'Intelligent' Authoring Systems: towards better Courseware.

Dick J. Bierman
University of Amsterdam[1]

ABSTRACT

After an introduction to 'Intelligent' Tutoring systems, two types of 'Intelligent' authoring environments are discussed. The first one produces 'Intelligent' Tutoring systems, that is a system with explicit representation of domain knowledge (it can solves problem in the domain), diagnostic knowledge (it maintains a cognitive student-model and knows about malrules, misconceptions and is able to reproduce incorrect solution paths of the pupil), and didactic knowledge (it knows about optimal teaching interventions given a diagnostic context). It has been claimed that if one 'Intelligent' Tutoring system exists it will itself be the Authoring environment for other domains because one has simply to replace the domain knowledge. It will be shown a.o. on the basis of the experience with the Amsterdam 'Intelligent' Thermodynamics coach that this is a false belief. The second type of 'intelligent' Authoring system produces traditional CAI-systems. The domain knowledge and didactic knowledge remains to a certain level obscured inside several presentation frames and the branching decisions. The 'Intelligence' of the authoring system would be that it knows about components of such a traditional CAI-program, about recurring structures and preferably how these structures might relate to a global teaching strategy (like mastery learning). The recently emerging generation of flow-chart based Authoring Systems are discussed within these context.

Introduction

Educational Technology has often become a goal in itself. It should be stressed that if a teacher wants to convey certain knowledge to a learner or if a teacher wants to correct false beliefs and misconceptions it is dependent on the educational context what kind of means should be choosen. Very often technological means are overkill (fig.1). This is not to say that Educational Technology is never useful but only to warn for the danger of using the technology because it appears to be fashionable.

---

[1] Dick J. Bierman, Psych. Lab., University of Amsterdam, Weesperplein 8, 1018 XA Amsterdam, The Netherlands, UUCP: .... mcvax!uvapsy!bierman

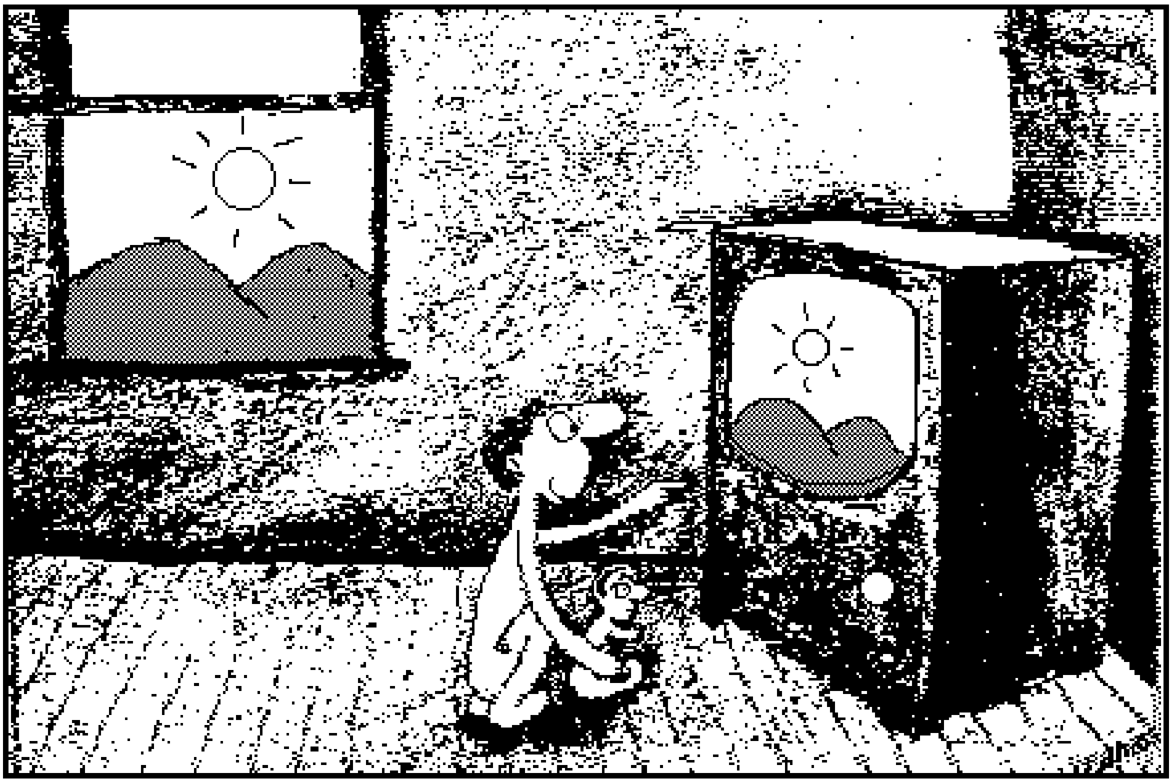

fig.1

Educational Technology ad absurdum

What is an 'Intelligent' Tutoring Systems?

