17

Designing Logo-based microworlds for effective learning - a road to improving teacher education

Márta Turcsányi-Szabó
Loránd Eötvös University
Department of General Computer Science
1088 Budapest, Múzeum krt. 6-8.
Hungary
Tele/Fax: + 36-1 266 - 5196
E-mail: turcsanyine@ludens.elte.hu

Abstract

A discussion of Hungary's choice of Comenius Logo© also provides the framework for clarifying the role of computers in education. Methods for getting acquainted with Comenius Logo's programming environment and designing educational microworlds are examined. The benefits of Logo as a means of introducing computers to children and the use of Logo as a means of preparing teachers for their mission of achieving enhanced computer use in schools provide issues for developing countries to discuss.

Keywords

Constructionism, software, Logo, teacher education

1   INTRODUCTION

The Eötvös Loránd University, the largest university in Hungary, trains teachers in the field of informatics and prepares all teachers for the use of Information and Communication Technology (ICT) in education. Our educational policy, which has been oriented toward programming, is now shifting to use of ICT in subject

areas and moving towards the use of ICT across the curriculum - a move away from
an emphasis on the learning of the technology and computer science itself. Students
graduating from our university mainly teach at the secondary school level and, due
to the reorganisation of many schools, also at the elementary level. So our teachers
should be able to think about the overall role of education.

The university's undergraduate and in-service programs provide a strong basic
computer science curriculum in:

*   operating systems, application systems, and the design, installation and
    operation of networks;
*   computer science subjects and programming in several languages with
    PASCAL as the main principal language;
*   a strong foundation in mathematics.

Even though students take courses in pedagogy and psychology, the emphasis on
computer science and mathematics is greater than the emphasis on pedagogy, and
determines their approach to informatics for school use.

The long-term educational practice in Hungary has mainly been instructionism,
with very few exceptions over the past few years. Today's teachers have
experienced instructionist learning practices and so rely on the model for their own
teaching. This is especially true for in-service teachers, who have adopted the
method as a result of decades of educational requirements.

Considering the past experiences of teachers, it is not surprising that their vision
of computers in education has two focal points:

*   teaching the subject of computer science and all related topics;
*   using computers with computer-assisted Instruction (CAI).

In developing countries the existence of any computers is much appreciated, even if
the quality is low, and many different types of computers must be used in a single
school setting. Under such circumstances standardisation can not be achieved and
curriculum guidelines mainly start out from 'the computer' and what we can do
with it, and not 'the pedagogy' and what we need to achieve through it. The
moment the emphasis is allowed, by circumstances, to be shifted to pedagogy and
its requirements, a basic question arises: 'For what reason do we intend to
introduce computers into the educational process?' Seymour Paper gave an answer
through constructionalism: 'The kind of knowledge children most need is the
knowledge that will help them get more knowledge' (Papert, 1993).

2   FROM INSTRUCTIONISM TO CONSTRUCTIONISM

Instructionism focuses on the teacher and her/his improvement of instruction, while
constructionism focuses on the learner and on improved ways to construct
knowledge. Instructionism is a passive process in which the learner absorbs
knowledge while constructionism is an active process in which the learner
constructs knowledge. Constructionism is based in part on Jean Piaget's theories.

Teachers find it easier to just follow the guidelines of the curriculum by directly
instructing using the required materials and then investigating the amount of

learning acquired by students. They are reluctant to introduce open-ended activities. Lack of experience in dealing with open-ended learning activities and assessing possible outcomes also plays a role in teachers' hesitance to move to constructionist activities. Both tools and methods have to be shown to them and long-term engagement in the activities is needed for teachers to realise for themselves the value of learning that takes place during constructionist learning experiences.

2.1 Courses developed

In an attempt to change from an instructionist to a constructionist view of computers in education, several courses have been modified and new courses have been introduced at Eötvös Loránd University so that teachers can work directly with the concepts and actions required when teaching in a constructivist setting.

Application systems
This two semester course in the first year shifts the emphasis from the technical point of view to the user's perspectives, needs and goals; shifts the emphasis to the problem to be solved and the proper tool for solving the problem.

Informatics curriculum
This one semester course in the third year guides future teachers to develop an overall curriculum for teaching informatics; the emphasis is gradually shifting to the use of ICT in subject areas.

Computer modeling in mathematics
This block elective one semester course in third year demonstrates the use of programming languages in the acquisition of mathematical topics; the emphasis is on the need for students to construct models in order to understand the underlying theory.

