Teaching by Design: An Early Introduction to Science, Technology, Engineering and Mathematics (STEM) Concepts

Authors:

Reza Nekovei, Texas A&M University-Kingsville, Electrical Engineering Dept., rnekovei@tamuk.edu
Deanna Nekovei, Texas A&M University-Kingsville, Early Childhood Education, dnekovei@tamuk.edu

Abstract *— In this paper we present the development of a six -week unit on computer programming and robotics to be used with fifth grade students. The project is twofold. The programming componen t of this project covers decision -making and robot control through a graphic user interface, the graphic user interface removes the complexity of learning programming syntax which tends to distract students' attention from learning the actual content. A s eries of game -like graphic simulations of a robot will be used to teach the students basic programming concepts such as: variables, repetition, decision -making, and concurrency. After mastering programming, a series of robotic competitions lesson plans ar e developed to teach problem solving and engineering skills. We follow a hands -on teaching method known as teaching by design. This method allows students to see the application side of learning while at the same time learning the theory behind the applica tion. Thus creating a scientific community of learners, an "Inventors' Workshop." The ultimate goal of this method is to not only teach theory through application, but also to teach critical thinking and problem solving skills that will open up new avenu es of learning for the students.*

Index Terms *— Creative learning, Critical thinking, Early childhood, Problem solving.*

INTRODUCTION

In 1960s, Seymour Papert and his colleagues at the Massachusetts Institute of Technology (MIT) designed a child friendly way of teaching computer programming concepts to young children [1], [2]. The development of Logo programming enabled young children to construct knowledge in a discovery oriented environment. Children could see their programming results downloaded into what was known as a floor turtle. The floor turtle was a simple robot attached to cord much like a computer mouse is attached, it could be programmed to obey simple logo commands such as move forward, backward, left, and right. Later, the personal computer led the floor turtle being modeled graphically on a small screen.

However, it wasn't until recently that the cost of computer technology became attainable to public schools. Additionally, computer technology has reached a level of graphic sophistication that can be understood by scientists and lay people, alike. Having these two barriers removed we are now at the stage that even children in elementary grades can benefit from interactive learning with computers. Seymour Papert [1] states "I have seen hundreds of elementary school children learn very easily to program, and evidence is accumulating to indicate that much younger could do so as well."

Throughout history educators have attempted to develop instructional material to influence student learning with minimal success [3]. Learning Science, Technology, Engineering and Mathematics (STEM) through lectures does not motivate students to learn the content and discourages students from pursuing STEM careers. These areas are best taught through hands-on interactive modules. Teaching by design enables students to work in a collaborative research oriented environment where they are provided with opportunities to think like inventors. Resnick and Ocko [4] suggest "design activities have the greatest educational value when students are given the freedom to create things that are meaningful to themselves (or others around them). In such situations, students approach their work with a sense of caring and interest that is missing in most school activities. As a result, students are more likely to explore, and to make deep 'connections' with, the mathematical and scientific concepts that underlie the activities."

Kolberg and Orlev suggest that the emphasis of creativity and problem-solving can be increased by utilizing ill-defined, open-end problem structures [5]. These types of problems do not have a single solution nor do they lead to one solution path. Further, providing hands-on learning opportunities in an environment that simulates how engineers work gets children to develop innovative critical thinking skills and it teaches them to solve complex ill-defined problems in orginal ways. When children work in teams to program robots they learn what it is like to work in an engineering environment where you may have a project manager, a programmer, and so on. From an educational point of view, children that are learning to work in an engineering environment to program robots are learning more than how to get along and get the job done. They are learning how math, science, and technology are integrated. For example when they learn measurement concepts such as distance, time, and temperature and geometry concepts such as angles, rotation, radius, circumference in mathematics they implicity understand how they can be applied to science and technology.

Thus, when children are deeply engaged as they often are when learning is made meaningful through the introduction of hands-on real world problem-solving tasks; children gain indepth knowledge of the task at hand, develop feelings ownership, and have a sense of accomplishment. For these reasons, this project had children work in pairs to program a robot in order to give them a feel for how engineers apply academic content (math and science) and technological systems such as mechanical and computations to produce something that is original.

In closing, this project seeks to apply Science, Technology, Engineering, and Mathematics (STEM) concepts by creating a hands-on real world learning environment.