The latest fashion emerging from the AI-community into the world of Computer Assisted Instruction is the appearance of so-called 'Intelligent' Coaching systems. Although there is no consensus on what actually constitutes the 'intelligence' of these systems there appear to be at least two recurrent criteria.

In the first place it is felt that the systems should be experts in the domain that they teach. So at least the system should be able to solve the problems that it requires the student to solve. For instance a system that teaches 'subtraction' should be capable of performing subtraction itself. Of course there exist no computer system that is not capable of subtraction. Therefore the requirement is not only that the system is capable of solving problems in the domain but that it does so in a 'human' way. A trace of the computer actions should resemble a trace of the cognitive actions of a human problem solver. In the case of subtraction one should observe the execution of procedures that correspond to the procedures a pupil would perform. For instance the procedure of 'borrowing'.

In the second place the systems are considered to be 'intelligent' if they can understand the pupils errors. Although 'understanding' is a rather vague concept it is very precisely defined in this context. A system is thought to understand the errors if it is capable of reproducing the error. For instance in the case of 'subtraction' the system could try to solve the problems by applying some wrong rules dealing with 'borrowing'. This could result in wrong answers that match the wrong answers of the pupils.

Due to these two aspects that give these 'Intelligent' Tutoring Systems their 'intelligence' the systems have a radically different architecture (fig.2) from the traditional CAI systems. To begin with , since they have to be experts in their domain, they incorporate a glass-box expert system. The transparency of the expert systems is reflected by the fact that the knowledge about the domain to be thaught and the didactic knowledge are in priciple explicitly represented and therefore inspectable and easy to maintain. This contrast with the traditional courseware where for instance the didactic knowledge is implicit in the branching schemes. Very often a traditional Courseware author is even not aware of the (intuitive) knowledge that is going into his or her courseware.