Investigation of educational software
This block elective one semester course in the fourth year examines and analyses educational software for understanding its real values, asks whether it provides any benefit when used and if the method of application enhances the learning process. The approach is an expensive way to use computers in Hungary, since few schools have adequate financial resources for this type of computer use.

Computers in education
This block elective one semester course in the fourth year examines the National Curriculum and concentrates on topics other than informatics to analyse the educational benefit of the use of computers in the learning process. Research from other countries is analysed, K - 12 Web pages are investigated, and applications such as *Word* and *Excel* are studied for their potential to contribute to language and science education. Educational microworlds developed provide an economical way to develop tools and exploit software already available in schools.

Comenius Logo
This block elective one semester course in the fourth year uses the special
environment of Comenius Logo as student teachers experience a Logo course in
ways they can transfer to a school setting.
Designing educational microworlds
    This block elective one semester course in the fourth year presents the advanced
features of Comenius Logo. The design principles for educational environments are
investigated and explored in individual projects.
    These courses emphasize the changing view from 'teaching about computers' to
'teaching with computers'. On the other hand, they show examples of
environments suitable for constructive activities in subject areas. Introducing the
environments in education provides a variety of tools which enable students and
teachers to solve problems the best possible way. The young generation has to be
brought up conscious of their possibilities to produce a workforce that is open to
environments and is able to function with the best available tools in hand.

2.2 Why Logo?

Teaching future teachers the educational use of computers can sometimes be a
demanding job. After many years of studying PASCAL, programming
methodology, data structures and other topics it is difficult for them to understand
why they are not going to teach 'pure' computer science in ordinary schools. It is
even more difficult to explain why they should teach Logo. Due to the scientific
approach to computers and programming that has characterized IT's introduction to
many schools, the Logo language has not been taken seriously. It has been
considered a toy or, at most, the first programming language children should learn
in order to get a brief introduction to programming. Teachers often say, 'I would
introduce smaller children to Logo to let them believe they were programming'.
'But they are,' would be my answer, not taken too seriously by teachers.
    To complicate the situation, fancy multimedia software obscures the necessity of
teaching any kind of programming language. Creating educational software is better
accomplished by multimedia authoring tools. Custom-tailored educational software
has its own attractive appeal together with its limitations. Only an authoring tool
can provide the flexibility of environment extension and the ease with which it can
be done guarantees the possibility that the extensions can be made by the learners
themselves.
    Why Logo then? Logo is an easy-to-use, user-friendly environment developed
especially for children. Moreover, some varieties are sophisticated enough to be
used as an authoring tool. Authoring a project by students themselves permits trial
and error in construction to attain an understanding of how things work within a
system.
    The approach of teaching Logo should involve direct strategies of problem
solving in programming and the transfer of these skills to other subject-oriented
problem situations. Investigations have reported positive results both concerning
the development of thinking skills using Logo-learning environments as well as in
the transfer of these skills into other subject domains (Clements, 1990; Swan,
1991; DeCorte, 1993)

3 THE COMENIUS LOGO ENVIRONMENT

3.1 Custom-tailored software vs. the Logo environment

Many computer games intend to convey knowledge about a specific theme and let students grasp important ideas without 'studying' but at the same time learning. By using Logo to produce educational games, the extension of a preprogrammed microworld becomes easier. Comenius Logo (© A. Blaho, I. Kalas, P. Tomcsanyi, Comenius University, Bratislava, 1994) is an easy programming language as well as a sophisticated authoring tool. So it has the most important properties for a good modeling tool - from simple drawing structures to complex problems of decentralised systems. It is easy enough to master without distracting attention from the problem itself and it has the power to act as a building block for constructing additional complexity. In addition the user-friendly environment provides effective tools to attain fast results.

'Hide and seek' in Comenius Logo
Logo allows children to create associations between words in several domains— such as 'square', 'jump' or 'flower' - and their meaning. The process of creation is in itself enlightening. After successfully using, and properly testing, a user-defined word it can be hidden to behave as any other predefined primitive. However the word must be defined properly - no more or less than needed, just exactly what the word means.

