

Network based math teaching using CAS¹

Csaba Sárvári, Pécs (Hungary)

Abstract: In the teaching of mathematics the CAS based knowledge-bank, placed on local network could be the link between the personal cognitive microcosmoses and the world of hypermedia. Modularization is one of the biggest didactical problems in connection with using of CAS in math teaching. Mainly the curricula and the role in enlarging of knowing representation net motivate working with module-types. There are some characteristic features of the CAS supported algorithm building we would like to discuss. The model-building has new and greater chances: decreasing the demand of the calculations and using compressed cognitive units, the CAS modules.

Zusammenfassung: Im Mathematikunterricht kann der netzbasierte Gebrauch von CAS eine verbindende Kette zwischen dem persönlichen Mikrokosmos und der Medienwelt darstellen. Modularisierung ist ein wichtiges und effizientes Strukturierungs- und Organisationsprinzip des - mit Einsatz von CAS verwirklichten - Mathematikunterrichts. Der Prozess der Modularisierung mit CAS ist ein dynamisches Verfahren, dessen Ablauf grundlegend von Zielen des Curriculums bestimmt werden soll.

ZDM classification: C30, D30, I10, M10, N80

1. Introduction

The beginning of the twenty first century is basically different from the previous centuries on domain of the education as well. The formerly never experienced rapid development of all life dimensions has changed and enriched over contents about knowledge and the methods of its acquisition.

Through regeneration of information-communication technology the learning environment has become wider with means of hypermedia.

The hypermedia is a uniform system of multimedia presentation, hypertextual acquisition of data and computer networks: The technical background of hyperlearning is the hypermedia environment and the didactical background is the new researches and results of the cognitive sciences and of the cybernetics.

The Internet gives a great deal of unselected information. The biggest challenge of the future's didactics: How can the internal knowledge interpreter system be developed, in interest to become able to upload, select and elaborate this enormous and continuously enlarging information-mass?

On the one side, there are the different microworlds, the personal microcosmoses with immediate mode of knowledge-acquisition. On the other hand, there is the hyperworld consist of the mediasphere and of Gutenberg-galaxy.

What could be the connecting, missing link between the two sides be? In the math teaching this mezzoworld can be – in our opinion –the CAS (and DGS) based, on the local network placed knowledge-center. (Fig.1.)

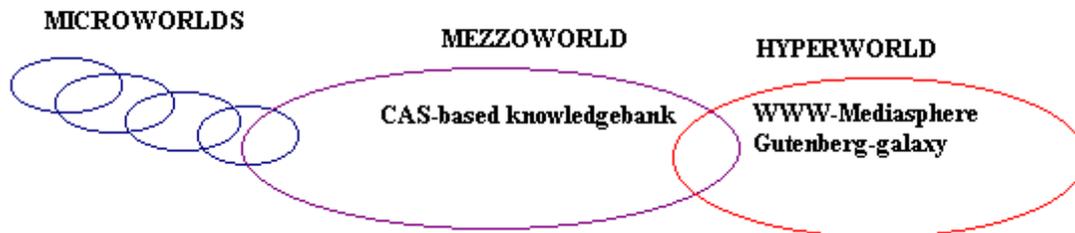


Fig. 1: The mezzoworld can be the CAS-based knowledgebank

¹ Modified version of a paper presented at the International Symposium, Anniversary of Pollack Mihály Engineering Faculty, May 31-June 1, 2002, Pécs, Hungary

2. CAS based knowledge system on the local network

In our institution – University of Pécs, Faculty of Engineering we introduced the Maple CAS in 1998 in teaching mathematics for students of technical informatics.

The complete material of the lectures, practice sessions, tests written under supervision, final tests and the set of questions have been made available on the local-network. (Fig. 2.)