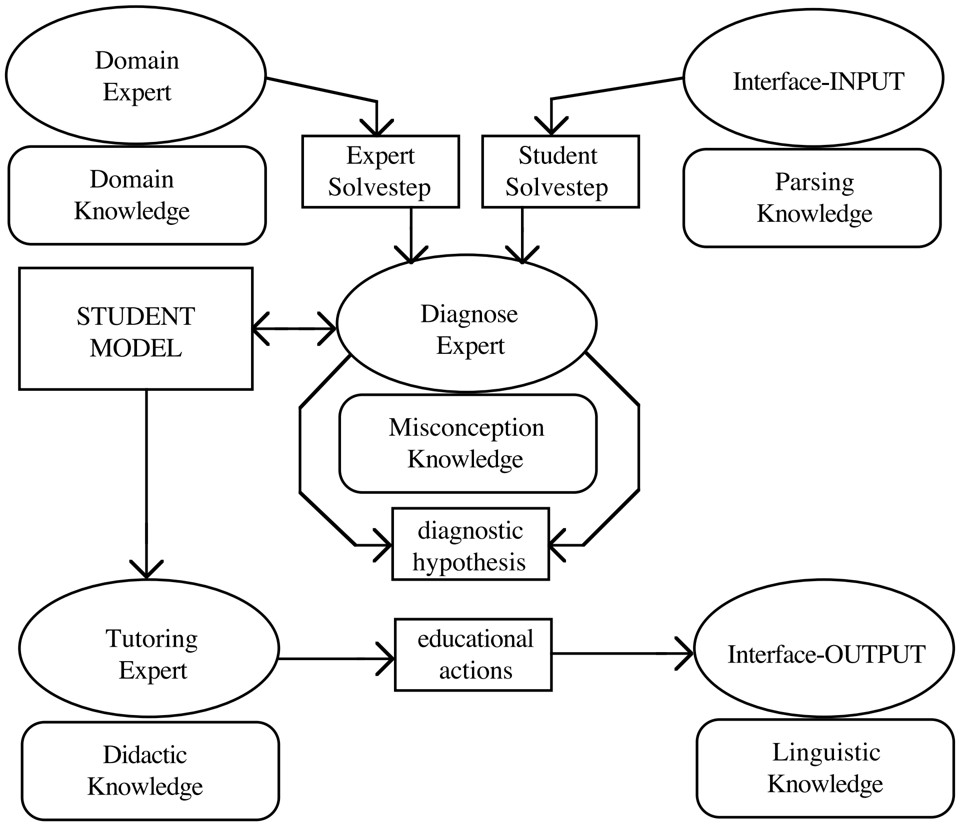

fig.2

Global architecture of the Amsterdam Thermodynamics Coach

The core of the architecture is the so called student model. In 'intelligent' systems the student-model is a dynamic cognitive model and this model should be 'runnable' if the system is required to be able to reproduce the errors of the pupil. However a part for some simple domains like subtraction no systems with runnable student-models have been implemented yet (this is due to the combinatorial explosion of malrules). So according to this criterion the number of really implemented 'intelligent' CAI systems is quit small indeed.

It should be remarked that the architecture sketched above does **not** imply that by adding some didactic component to an existing (off the shelf) expert system necessarely results into an adequate tutoring system. The knowledge represented in (off the shelf) expert systems is superficial by its nature: no deep knowledge is available that would justify the rules (Clancey,1982). Also the control structure of these expert systems does not always allow for inspection of the solution trace to the level of detail necessary for the diagnosis nor can most expert systems handle the non-availability of knowledge. The requirements that an expert system should fulfill in order to be useful in an 'Intelligent' Tutoring System are described extensively elsewhere (Kamsteeg & Bierman, 1987).

The approach taken in the Amsterdam ThermoDynamics coach project. (Bierman & Kamsteeg, 1986) was to use an existing glass box expert-system and adapt it so that at least for each correct step it was possible to trace what procedural and factual knowledge had been used. A cognitive student model. and a tutoring component were added. It was not tried to make the student model runnable Also the system did not construct the problems presented to the student by itself. Instead a problem was picked from a library of problems.

Problems with 'Intelligent' Tutoring Systems

A major problem with this approach is the problem of the diagnosis of the student. Since a cognitive model is rather detailed it seems that detailed diagnosis is necessary which might strongly interfere with the ongoing interactions. We solved this problem with a special interaction feature called the 'electronic scratchpad'. A detailed description of this scratchpad and the underlying Object-oriented extension to Prolog which enables easy parsing of the symbolic contents of the scratchpad is given elsewhere (Bierman & Anjewierden, 1986).

fig.3

'Electronic Scratchpad'

A second more fundamental problem is that the necessary didactic knowledge is not available. The current theories of learning and teaching do not have enough detail to be used as a basis for the implementation of an 'Intelligent' Tutoring Systems. In the Amsterdam Thermodynamics project this problem resulted in the development of a new methodology to elicit this knowledge from expert teachers (Kamsteeg & Bierman, 1986). However it turned out that the amount of implementable knowledge elicited from expert teachers was also rather dissapointing. Although this could be due to the techniuqe that was used used, it was felt that a more reasonable explanation was that teachers do not maintain such detailed cognitive student models. Therefore recent research focusses again on the student and his/her individual learning processes in a 'intelligent' simulation environment (Kamsteeg & Bierman, 1988). The hope is that this research will eventually result in a theory on individual learning which is specific enough to be implemented in an 'Intelligent' Tutoring System.

What is an 'Intelligent' Authoring System?

If there is already a considerable amount of confusion about what constitutes a 'Intelligent' Tutoring Systems this confusion becomes even larger if we talk about 'Intelligent' Authoring Systems ('intelligent' authoring systems).

In principle there are 2 main types to be considered.

a) The first one, let's call it 'Intelligent' Authoring Systems, type-1, helps the author to produce an 'Intelligent' Tutoring Systems.

b) The second one, type-2, helps the author to produce traditional courseware.

One can illustrate this by looking at the knowledge that will eventually be in the Courseware and where it originates.from. Traditionally we have an 'intelligent' author producing Courseware and supplying all the knowledge therein. Now consider the 'Intelligent' Authoring Systems-type1 system. This would produce 'intelligent' courseware.

'Intelligent' courseware has the following explicit knowledge components:

1. Domain Knowledge
    a. factual
    b.procedural
2. Didactic knowledge
    a. Diagnostic (Cognitive)
    b. Tutorial
3. Interaction

In a system of the first type only the Domain knowledge originates with the author. The other knowledge components are thought to be more-or-less domain independent and should be a permanent part of the 'Intelligent' Authoring Systems. Back in 1985 a prototype of such a system called 'FITS' was under construction at the University of Essex (Woodroffe, 1985). However, FITS never became quite fit. And that does'nt come as a big surprise.