Every word (procedure) must contain a concise description of its function as general as possible. Then creating new words using previous primitives produces hierarchical structures of complexity, which can still be easily surveyed. The possibility of hiding and, if needed, seeking hidden words for understanding or modification allows the creation of models, which give us insights to the 'backstage' - how the program works. Predefined modules can be developed, exchanged and built in to enhance emerging systems, thus becoming expandable. (Turcsányi-Szabó, 1996) Microworlds containing words that are meaningful in subject areas *e.g.* 'speed', 'kick', 'bounce', 'temperature' and other words can be predefined, built-in yet hidden to function as normal primitives, expanding the original set of commands.

4 NATIONALISATION OF COMENIUS LOGO

4.1 Introducing Comenius Logo to Hungary and other countries

Realising the educational potential of Comenius Logo, the Hungarian Ministry of Education has financially supported the nationalisation of Comenius Logo in Hungary and also supported teachers as they learn to exploit it in the learning process. From March 1997 all schools are able to buy Comenius Logo for a favourable price that includes teacher education. Teachers are interested in Comenius Logo since our National Curriculum, issued in Fall 1995, clearly emphasises the educational use of ICT environments for benefit of learning and

enhancing means of communication (Turcsányi-Szabó, 1997). With the emergence of compulsory informatics requirements and little time to fulfill those requirements, Logo's significant contribution to achieving curriculum goals increased. It also helped that teachers saw their students' motivation increase as they developed projects with Comenius Logo.

The Hungarian National Curriculum motivates teachers and researchers to develop environments and learning material to accomplish curriculum goals. Comenius Logo contributes because it can be used with young children while, at the same time, it enables older children, teachers and developers to produce sophisticated learning environments.

Comenius Logo has been introduced into many countries (Austria, Belgium, Brazil, Bulgaria. Czech Republic, Netherlands, Germany, Greece, Hungary, Poland, Portugal, Switzerland and the UK), producing a local version with the translation of the environment and the commands into each country's own language. Yet the environment leaves a common English command set for all versions to use as a choice. This allows a common platform for exchange of programs as a cross-cultural benefit. Comenius Logo is spreading worldwide and so has the potential of a common (yet nationalised) language for communication among children that breaks down boundaries and builds closer ties from one nation to another. The exchange of children's work can also bring teachers closer together as they compare educational strategies.

4.2 Teaching Comenius Logo to pre-service teachers

The Logo course was developed to be implemented as it would be in an elementary or secondary school classroom:

- toys and games prepared with Logo are introduced while teachers investigate their educational potential;
- Comenius Logo environment is explored as well as how commands can be introduced visually;
- bewildering animation programs are examined to show how amazingly few and easily understood commands make them work;
- little games are modified so users can realise the potential for control over the programs;
- new procedures - adding a new word to the vocabulary - deepen involvement with the programs and increase understanding of the functions of routines;
- variables are introduced to develop frequently used words;
- visual problem solving is encouraged, as discussed by Turcsányi-Szabó, and Senftleben, 1986;
- creating interactive animation occurs;
- conditional functions are introduced;
- music is created through a music add-on and recorded wave files;
- animated variables are introduced to explore the effect of change in inputs;
- functions are investigated;

- Cartesian and polar coordinates are explored in addition to Turtle Geometry and vector geometry;
- word games and text modeling are created, as discussed byTurcsányi-Szabó, and Senftleben, 1986;

Topics are investigated one at a time so teachers can create motivating problems for children that can be solved using few easy commands. Possible extensions are investigated to challenge fast-paced students or those in higher-level classes. The main motto of the course is 'Let's create games' while all the programming topics mentioned in the National Curriculum are covered at the same time subject areas such as music, visual arts, language, and mathematics are integrated into computer use. Computer-aided modeling has to be preceded by real-life experimentation and observation in order to understand behaviour and record properties. Modeling builds on the knowledge acquired and can be used as building blocks for more complex experiments in a safe environment provided that 'hands on' activities allow 'heads in' recasting experiments with 'play back' possibilities (Ackermann, 1993). The abstract knowledge attained by such experiments provides the basics for real- time constructions.

5 THE BENEFIT OF COURSES SHOWING HOW CHILDREN CAN USE COMPUTERS

5.1 Understanding that the point of interest is the child

The benefit of computers to children does not come by learning sophisticated topics of computer science, but by letting teachers understand the tools and how they can make use of them in their own world. My favourite appeal to student teachers is, 'Don't just notice the computers in class but the children who sit in front of them'.

