Starting learning with computer controlled models

G. Futschek

Vienna University of Technology, Austria

ABSTRACT

Computer controlled models provide an excellent situation for creative and explorative learning. The functionality and the limits of automata and machines can be explored and understood by building their models. The paper presents a didactic concept for starting learning with computer controlled models. We discuss the aspects of interdisciplinary learning combining the topics engineering, mechanics and information technology. Special attention is given to the first steps in learning the programming of computer controlled models. The chosen model allows a stepwise introduction of concepts without using step by step instructions. It supports creative ideas, even at the very beginning of learning. This paper is based on our experience in introducing computer controlled models to students of ages 9-16.

Main conference themes:	learner centred learning
Educational areas:	primary education, secondary education
Study topics:	computer literacy, design/technology. science/engineering
Secondary keywords:	control technology, creativity, interdisciplinary, learning materials

INTRODUCTION

The first impressions about a new topic may influence the learner's attitude to this topic in future life. A good teacher or a well chosen learning environment may raise deep interest in the pupils and therefore improve ease and speed of learning.

There exist several well-known computer based learning environments with proven educational background which help the learner to develop problem solving skills. Karel the robot [1] and Logo's turtle geometry [2] are two famous examples of microworlds which should help the learner especially at the beginning.

Kits for building models of computer controlled machines have been used in some classrooms for several years [3, 4, 5]. These kits promise to encourage creative learning using many different parts which can be assembled in a nearly infinite number of ways. The written materials enclosed in the kits provide for the beginner detailed step by step instructions on how to build and program a specific machine. Using these instructions the students cannot gain experience about why the machine is built in this special way. In particular the Austrian curriculum forbids step-by-step instruction manuals. The children should be encouraged to learn in a creative and explorative way.

The author gained experience in several workshops and computer camps (see [6]). In the following we describe our didactic approach and give practical suggestions on how the students can stepwise learn the concepts of computer controlled machines without using step-by-step instructions.

COMPUTER CONTROLLED MODELS IN EDUCATION

Computer controlled machines and automata are used in every day life. Cameras, video equipment, washing machines, supermarket cash registers, airplanes and cars are some examples. We usually rely on their functionality, but we usually do not know in depth why and how these machines work. Only a deep understanding of the underlying concepts gives us the possibility to create new systems according to our own thoughts. Building the mechanics of machines, measuring physical values and controlling the machine via these measured values by a computer system, these are the three essential concepts which are necessary for deeper understanding.

The most effective learning strategy is learning by doing. The teacher's task is to support the students own experiences. The teacher has to prepare learning situations and open questions which should be investigated by the students. The teacher may give support in finding the answers, but he never gives the answers himself.

When starting to learn complex systems, the learning situations should be simplified models of real machines. Kits containing building bricks for the mechanical part including motors, sensors which measure physical values, and interfaces to a computer system which will control the machines built are needed. For several years these kits have been sold for educational use or creative play, e.g. Fischer Technic or Lego Control Lab. The motors, lamps and sensors can be connected to a computer system. The computer system usually has special software which offers the user a more or less simple programming language for controlling the motors, lamps and sensors.

The instructions included in these kits usually give a description of the functionality of the bricks, sensors and programming language commands. Furthermore some complete projects for building machines and robots are presented. The user can reproduce all steps of building a machine and can then copy the corresponding control program. As a consequence users will not be able to understand the underlying concepts in depth.

From the educational point of view the complexity of a system plays an important role. The more complex systems are closer to reality and provide more functionality, but the complexity must be managed by the children and ability in managing complexity increases with age and mental development. This aspect of complexity therefore needs special attention. In the beginning very simple models are used so that amounts of details cannot prevent the sight on the underlying principles and concepts.

Logo Turtle versus Lego-Logo for beginners use

The Logo turtle geometry provides a well-known environment for exploring geometry and for learning principles of programming [2]. The turtle is a powerful vehicle to express the children's ideas and gives them the possibility to easily create their own drawings.

Lego-Logo is an ideal combination of the educational concepts of Lego and Logo [7]. Children can use the relatively simple Logo programming system to control machines built with Lego bricks. Similar to the construction of a Lego model using bricks the programming language Logo allows the children to create their own commands based on existing commands. At the first glance it might seem that computer controlled models merely replace the Logo turtle in a way similar to the classical floor turtle, a mechanical turtle with a pencil in the middle controlled by Logo commands. But computer controlled models integrate much more educational aspects than simple variations of the classical floor turtle:

- Children can construct models of real machines or can creatively invent their own models. In this way they are encouraged to learn something about mechanical engineering. Using a kit children can build a wide variety of different models.
- The sensors form a completely new aspect. Children are able to measure the brightness of light, the temperature of air or liquid, the distance to an obstacle and other physical values of their environment (depending on the available kit). In order to use these values in a control program they have to understand the meaning of the measured values. While measuring and doing something with the measured values the children learn a lot about physics.
- The structure of a controlling program is quite different to turtle geometry programs. The programmer has to encounter the additional aspects of time and parallelism. In a real machine we can observe several independent processes which can be executed in parallel. In some cases more than one sensor is used as parallel guard waiting for special events. The movements of a machine take some time, so the sensors have to be inspected all the time. These programming aspects are closer to real systems and open to the programmer a new field for experimental and creative work.