In the first place for domains slightly more complex than 'subtraction' the amount of possible malrules is enormous. Many of those rules or combinations of them result in the same cognitive mis-behaviour. Therefore it is impossible to get at an adequate diagnosis based on the malrule approach; **domain-dependent** heuristics are necessary to give direction to the cognitive diagnosis. Thus the assumption that the author only has to supply domain knowledge is incorrect. Secondly, as was mentioned before, even if a detailed cognitive diagnosis of the student is available it is just unknown what constitutes an optimal tutorial intervention. Furthermore the idea that the majority of the system is domain independent requires the representation of diagnosis in an abstract formalisms like eg. '*a confusion between concept A and B, due to attribute C of A*' in which the variables A, B and C, might become instantiated. Consequently the representation of tutorial interventions should be based upon the abstract formulations like above and upon an abstract template of the student-model. It has been suggested to use for this student model a replica of the domain expert system augmented with a formalism to indicate to what extent each knowledge item in this model is known by the student ('overlay-model'). The advantage of this approach in the context of 'Intelligent' Authoring systems is clear. However this overlay-model does not allow for the representation of false knowledge. Misconception are often very domain specific and can hardly be represented across domains in an abstract formalism.

Finally although it is feasible within a certian class of domains to have the system itself construct an optimal problem for the student, the approach generally adhered to is to have a pool of problems from which the system on the basis of the difference between actual student-model and the educational goal can choose that problem from which the student might benefit most. Thus the author has to furnish a pool of problems together with a formal description of these problems.

Currently, a more modest approach for building a shell for 'intelligent' help-systems with limited coaching capabilities is sponsored by the european ESPRIT program (Breuker, Winkels & Sandberg, 1987).But even in this limited approach the requirement of a domain-independent shell creates tremendous problems.

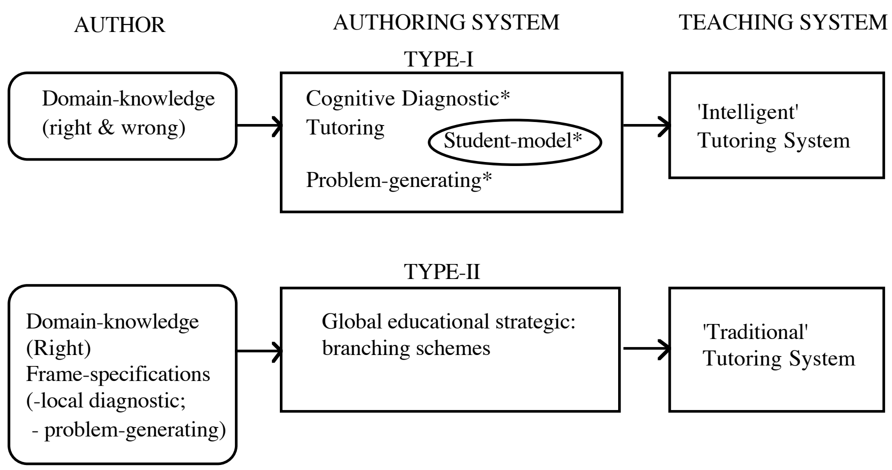

fig. 4

Knowledge sources for two types of 'Intelligent' Authoring[2*]

* The knowledge embedded in type II Authoring system corresponds to the 'ideal' situation. In practice parts of the knowledge marked with * would originate from the author (see text for details)

For the second type of Intelligent Authoring systems the situation is rather complex. The system might supply some didactic knowledge in the form of commonly used control structures (branching schemes). This would be knowledge that is generally generated by an didactic expert. For instance a representation of a mastery-learning type of educational strategy. Ideally the system should be capable to derive from domain characteristics and the educationale goals the optimal control structures. This is what contitutes the 'intelligence' of the educational expert. The basic assumption is that there are certain (combination of) strategies which are beneficial for each student, independent of the specific student characterisitics. Thus this approach might result in less diversity in teaching paths. Even less than what can be found in current Courseware which has been produced 'by hand'.
Another feature of these systems is that they might help the author in structuring the knowledge about the domain. In fact the system would act as a knowledge elicitation device, asking the user to supply relevant characteristics of the domain (concepts, attributes, relations, procedural knowledge etc.). The system could then on the basis of this domain specification and of the educational goals generate the appropriate control structures. Since the system would not know about common errors each node in the control structure would have to be specified by the author.

