Control and Cybernetics

vol. **22** (1993) No. 1/2

An experiment in computer control of the learning process

by

Władysław Homenda

Institute of Mathematics
Warsaw University of Technology
Warsaw
Poland

The paper presents the idea of an experimental computer system for controlling of learning process. This system is developed on the bases of the following general assumptions:

1. the teaching–learning process is student–centered,
2. student's attention is focussed on the main goal of the problem and not on side effects,
3. knowledge structure is established in strictly bottom–up way,
4. knowledge is explained in terms of various contexts.

The difference between the above system and other systems is that its assumptions, especially the first one and the second one, are truly applicable. The system is applied in exemplary field of linear algebra, but methods used in this system can also be applied in other fields.

1. Introduction

Applications of computer systems in education can be divided into two kinds. The feature that makes such distinction possible is the influence of the system on the control of educational process. The first kind of systems can be of use in education in many ways, but they do not control educational process or control it on a limited scale.

Mathematical libraries used in teaching of numerical analysis, integrated environments of programming languages, i.e. BORLAND's TURBO–PASCAL

In memoriam Professor Edmund Lipiński

used in teaching programming languages are characteristic examples of this kind of systems. Of course one can give many other examples of such systems applied in educational process.

The other kind of systems do not only aid process of teaching, but also control it. They can replace a teacher in many ways, e.g. while testing knowledge of the grammar or vocabulary of a foreign language system can provide additional exercises in case there are too many wrong answers. The exercises are properly selected to help the student to correct mistakes commited. It ought to be noted that a student should be given a possibility of free choice of proper procedure leading to the same solution of a problem. Students should not be forbidden to choose some other procedure leading to this same solution only because system does not know it. But systems of this kind are not often applied in education. The reason is that development of such systems encounters many difficulties in theory and in practice both in education and in computer sciences.

I believe that methods based on the theory of machine learning and artificial intelligence make introduction of such systems easier. Especially, application of expert systems in education seems to be very effective.

2. General Assumptions

The main assumption of the EUCLIDO system is to focus student's attention on the general goal of the problem being solved. It is advisable that the student's attention be entirely concentrated on the general goal because focusing it on side effects, for example: adding and multiplying fractions of large numbers, causes that the general goal of the problem is lost.

The next important feature of the system is its ability to resolve problems in a way similar to the approach of human mind, to control student's solution in this way and to explain steps leading to a solution. The system does not disturb student if he/she does not make a mistake while resolving the problem, it simply adopts itself to the student's path of solution.

There is an approach to education which assumes that the main problem of learning is creation of knowledge in a truly incremental and structured shape. It is also believed that well acquired knowledge allows to solve practical problems, and vice versa — ability to resolve problems makes knowledge to be easily acquired. To provide evidence for these statements the EUCLIDO system was based on general assumptions about knowledge to be explored :

1. the base units of that structure are elementary instances of knowledge,
2. low level concepts are generalized into higher level ideas,
3. analogies are applied for further generalization,

4. the structure of knowledge, which is being explored, is created from point of view of a small number of elementary units of knowledge.

These assumptions simply imply a bottom–up hierarchy structure of knowledge.

Knowledge structure is formed from different points of view. These same concepts and ideas may be explored as generalizations of different elementary units of knowledge, so they can be learnt and/or explained in different contexts.

It should be stressed that the goal of the system is not only to explain and learn theory or to solve practical problems. The goal is to achieve these two subgoals together.

This approach is similar to that of Forbus and Genter (1986) in bottom–up knowledge creation, stressing the importance of analogy, applying various contexts. It differs, however, in :

- adopting to student's path of the solution of the problem,
- stressing the importance of learning in the context of problem solving,
- more general meaning of analogy,
- application domain (linear algebra instead of physics).

The EUCLIDO system may operate in two modes:

- checking mode i.e. checking student's path of problem solution,
- exploring mode i.e. resolving problem and explaining steps on the path of solution.

At the current state of system design and development a student can choose and apply rules from given list.