Computer controlled models allow pupils and teachers to integrate the materials of different subjects. The complexity of the machines to be built also encourages to perform teamwork and to use principles of project management.

A newly opened kit provides no good starting point for classroom learning. We have seen many schools opening kits, playing a little bit and then putting the kits away for the next three years. A big number of different parts is not the right abstraction level for beginners and does not contain enough stimuli for effective learning. What we really need is an educational plan which describes learning situations which stimulate answering questions about this situation.

STARTING LEARNING WITH COMPUTER CONTROLLED MODELS

Reduction of complexity is necessary. Real machines are too complex to experiment with. But the provided kits are also too complex. Learning means revealing black boxes and understanding the concepts behind black boxes. A computer controlled machine is a black box. The corresponding model is a simplified black box with in principle the same hidden details.

When we open this black box a little, we see again several new black boxes:

- the mechanical engineering elements (motors, wheels, axles, joints, etc.);
- the measurement elements (sensors);
- the control program (programming language).

Each of these black boxes needs extensive attention, but in this paper we will put special emphasis on the aspect of control.

Learning the concepts of mechanical engineering

Building machines requires some basic knowledge of applied mechanics. At first glance it is not easy to see how the different technical bricks like motors, wheels, axles, joints, etc., work together. Additionally children usually want to assemble a special machine with a sometimes complicated mechanical behaviour. We highly recommend experimentation with the supplied mechanical parts in the classroom before building computer controlled machines.

Learning mechanics can be made easy. For example: in order to learn something about the transformation of a linear movement (inside a car's engine) to a circular rotating movement (wheels of a car) children can easily be motivated by the following learning situation. The teacher prepares a model which solves this task, but hides the mechanical details inside a closed box. Only the rotation axis and the linear moving axis are visible and moveable by the children. The children's task is to rebuild this model so that it has the same functionality. We prefer this creative learning aspect. It is much more effective than simple step-by-step rebuilding of models using detailed instructions.

Learning the physical aspects

Sensors can measure physical values. Physical values are often too abstract and too hard to understand. But active measuring using a sensor can be done easily. If the sensor is connected to the computer system the actual sensor values can be monitored on the screen. In some systems (e.g. Lego Control Lab) a graph of the measured values can be seen on the screen. Children can experiment with different sensors and report their experiences.

For example the light sensor which measures the brightness of light, can be used to discriminate different colours on different materials. The students may answer the question 'which colour is brighter than another?'. They will probably experience that colour detection using a light sensor is very difficult and depends highly on the underlying material and the intensity of day light.

Learning the programming aspects

For learning the basic concepts of control programming we propose the following situations and tasks. The students do not need special experience in building machines, but some experience in using sensors and measuring physical data. In order to hide the mechanical aspects we decided to expose the pupils to ready-built machines. We advocate group work and the optimal number of students working with one system lies between 2 and 3. One student is mainly concerned with programming, the second student has to handle and eventually repair the mechanical parts of the model. The third student may observe and help where needed. Pupils can learn from each other, they can discuss their problems and they can help each other.

Basic learning situation

An electrical toy train on a limited piece of track forms the carefully chosen learning situation. The track has a starting point and an end point. It is intentionally not a closed circle. So pupils have to think about the end positions of the train. The first 4 tasks do not involve sensors.

Task 1: The train should go from one end to the other and then stop.

This task encourages students to solve the following problems (open questions):

- How can I start and stop the train?
- Which of the two directions of the train's motor should be used?
- How long does it take from one end to the other?

Usual experiences gained by the students: After turning on the train goes very fast and shoots off the end of track. To avoid this accident the students need a program for starting, running for some time and stopping automatically. Exact measurement of time is necessary. Some of the students may try to use a slower speed.

Task 2: The train should go at different speeds.

This task encourages students to solve the following problems:

- How can I change the speed of the train?
- How does speed influence the time to go a given distance?

Experiences gained in this task: They start to understand that absolute time values are not sufficient to control machines.

Task 3: The train should go to the end, stop there for some time before returning to the starting point. This task encourages students to solve the following problems:

- How can the train be made to wait?
- How can the train reverse its direction of travel?

Experiences gained in this task: The built-in time is sufficiently exact for waiting a given amount of time. Going to and coming back are in principle the same programs, with different directions for the train.