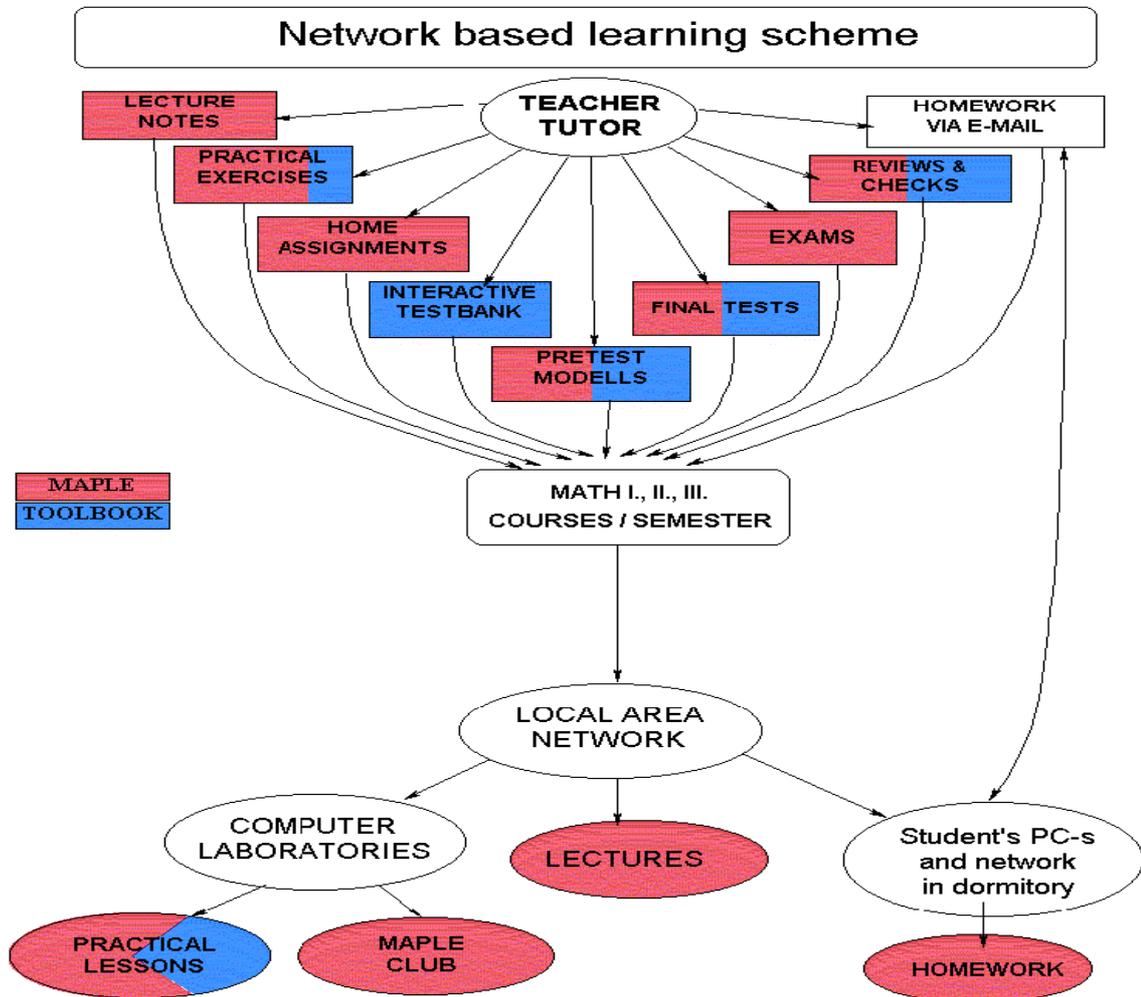


Fig. 2: Network based learning system

The students using the Maple-system are now capable of acquiring learning materials facilitating flexible, experimenting and exploratory learning. During practice sessions and home preparation students can use the complete mathematics material in a ready-to- edit format mode, they have effective means of creative material-acquisition at their disposal by modifying the set tasks.

The teaching material on the network makes possible an off-line knowledge-building. It means that the learning material has a modular-system. The different units can be used combined, and easily reorganized. (Klincsik, Sárvári 1998)

In our system we have different and partly new opportunities of feedback. Using Maple CAS worksheets and books of ToolBook authoring system we continuously test the practical skills of the students. With

help of regular, immediate confirmation and feedback the latter system is mainly used for the acquisition of basic knowledge, while the CAS is the first of all suitable systems to develop the student’s creativity. Next we would like to deal with one of the most important didactic question, with modularization, in detail.

