marks are given according to the student's performance and initiative. Marks should not be allocated based entirely on the level of knowledge of an individual student as some may not have the necessary prerequisites. The remaining 55 marks are given based on the final project. Of these, 20 marks are allocated for hardware construction and design. The marking criteria are based on the efficiency, elegance, ingenuity and resourcefulness of each robot design. 20 marks are allocated for the software component of the robot's function. The marking criteria are based on code optimisation and programming techniques used. Special attention is paid to the use of intelligent systems technology. 10 marks are allocated for documentation. The remaining 5 marks are for teamwork.

References

Ferrari, M., Ferrari, G. and Hempel, R. (2002) *Building Robots with Lego Mindstorms*. Rockland, MA: Syngress Publishing.

Fischertechnik (2002) [Online] Available: http://www1.tpgi.com.au/users/p8king/ [2002, April 28].

Lego (2002) [Online] Available: http://www.lego.com/mindstorms/ [2002, April 25].