Task 4: The train should repeat Task 3 several times and then stop. This task encourages to solve the following problems:

- How can a task be executed several times?
- Why does the train not stop at exactly the same point each time?

Experiences gained in this task: Students see that time is not sufficient to control a train because it does not return to exactly the same starting position. Starting, stopping, the condition of the track and other external influences change the travelled distance. Sometimes the train goes over the end of track and drops off. Nobody would go with a train controlled in this way. A feedback is needed to ensure that the train stops at the end of the track.

This need for a feedback leads directly to the use of sensors for measuring a physical value which changes when a train comes to the end of track. The following tasks therefore include touch sensors or light sensors to recognize the end positions of the track. The sensor can be placed in front of the train, at the end of the track or anywhere at the side of the track. Using a touch sensor the train will collide with the sensor before stopping. A light sensor can prevent a collision.

Task 5: The train should go to the end of the track where there is a sensor, and then stop. This task encourages students to solve the following problems:

How can a sensor recognize an obstacle? What is the sensor condition for stopping? What is the sensor condition for going? Why it is insufficient to ask the sensor status only once? How can the sensor be checked again and again during the journey?

Experiences gained in this task: Permanent checking of the sensors values is obligatory. Knowledge of the sensor values under different conditions (day light and spot light) is necessary.

Task 6: The train should go to the end with the sensor, stop there for some time (station) and then return to the starting point. This task encourages students to solve the following problem:

How can the train restart, even if the sensor indicates that it should stop?

Experiences gained in this task: At the end of track it is necessary to leave the stopping program and start the returning program.

Usually students go on to extend their models. They install sensors at both ends, so that the train goes repeatedly from one end to the other. Other students add light signals or they improve the sensors. For example they construct a lamp-and-sensor combination so that the light sensor even works at night. We observed that the students worked with this basic situation several hours without needing teacher interaction.

Some students try to use two or more sensors which should react at the same time (e.g. an automatic light in front of the train and the endpoints of the track). They learn that the program has to inspect the sensor values all the time in parallel. Principles of parallel programming have to be adopted.

The train model is very simple and easy to understand because there is no hidden behaviour. The model is easily built (low preparation time for the teacher). We observed that a lot of the students have such trains (e.g. Lego) at home and are very glad that they can control these with a computer system.

ADVANCED SITUATIONS

Other ready-built models can be used which let students build on the skills developed:

- an automatic door which should open when a person approaches it;
- a chewing gum dispenser with choice of flavours and coin insertion;
- a green house with heater, fan and door opener;
- a conveyor belt which stops when a product comes to a certain point;
- a car which can be steered as it runs on the floor.

Students have the choice which model to program. The car model allows a variety of simple artificial intelligence projects. Equipped with a light sensor the car can be programmed to circumvent any obstacles put on the floor. Another car can be programmed to run on a table without falling off the side. A third group may program the car to follow a drawn black line. It is astonishing that the programming work of these advanced projects is in fact no more difficult than the train tasks.

Inspired by these basic projects the pupils usually have a lot of own ideas about what they can do with the offered materials. If the pupils are additionally trained in some principles of mechanical engineering, they will be able to build their own machines and an interdisciplinary project can be started.

SUMMARY

The study of computer controlled systems is an interdisciplinary task with inherent complexity and strong connections to real life. We learned from our practice that children can learn a lot about real machines if they are provided with adequate learning situations and tasks. Students learn by doing, can creatively solve their own tasks and during their experiments they gain firsthand experience. Especially for beginners we presented an easy to prepare learning situation combined with a set of corresponding tasks.

REFERENCES

1. Pattis, R. E. (1981) Karel the Robot: A gentle introduction to the art of programming. John Wiley & Sons.

2. Papert, S. (1980) *Mindstorms: Children, Computers and Powerful Ideas.* Basic Books.

3. Banerji, P. (1993) *Teaching Artificial Intelligence in High Schools.* Proceedings of the IFIP Open Conference "Informatics and changes in learning", Gmunden, Austria.

4. Bariamis, G. et al. (1993) A Micro world Oriented Approach in a Multi-Functional Logo-Based Curriculum. Proceedings of the 4th European Logo Conference, Athens, pp. 163-173.

5. Suomala, J. (1993) Natural Learning in a Lego-Logo Learning Environment. Proceedings of the 4th European Logo Conference, Athens, pp. 69-74.

6. Futschek, G. and Berger, L. (1994) *Explorative and Creative Learning using Computer Controlled Models*, in Exploring a New Partnership: Children, Teachers and Technology (eds. Wright, J. and Benzie, D.). IFIP Transactions A-58, North Holland, pp. 143-151.

7. Papert, S. (1993) The Children's Machine: Rethinking School in the Age of the Computer. Basic Books.