3. Curricula based dynamic modularization

The module is more or less complex and connected system of generalized knowledge, which can be called as unified whole procedure, without to be executed. The CAS module is an immediately callable cognitive unit. The functions of the CAS module are

- To disburden the thinking
- Decrease of the complexity (Dörfler 1991)

With other words we can say, that the knowledge is deposited and compressed in the module. Application of the modules is one of the most important didactic questions in using of the CAS.

There are two fundamental idea connected with application of modules:

- In opinion of followers of White-Box/Black-Box principle it is not allowed to apply a module without knowing of the internal structure of the module
- In opinion of followers of the deposit principle there are cases, when it is useful to apply modules, without knowing internal structure at all. (Schneider 2000)

In our opinion, it is appropriate to apply both types of modules. The manner of making and using modules depends on the role of regarded module in the knowing representation net. (Fig. 3.)

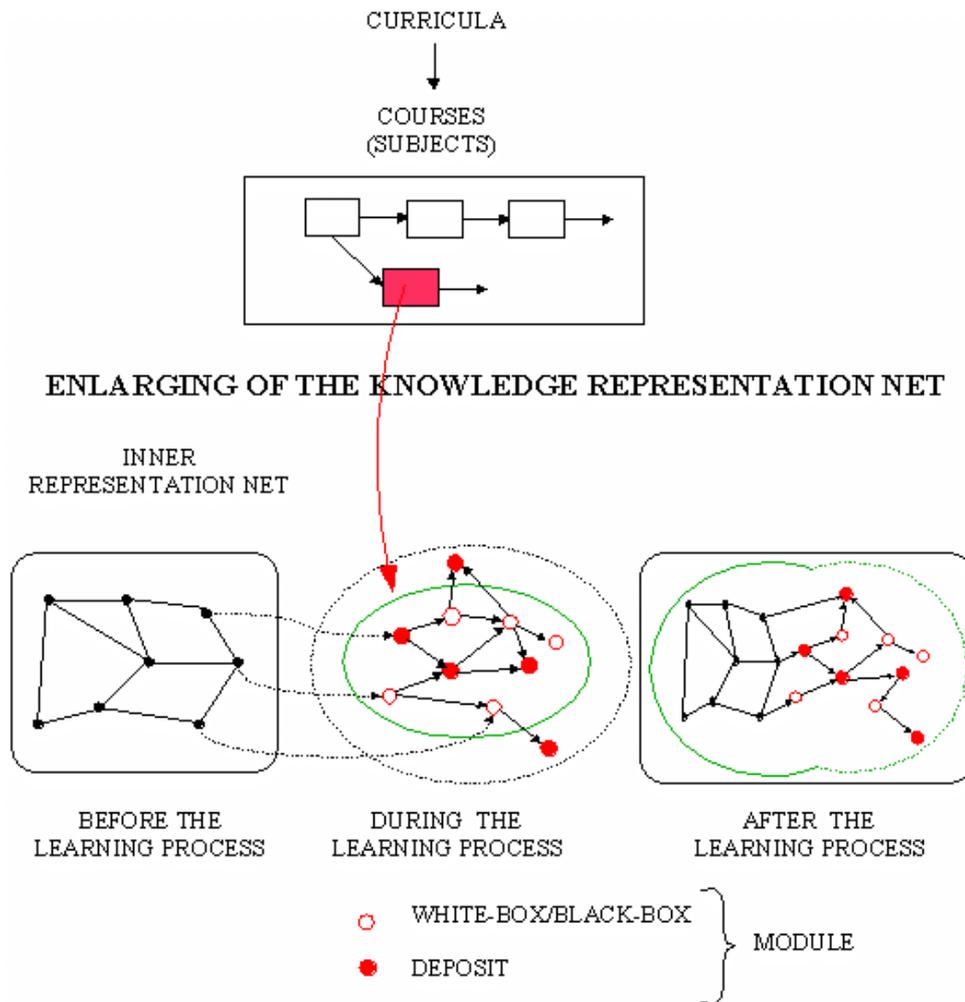


Fig. 3: Enlarging of the knowledge presentation net

Certain modules contain elements, which are required to know to be able to understand and build other modules of curricula. These modules must be built as White-Box/Black-Box modules.