While in checking mode, the system may decide if there is necessity to switch to the exploring mode. The decision of switching is taken accordingly to its estimation of correctness and quality of student's path of solution. Every step in solving the problem is compared with the optimal one (i.e. the step which would be made if decision was taken by the system). Quality is measured by the value of difference between the certainty factors of rules applied by the student and by the system. Measures of quality of all steps are combined into global coefficient of quality using methods of linear space of fuzzy sets (see Homenda (1991) and Homenda & Pedrycz (1991)). If this coefficient exceeds a predefined bound or if a step is incorrect, the decision of switching is taken.

3. Problem knowledge base

Knowledge creates separate areas called nests. A nest is a hierarchical structure of elementary units of knowledge, of abstract concepts and of ideas. Elementary

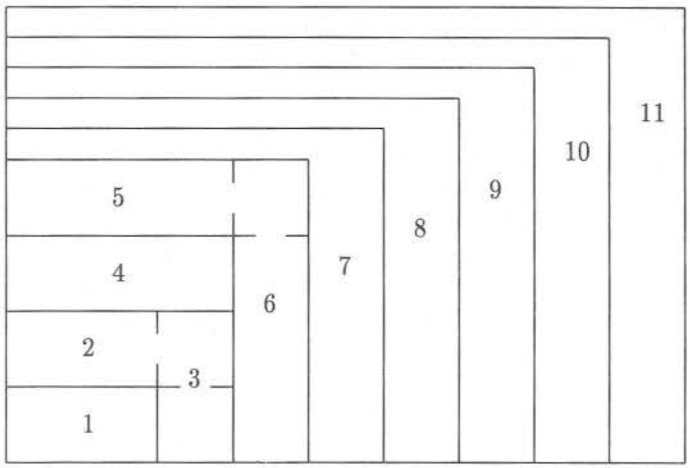

Figure 1.

units are objects, methods, algorithms which are not split into smaller units in the nest. Examples of elementary units are: matrices, vectors, rows of matrices, algebraic operations on numbers, on vectors and on matrices (addition, multiplication and so on), Gaussian elimination algorithm, etc. Collection of elementary units of a nest is called a kernel of the nest. More complex ideas are: rank of a matrix, system of linear equations, coordinates of a vector in a basis of linear space. The example of a nest is given in the Fig. 1.

In Figure 1 we have :
1 — kernel, 2–9 — more complex units of knowledge. I.e. unit 7 is a generalization of units 5 and 6 (and also units 1–4), units 5 and 6 are generalization of units 1–4 and have common area of ideas. Concepts of those areas are:

1. matrix, multiplying row of a matrix by a number, adding rows of a matrix, Gaussian elimination algorithm,
2. triangular matrix,
3. diagonal matrix,
4. determinant of a matrix,
5. rank of a matrix,
6. inverse matrix,
7. system of linear equations,
8. linearly independence of vectors,
9. basis of a linear space,
10. coordinates of a vector in basis,
11. linear mappings and matrix of linear mapping.

Another example of the nest includes the kernel consisting of elementary units: matrix and determinant of a matrix.
Generalization of those units are: submatrix, inverse matrix, rank of matrix, system of linear equations, Cramer's rule and so on. It is worth noticing that many concepts and ideas are the same as in the previous example of the nest. Moreover, the concept of matrix rank is now an elementary unit of knowledge while in the previous example it was a complex concept. It is because similar and overlapping fields of knowledge are explained from different points of view. In the first nest, determinant of matrix and rank of matrix, for example, are explored by application of the elimination algorithm. In the second nest, rank of matrix is explored by application of determinant values of submatrices.

Concepts of triangular and diagonal matrix are excluded from the kernel of the nest. The reason is that they are explored by applying elimination algorithm to any matrix. From this point of view, they are generalizations of elementary units of knowledge.

4. Commonsense knowledge base

Many difficulties are connected with assumption of flexibility and friendliness of the system. A student should be left as much initiative as possible provided steps on his path to the solution of a problem are correct. This assumption causes many problems concerning student's path of solving the problem, controlling steps on the path and eventually taking initiative over from the student.