5.2 Understanding the educational value of programs and necessity of computer use

The use of computers is not a must in any case. The value of a computer program should be thought of by investigating whether it contains virtues that could not be otherwise achieved. In other words apply the slogan, 'If something can be done as effectively, without the computer, then do it without a computer', which applies to a basic idea in the creation of educational software: programs should exploit the significant characteristics of computer power thus enhancing the form of educational environment created to be unique in this respect.

5. 3 The difference between using a tool and a procedural programming language

It is of vital importance to realise the difference between using a tool and a procedural programming language for modeling - one focuses on the end result

while the other emphasises the process itself. The process of solving a problem - modeling a theory - is the process of learning how to do or learn something.

5.4 The motivation of creating a game

Children like to play all sorts of computer games and, after getting bored with the games, their dream is to be able to author their own games. Comenius Logo is a good environment for that type of authoring and, at the same time, children learn the compulsory topics in informatics.

5.5 Integration with subject areas

Learning to program should not be a final aim in itself. Problem-solving skills should be identified and transferred into subject content areas. Projects to be tackled should be motivating, deal with topics close to the children's world and benefit overall education.

5.6 Experiences of students

Future informatics teachers were highly motivated when they produced microworlds relating to other subjects. Developing microworlds was easy to do and the teachers found they could easily let their own students experiment and create models. Comenius Logo proved to be a highly sophisticated authoring tool in the production of environments for modeling, while many of its features could be well used in the experimenting process itself.

5.7 Cheap educational environment

Microworlds created with such authoring tools are easily developed and do not have a high cost effect like the purchase of constantly upgradable official educational software.

6   CONCLUSION

Hungary's future teachers are offered several courses on the educational use of ICT, the role of Comenius Logo in the learning process and strategies for designing educational microworlds. The use of Comenius Logo and the constructionist approach to learning is gaining ground and, although teachers might be reluctant to use such open ended strategies in their teaching, the existence of Comenius Logo and similar tools motivates teachers by motivating the children. The easy-to-use authoring environments help children develop their own ideas, and encourage teachers to observe children and help them as they work. By better understanding the learning process through observation of children at work in Logo environments informatics teachers can judge more realistically the value of educational software and can help their schools develop computer-based activities that enhance the learning process.

7 REFERENCES

Ackermann, E. (1993) Tools for Constructive Learning: Rethinking Interactivity. MIT Media Lab, E&L Memo No. 15.

Clements, D.H. (1990) Metacomponential development in a Logo programming environment. *Journal of Educational Psychology*, **82**.

De Corte, E. (1993) Towards Embedding Enriched Logo-Based Learning Environments in the School Curriculum: Retrospect and Prospect, in (eds. P. Georgiadis, G. Gyftodimos, Y. Kotsanis, C. Kunigos), Proceedings of the Fourth European Logo Conference, Doukas School S.A., Athens, Greece, August 28-31, 1993).

Papert, S. (1993) *The Children's Machine*. Basic Books, New York.

Swan, K. (1991) 'Knowing what': Declarative knowledge and the cross-contextual transfer of problem-solving skills, in (ed. E. Calabrese), Proceedings of the Third European Logo Conference, A.S.I.-Parma, Parma, Italy, August 27-30, 1991.

Turcsányi-Szabó, M. and Senftleben, D. (1986) The LOGO Programming Language. *Mûszaki Kiadó*, Budapest, Hungary.

Turcsányi-Szabó, M. (1993) Where to place LOGO in teacher training. in (ed. P. Georgiadis, G. Gyftodimos, Y. Kotsanis and C. Kunigos) Proceedings of the Fourth European Logo Conference, Doukas School S.A., Athens, Greece, August 28-31, 1993.

Turcsányi-Szabó, M. (1996) Learning through modeling with Comenius LOGO, in (eds. P. Bakonyi and M. Herdon), Proceedings of Informatics in Higher Education '96. Networkshop '96 Conference, Debreceni Universitas, Debrecen, Hungary, August 27-30, 1996.

Turcsányi-Szabó, M. (1997) Present role of informatics teachers in view of applications, in *Information Technology: Supporting Change through Teacher Education* (eds. D Passey and B. Samways), Chapman & Hall, London.

8 BIOGRAPHY

Márta Turcsányi-Szabó received B.S. and M.S. degrees in computer science from Eötvös Loránd University (ELTE) and started working on research at ELTE before graduating in 1979. Since then she has continued her work as an assistant professor, teaching teachers of informatics. She has been devoted to Logo since 1982, and has written several educational materials and programs on the subject. Her present research includes the study of application systems, the educational use of computers and the design of educational programs.