The inner structure of other modules does not play a part in to understand of other modules, of other cognitive units. These modules can be discussed without discovering of their inner structure. These are the so-called outsourcing-modules.

4. The CAS supported algorithm building

In this chapter we would like to discuss the two ways of the algorithm building, which we generally use in our courses CAS supported.

The mode of building of algorithm depends on the type of problem

First let us examine the case, when we build the CAS algorithm directly to solve the problem.

The phases are:

- 1) to pose and analyse the problem
- 2) to formulate or select the modules of the solution
- 3) to prepare the solution plan which contains the main steps of the algorithm
- 4) to write the pseudocode of all procedures, containing the algorithm.

Pseudocode provides an intermediate step between an English language description of an algorithm and an implementation of this algorithm in a programming language. The steps of the algorithm are specified using instructions resembling those used in programming languages. However, in pseudocode, the used instructions can include any well-defined operations or statements. A computer program can be produced in any computer language using the pseudocode description as a starting point.

- 5) to write the CAS procedures from the pseudo code.

In the steps 1-4 we work usually with method which is called “top down”. But in the Step 5. we work often with method, which is called “bottom up” as well. We

build details of procedure from algorithm primitives and test the components, the part-procedures.

- 6) to organise the procedures to a complete run able CAS program
- 7) to test, analyse and use the complete CAS program.

There is another situation, where we would like to write a CAS program of an algorithm, which was discussed in math course to make processing easier and to reduce the complexity.

Let us see an example! Our example is the elementary basis transformation. This algorithm is to involve a new vector into the basis instead of one vector of the basis. The vectors are placed in a matrix. The procedure works with this matrix. We would like to recover the new matrix, containing the new basis.

- 1) Studying and understanding of the algorithm with the help of lectures and textbooks
- 2) As a result of understanding of the algorithm, we become able to execute the procedure as a series of algorithmic primitives.

At this step we can make execution of the disassembled primitives of the algorithm practised with the help of the ToolBook. (See Fig. 4.)

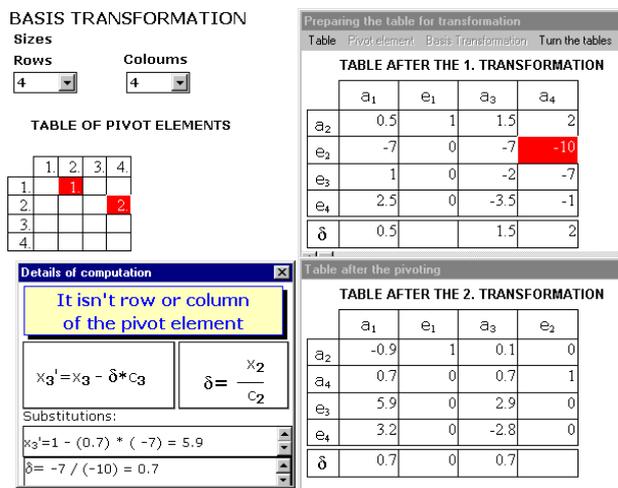


Fig. 4: Execution of basis transformation

As Figure 4. illustrates, if the procedure is half-automated, we have access each steps. It means that we can carry out of the following interactive steps:

- We can observe the details of the calculation of the new co-ordinates;
- We can follow up the partial modification and execution of several versions of the pivot procedure.

This manner of the training has got the following advantages:

- It gives the possibility of self-control, while doing the calculations without computer,
- It makes it possible to overview and grasp visually the whole algorithm (mind map)
- It provides a fine transition to the essentially automatic computer procedure

3) If the algorithm is more complicated we have to write a so-called main flowchart, which contains the main steps of algorithm.

This is a top-down plan and our aim to build a Maple procedure series to realize all these steps. (See Fig. 5.)