To explain this, the Gaussian elimination algorithm should be considered in any form taken from any handbook, for example from Stewart (1973). Description of the algorithm is simple, controlling of its application to any problem does not cause much trouble for the computer system, but the algorithm in this form is not comprehensible for the beginners. The description of the algorithm is as follows :

if A is a matrix with m rows and n columns then,

for $k \leftarrow 1, 2, \ldots, min(m, n) - 1$ do

$$A_{ik}^{k} \leftarrow U_{ik}^{k} = \frac{A_{ik}^{k-1}}{A_{kk}^{k}},$$

$$i = k + 1, \ldots, m$$

$$A_{ij}^{k} \leftarrow A_{ij}^{k-1} - U_{ij}^{k} * A_{kj}^{k-1}$$

$$i = k + 1, \ldots, m$$
$$j = k + 1, \ldots, n$$

Attention of a beginner is entirely concentrated on the superscripts and subscripts. The goal of elimination is lost. Thus the sense and simplicity of idea of that method is lost.

It is better to explain the idea of elimination applying other form of the algorithm, which defines possible operations on a matrix and the goal to be achieved.

1. The following rules are permitted to be applied to the matrix:
   - any row can be multiplied (divided) by any number other than 0,
   - any two rows can replace each other,
   - any row can be added another one multiplied by any number,
2. The goal of the algorithm is transformation of a matrix to a triangular matrix (it should be noted that the goal of elimination depends on the problem and may be different, for example: to transform to diagonal matrix if inverse matrix would be obtained).

This form of the elimination algorithm gives the following application:

$$\begin{bmatrix} 3 & 2 & 1 & 2 \\ 2 & 1 & 1 & 3 \\ 5 & 2 & 4 & 1 \end{bmatrix} \quad \begin{array}{lcl} R3 & \leftarrow & R3-(R1+R2) \\ R1 & \leftarrow & R1-R2 \\ R2 & \leftarrow & R2-2*R1 \end{array}$$

$$\begin{bmatrix} 1 & 1 & 0 & -1 \\ 0 & -1 & 1 & 5 \\ 0 & -1 & 2 & -4 \end{bmatrix} \quad \begin{array}{lcl} R2 & \leftarrow & -R2 \\ R3 & \leftarrow & R3-R1 \end{array}$$

$$\begin{bmatrix} 1 & 1 & 0 & -1 \\ 0 & 1 & -1 & -5 \\ 0 & 0 & 1 & -9 \end{bmatrix}$$

rather than :

$$\begin{bmatrix} 3 & 2 & 1 & 2 \\ 2 & 1 & 1 & 3 \\ 5 & 2 & 4 & 1 \end{bmatrix} \quad \begin{array}{lcl} R1 & \leftarrow & R1/3 \\ R2 & \leftarrow & R2-2*R1 \\ R3 & \leftarrow & R3-5*R1 \end{array}$$

$$\begin{bmatrix} 1 & 2/3 & 1/3 & 2/3 \\ 0 & -1/3 & 1/3 & 5/3 \\ 0 & -4/3 & 7/3 & -7/3 \end{bmatrix} \quad \begin{array}{lcl} R2 & \leftarrow & -3*R2 \\ R3 & \leftarrow & R3+4/3*R2 \end{array}$$

$$\begin{bmatrix} 1 & 2/3 & 1/3 & 2/3 \\ 0 & 1 & -1 & -5 \\ 0 & 0 & 1 & -9 \end{bmatrix}$$

The above way of algorithm application is the result of our tendency to avoid fractions in arithmetical calculations.

The advantage of this form of algorithm is that attention is focussed on comparing the current form of the matrix with the goal of algorithm i.e. the triangular matrix.

Changes in the form of algorithm do not matter from theoretical point of view, but they make system more and more complicated. For every nest there is necessity to consider many rules.
Examples of such rules are:

1. if
   row R_k is divisible by the whole number (that means: every element of that row is divisible by that number)
   then
   divide it,
   $cf = 1$,

2. if
   there is an element $A_{ik} = 1, \quad i > k$
   then
   replace rows i–th and k–th by each other,
   $cf = 1$,

3. if
   $A_{ik} = A_{ik} + A_{lk}, \quad i, j, l \geq k,$
   then
   replace row R_k by sum of rows $R_k + R_l$,
   $cf = 0.7$,

where cf stands for degree of strength of the rule.