INSTRUCTIONAL MATERIALS

Rapid advancement in technology has made a great contribution to hands-on learning. The low cost and availability of many user friendly robotic tools has opened the door for young children at public schools to access the technology and speed up the learning process. In the following sections, we describe some of the common hardware and software currently available and appropriate for use in the school curriculum.

Hardware Platform

Many robotic kits exist and they range widely from hobbyists robots to industrial manipulators. However, for our educational purpose, we require to satisfy the following criteria:

- Reasonable cost and assembly time
- Support for multiple sensors with concurrency
- Fully reconfigurable structure for reusing in multiple projects
- Programmable and availability of user -friendly programming tools
- Availability of related curriculum resources

Based on popularity of Lego building blocks, availability of Lego technique blocks with motors and sensors, and familiarity of children and low learning curve the Lego bricks are the obvious choice for the mechanical body of robots. This will narrow our choices of controller to one of the following two:

- **Mini board or Handy board** - The Handy board and its smaller sibline Mini board have their root in the MIT Lego Robot Design project. They were especially designed for Lego mobile robot controllers. They are available both fully assembelled or as kits. Their parts and schematic is also widely available. The handy board has four DC output drivers, sixteen input ports for both analog and digital sensors, a rechargeable battery, 32k RAM, and a LCD display. The board gives a lot of flexability in terms of sensor and interfaces in a trade-off for ease of interfacing.
- **Lego RCX brick** - The RCX is the controller in both Lego's Mindstorms and ROBO Technology Set. The RCX has three output ports (motors or light), three input ports (touch, light, and rotation), temperature sensors, 32k RAM, and a LCD display. The RCX uses an IR transceiver for communicating with a computer or another RCX controller.

We believe for younger children RCX is a better choice mainly due to its ease of setup, low learning curve, and availability of extension kits and abundant educational resources for teachers.

Software tools

To bring the mechanical robot to life with intelligence, we need simple and yet powerful tools to program its behavior. Lego provides two programming tools with RCX, a graphic programming interface which uses simple building blocks to create a program. It enables visual programming with support for basic programming constructs such as loops, conditions, subroutine, and concurrency. The Lego also provides a graphic simulator for this GUI environment with a series of exercises at Lego's SIM:BOT website. Lego also provides a library for controlling the robots in Visual Basic. Additionally, there is a fleet of other programming tools developed for the RCX. They support both graphic interfaces and commonly used languages such as Ada, C, and Java. Here we brifely describe software packages that we found useful for our project.

- **NQC (Not Quite C)** - A limited subset of C language. It provides a textual language with C syntax, which compiles to RCX bytecode. Its limitations are compatible with the RCX firmware limitations (only 16 bit integer variables, 10 concurrent tasks, 8 one-level subroutines without parameters, no return value for functions).

- **leJOS** - A Java Virtual Machine implementation for RCX. The original RIS firmware on the RCX need to be replaced by the provided firmware. It provides classes and methods for integer and floating point operations, threads, exceptions, memory allocation, and input/output interface (motors and sensors). RCX LCD can be used for debugging the program.
- **LegOS** - A multitasking operating system, which replaces the RIS firmware. It also includes a library of C API to program the RCX in standard C.
- **ROBOLAB** - A graphic programming tool similar in interface to the one provided with RCX by Lego, but more powerful. The ROBOLAB was developed based on the National Instrument's LabVIEW software and it is very popular with educators.

COURSE STRUCTURE

The project will involve 16 fifth grade public school students in South Texas. The students will be selected by the Principal and Assistant Principal to participate in this project during "Fabulous Fridays."

On Fabulous Fridays, students participate in activities designed to expose them to a wide variety of disciplines that extend the range of the academic curriculum. During these enrichment clusters, students are grouped in each grade level and/or across grade levels, according to interest. Students may choose topics in both academic and non-academic areas in the hope that they will find lifelong interests. Topic offerings correspond with Howard Gardner's Multiple Intelligences (visual, kinesthetic, oral, written, etc.) insuring a wide range of topics. Students provide input for topics for the following year through surveys. These one and half hour blocks last for six weeks.