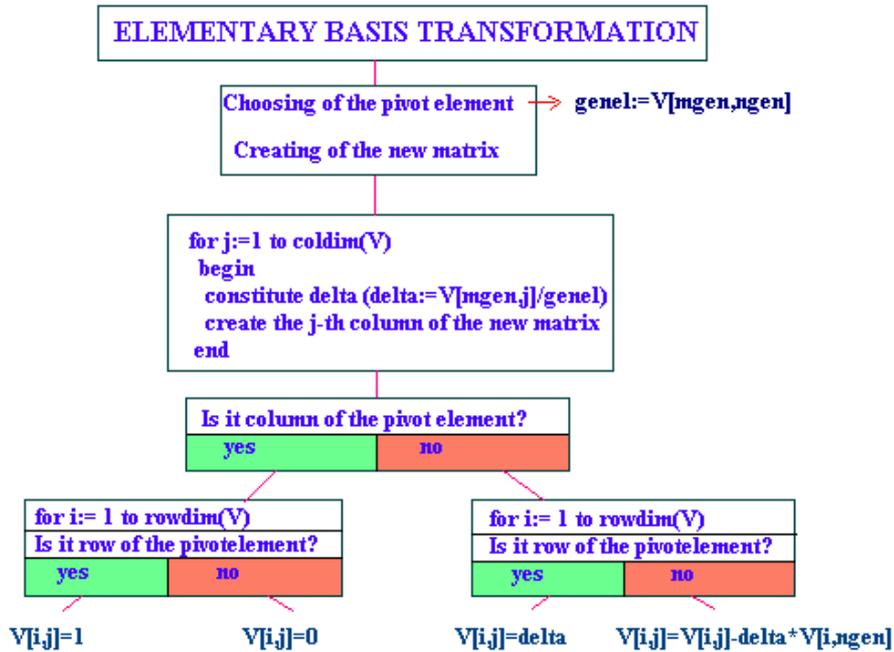


Fig. 5: Blockdiagram of the algorithm

4) We write a pseudocode of the procedure (or of all the procedures). (see Fig. 6.) Writing the pseudocode we have the following didactical goals

- to develop the skill of making an algorithm
- to fix the mental picture of the procedure(s)
- to have a language-independent form
- to prepare for writing the Maple procedure.

```

procedure elementary (mgen, ngen positive integers,
                    V: matrix)
    genel:=V[mgen,ngen] {choosing of pivot-element}
    for j:= 1 to coldim (V)
        begin
            if j≠ngen then {forbidding the column of the pivot-element}
                begin
                    delta:=V[mgen,j]/genel
                    for i:=1 to rowdim(V)
                        begin
                            if i≠mgen then
                                V[i,j]:=V[i,j]-delta*V[i,ngen]
                            if i=mgen then V[i,j]:=delta
                            end
                        end
                    end
                end
            end
        end
    V[mgen,ngen]:=1
    for i:=1 to rowdim(V)    {It is created the column of the pivot-element}
        if i≠mgen then V[i,ngen]:= 0
    V=evalm(V) {it is printed the new matrix}

```

Fig. 6: The pseudocode of the algorithm

5) We write the Maple procedures and the complete program. (see Fig. 7.)

```

elemi := proc (mgen , ngen : posint , V : matrix )
local i, j, genel , δ;
genel := V[mgen , ngen ];
for j to coldim ( V ) do
δ := V[mgen , j] / genel ;
if j ≠ ngen then for i to rowdim ( V ) do
if i ≠ mgen then V[i, j] := V[i, j] - δ * V[i, ngen ] end if ;
if i = mgen then V[i, j] := δ end if
end do
end if
end do ;
for i to rowdim ( V ) do if i ≠ mgen then V[i, ngen ] := 0 else V[i, ngen ] := 1 end if end do ;
V = evalm ( V )
end proc
    
```

Fig. 7: The Maple program of the algorithm

As we can see, the pseudocode and the Maple code have the same structure and language elements are same. It thanks that there are built from the same algorithmic primitives.

6) Testing, analyzing, debugging and using the complete CAS program.