Many paths may be applied in solving a problem and in every path many rules may be used. Paths are equivalent from theoretical point of view. They lead to this same goal i.e. triangular or diagonal matrix. Unfortunately, every path is different from another one from the point of view of the system and may be considered by the system while controlling path of solving.

All this constitutes an embryo of an expert system (see in Homenda (1988)), and for every nest an expert system is applied. The number of rules varies between different embryos as well as between different variants of a given embryo.

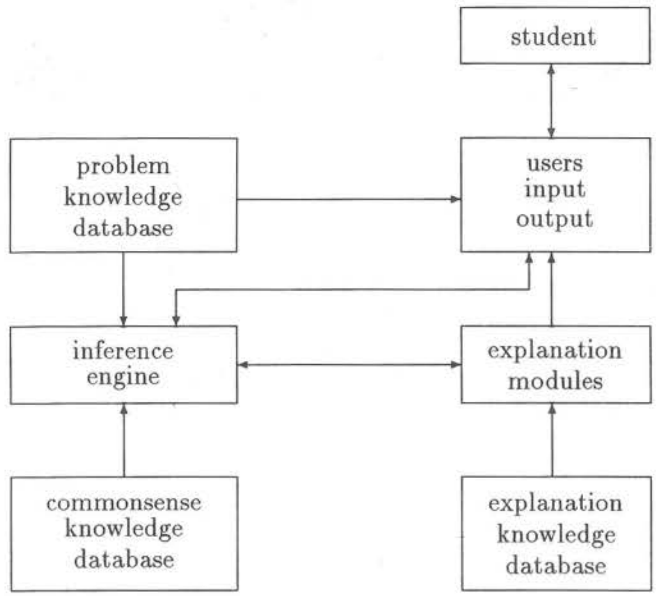

Figure 2.

5. Explanatory knowledge database

The EUCLIDO system is equipped with an explanatory knowledge database. This database consists of knowledge meant to provide the student with good explanations. It does not affect the basic inference structure and is accessed only on request by the student. Its task is to explain the student HOW the subproblem can be solved and WHY the given step on the solution path was done.

6. Structure of the system

As it was mentioned, expert systems are applied in the EUCLIDO system — a unique expert system for every nest. The general structure of the system is shown in the Figure 2.

Expert systems are created by applying expert system shell, i.e. system of the structure shown in Figure 2, but with empty databases of problem knowledge and of commonsense knowledge database. Through filling of these databases with knowledge specific for definite nests, the expert systems are created.

EUCLIDO system is being created in SMALLTALK/V in configuration given in Figure 2. The user input–output modules are still undergoing development and are not yet integrated into the system.

Next step of developing the system is to provide it with ability to acquire knowledge automatically and to improve knowledge.

Reference

FORBUS K. B. AND GENTNER D. (1986) Learning Physical Domains, Towards a Theoretical Framework, in : Michalski R.S., Carbonell J.G. and Mitchell T.M. (eds), *Machine Learning, An Artificial Intelligence Approach*, Vol. II, (M. Kaufman, inc., Los Altos), pp. 311–348.

HOMENDA, W., (1988) Methods and Tools for Expert Systems Creation (in Polish), Technical Report, Institute for Industry Organization, Warsaw.

HOMENDA, W., (1991) Fuzzy Relational Equations with Cumulative Composition Operator as a Form of Fuzzy Reasoning, Proc. of the Intern. Fuzzy Engineering Symposium '91, November 13th–15th, Yokohama, Japan.

HOMENDA W. AND PEDRYCZ W., (1991) Processing of Uncertain Information in a Linear Space of Fuzzy Sets, *Fuzzy Sets & Systems* 44, 187–198.

STEWART G.W., (1973) Introduction to Matrix Computation, Academic Press, New York.