At the moment these systems do not exist yet. There are two major problems. In the first place the intuitive knowledge about the relation between domain characteristics and teaching strategies as reflected in global control structures is not available. In the second place the methodology to elicit knowledge from an expert is still in its childhood.

However the situation is not hopeless. At the University of Amsterdam a methodology has been developped (Breuker & Wielinga, 1987) which, in the process of building an expert system, enables knowledge engineers to get grips on a domain in a structured way. This methodology will become available in a computer assisted form. Such a computer assisted method for knowledge elicitiation could form the basis of an 'Intelligent' Authoring Systems system of type-2. The system will know a taxonomy of domains with specific characteristics. Research proposals have been submitted to apply this approach to the field of education.
Although there is hope for type-2 systems it will take at least another 4 years before they become available.

Present Situation

Currently a new generation of Authoring Systems are becoming available which might bridge the gap between traditional Authoring Systems and 'Intelligent' Authoring Systems of the second type. These systems are on itself not 'intelligent' at all. What they do however is to stimulate the intelligence of the author who is using these systems and to make the elicited knowledge very visible. The courseware is produced by directly building a flow-chart of the lesson. A simple example of the author's screen is given in figure 5.

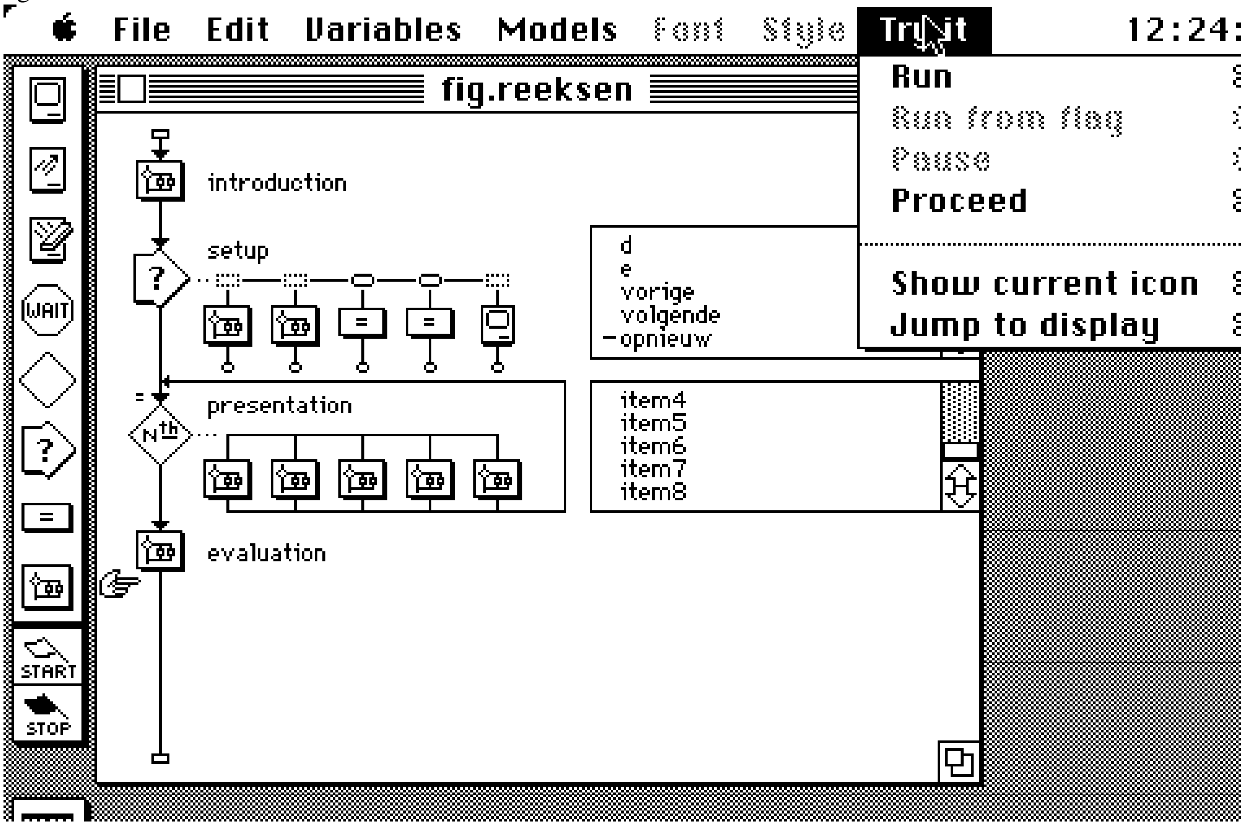

Figure 5

Screen dump of 'flow-chart' based Authoring system

On the left hand side of the screen are symbolic representations of several possible nodes in the flow-chart. Main nodes are the animation, question and decision nodes.