Compatibility of vectors with vector system

> A:=matrix(4, 6, [[-1, 2, 0, 1, 1, 6], [1, 0, 2, -1, 2, -2], [0, 1, 1, 2, 3, 4], [2, 1, 5, 0, 4, 0]]);

$$A = \begin{array}{cccc|cc} a_1 & a_2 & a_3 & a_4 & v_1 & v_2 \\ \hline -1 & 2 & 0 & 1 & 1 & 6 \\ 1 & 0 & 2 & -1 & 2 & -2 \\ 0 & 1 & 1 & 2 & 3 & 4 \\ 2 & 1 & 5 & 0 & 4 & 0 \end{array}$$

> elemi(1, 1, A);	> elemi(3, 2, A);	> elemi(2, 4, A);
-------------------	-------------------	-------------------

$$A = \begin{array}{cccc|cc} a_1 & a_2 & a_3 & a_4 & v_1 & v_2 \\ \hline 1 & -2 & 0 & -1 & -1 & -6 \\ 0 & 2 & 2 & 0 & 3 & 4 \\ 0 & 1 & 1 & 2 & 3 & 4 \\ 0 & 5 & 5 & 2 & 6 & 12 \end{array}$$

$$A = \begin{array}{cccc|cc} a_1 & a_2 & a_3 & a_4 & v_1 & v_2 \\ \hline 1 & 0 & 2 & 3 & 5 & 2 \\ 0 & 0 & 0 & -4 & -3 & -4 \\ 0 & 1 & 1 & 2 & 3 & 4 \\ 0 & 0 & 0 & -8 & -9 & -8 \end{array}$$

$$A = \begin{array}{cccc|cc} a_1 & a_2 & a_3 & a_4 & v_1 & v_2 \\ \hline 1 & 0 & 2 & 0 & \frac{11}{4} & -1 \\ a_4 & 0 & 0 & 1 & \frac{3}{4} & 1 \\ a_2 & 0 & 1 & 1 & \frac{3}{2} & 2 \\ 0 & 0 & 0 & 0 & -3 & 0 \end{array}$$

This is the end of the basis transformation because of all elements are zeros in the last row.

Fig. 8: Using of the CAS program

Final remarks

In the teaching and learning of mathematics the appropriate use of CAS can greatly enhance the effectiveness of the learning process.

Using the learning materials, placed on the local network enables us to produce flexible, continuously developable learning materials.

Working with easily variable CAS worksheets helps the students to develop all levels of mathematical skills. Using the built-in CAS-modules, modules, which are created by the teacher and sometimes by the students, can lead to a adequate enlarging of the knowledge representation net.

It is recommended to develop the algorithms, the new modules in staggered way, using pseudocode.

References

- [1] Klincsik,M.; Sárvári,Cs. (1998): Modelling the Open and Distance Learning Methods in a Face-to-Face Mathematical Teaching Environments, In: Szűcs,A.; Wagner,A. (eds), Proceedings of the 1998 EDEN Conference., Volume 2, (1998) 524-527.
- [2] Dörfler,W. (1991): Der Computer als kognitives Werkzeug und kognitives Medium, In: Dörfler,W.; Peschek,W.; Schneider,E.; Wegenkittl, K. (Hrsg.) Computer – Mensch – Mathematik. Wien: Hölder-Pichler-Tempsky,(1991) 51-75.
- [3] Schneider,E. (2000): Computeralgebrasysteme in einem allgemeinbildenden Mathematikunterricht, Habilitationsschrift, Klagenfurt, (2000).
- [4] Kutzler,B. (1999): Der algebraische Taschencomputer als pädagogisches Werkzeug im Mathematikunterricht, http://www.kutzler.com/article/art_alge/ped-tool.html(1999).
- [5] Klincsik,M.; Sárvári,Cs. (): How can we combine the CAS with authoring system tools to create a learning environment containing flexible feedback opportunities, In: Böhm,J (ed), Proceedings of VISIT-ME-2002, bk teachware, Nr. SR-31, ISBN 3-901769-49-8, (2002)

Author

Csaba Sárvári Department of Mathematics
University of Pécs,
Boszorkány út 2., H-7624 Pécs Hungary
E-mail: sarvari@witch.pmmf.hu