The projects will follow a problem-based methodology approach using Lego robots [6]. Initially, students will be taught programming with graphic interface [7] in order to remove the complexity of learning programming syntax, however, once this is mastered students will be introduced into more powerful textual programming. Further, the projects will provide an emphasis on engineering and scientific problem solving. We plan to follow a five step process for solving engineering problems utilizing a computer programs and the robots:

1. Students will be asked to clearly state the problem.
2. Students will state in complete sentences the input and output information.
3. Students will outline and illustrate a solution on paper.
4. Students will convert their solution (algorithm) into a program.
5. Students will load their program into the robot and test & debug their program.

Each project will emphasize a new programming fundamental in addition to the problem solving task at hand. The fundamentals covered are the same as any introductory programming course with the difference that they will be executed on a robot, for example, the use of conditional statements on a sensor output; the use of loops for repetitive tasks; procedures to simplify the program and make it more manageable; and multitasking to simulate concurrent actions. After a series of introductory projects using the graphic mode, a more complex project will illustrate the advantage and power of textual programming. The projects titles are as follows:

1. Introduction to RIS and SIM-BOT
2. Environmental Sensors
3. Race with Gear Power
4. Measurement Bot
5. Intelligent Bot
6. C/Java Contest

At the completion of the pilot project students and teachers should be confident enough to attempt to access some online projects and curricular materials such as:

- Tufts University: http://www.ceeo.tufts.edu/
- Carnegie Mellon University: http://www.rec.ri.cmu.edu/education/index.html
- Apple computers: http://www.apple.com/education/LTReview/spring99/robolab/robo3.html
- Lego Education: http://www.lego.com/education/

PILOT PROGRAM

The pilot program was built upon constructive learning theory's premise that people learn best by actively manipulating their world, that is, by hands-on experimentation. At the outset of the study sixteen gifted & talented fifth grade students from a South Texas elementary school were selected to participate in the pilot program. Likewise, an additional sixteen gifted & talented children from the same school acted as the control group. Specifically, participation in the pilot program was based on student's rank listing of activities they would like to learn more about. Students that expressed a ranking of one in regards to the pilot program were chosen to participate in the study. The control group was composed of students that had rated the pilot program favorably with rankings of either one or two.

Research also suggests that learning is enhanced when students work together. For this reason students worked in pairs to complete all learning tasks. Furthermore, motivation to learn difficult concepts improves when learning is embedded in fun and entertaining activities.

In addition to seeking ways to promote increased understanding of computer programming, problem solving, and robotics this study also sought to measure attitudinal change toward STEM areas; and if an early introduction would encourage students to seek additional STEM courses; and to consider STEM careers. The pilot program was designed for 1.5 hours per week for a total of six weeks to be conducted during the fall of 2004.

SUMMARY

In this paper, we discussed the design and development of a six week robotics and computer programming project for fifth grade public school children. This is a gateway for the development of interest and to suggest other possible avenues of learning through playing with programmable robots. It is the hope of this project that the children and their subsequent teachers will seek out other hands-on experiences in an effort to develop a deeper understanding of STEM areas.

ACKNOWLEDGEMENT

This project was funded through Texas A&M University System Regents' Initiative Collaborative Research Grant from the Institute for School-University Partnerships Academy for Educator Development.

REFERENCES

[1] Papert S., "Mindstorms: Children, Computers, and Powerful Ideas, " *Basic Books*, New York, 1993.

[2] Pepert S., "A Critique of Technocentrism in Thinking about the School of the Future," Espitemology *and Learning Group Memo No. 2 MIT Media Lab*: Cambridge, MA.

[3] Brown M. and Edelson D., "Teaching by Design: Curriculum Design as a Lens on Instructional Practice," *American Educational Research Association Annual Meeting*, April 2001.

[4] Resnick M. and Ocko,S., " LEGO/Logo: Learning Through and About Design," *In I. Harel & S. Papert (Eds.),* Constructionism, Ablex Publishing, Norwood, NJ, 1991.

[5] Kolberg E. and Orlev N., "Robotics Learning as a Tool for Integrating Science-Technology Curriculum in K-12 Schools," *The American Society for Engineering Education (ASEE) Frontiers in Education Conference*, Oct. 2001.

[6] Martin F., "Kids Learning Engineering Science Using LEGO and the Programmable Bricks," *AERA Annual Meeting*, April 1996.

[7] Musavi M.T., and Nekovei A.R., "Robot Control in Graphic Mode," *Proceeding of Third International Robotics System Education and Training Conference*, Robotics International of SME, Dearborn, MI, August 1986.