Thus a structured approach.is stimulated in a natural way. In at least one of these systems one can save these structures in 'models' and re-use the same global branching scheme in other lessons. In this way the author is able to make a library of models each representing some teaching strategy. (Actually inspection of these models might teach us a lot about relations between domain characterisitics and teaching approaches, thus could be helpful in the development of 'Intelligent' Authoring Systems-type2). Interestingly there is no jump node on the screen of fig. 4. This might look at first sight as a real draw-back . However it might reflect the desire of the people who designed this Authoring Systems to make the author think about what he/she is doing If one allows to jump from any node to any node the overview easily gets lost (and so does the student). After some reflection one realizes that branching back is still possible but only if the author has structured the domain in a thoughtful way (for instance as a tree of knowledge nodes). Of course the author has to do this on his own, there is no help offered by these systems yet.

Conclusion

In conclusion it appears that 'Intelligent' Authoring Systems-type2 will become reality and that the present generation of flow-chart based Authoring Systems are a natural way towards there more 'intelligent' systems. They do however require still a number of experts to be in the CAI-development team.

One could ask what about the 'Intelligent' Authoring Systems-type1. I personally doubt whether they will ever become an applied reality. Not because it is impossible but because it is very doubtful whether the increase of performance of 'Intelligent' Tutoring Systems over traditional Courseware is ever worth the extra effort that has to be spended in the development of these systems. There are strong doubts within the 'Intelligent' Tutoring Systems-community if a detailed student-model might enhance the teaching efficiency in a measurable way (Kelly et al, 1987). Of course these student models are instrumental to bring about understanding of the individual learner. Therefore research on 'Intelligent' Tutoring Systems will continue and in the end might yield spin-off for all educating systems not in the least the human teacher.

References

Bierman, D.J. & Kamsteeg, P.A (1986) *A computer Coach for thermodynamics*. Final report of SVO-project 1039, University of Amsterdam, Institute for Cognitive Studies (ICO), 1986.

Bierman, D.J. & A.A. Anjwierden, (1986), *The use of a graphic scratchpad for students in ICAI*. Proceedings of the 27th ADCIS Conference (pp.68-71), New Orleans.

Bierman, D.J. & Kamsteeg, P.A. (1987), *A methodology for the development of 'Intelligent' Educational Systems*. In (C.Schwarzer & B.Seipp, Eds.) Trends in European Educational Research. p.153-166. Braunschweiger Studieen Reihe, Dusseldorf.

Breuker, J., Winkels, R. & Sandberg, J. (1987). *Coaching Strategies and Tactics for Help systems: Empirical findings and model construction*. Proc. of the 3rd Int. Conf. on AI & Education, LRDC, University of Pittsburgh, May 8, 9, 10 1987. p. 16.

Breuker, J. & Wielinga, B.J. (1987). User models in the Interpretation of verbal data. In A.L. Gibbs (Ed.) *Knowledge Acquisition for Expert Systems, a practical handbook*. Plenum Press, London.

Clancey, W.J., (1982), Tutoring Rules for guiding a case method dialogue. In: D. Sleeman & J.S. Brown (Eds.), *Intelligent tutoring systems*. London: Academic Press.

Kamsteeg, P.A. & D.J. Bierman, (1987), *Constraints on an expert system for use in ICAI*. Proceedings of the

2nd World Basque Congress: conference on AI, San Sebastian.

Kamsteeg, P.A. & Bierman, D.J. (1988) *A Prolog-based Simulation Environment for Physics as a tool for fundamental Educational Research*. Submitted for presentation on the first European Conference on AI and Education.

Kelly, A., Martinak, R. & Sleeman, D. (1987) *How important is diagnosis for Remedition?* A concern for Intelligent Tutoring Systems. In: Heinz Mandl, Angelika Grauer & Emmanual Picard (Eds.) Abstract Volume of the second European Conference for Research on Learning and Instruction. Tübingen, FRG. Sept. 19-22,1987. p123.

Ohlsson, S., (1986), *Some principles of intelligent tutoring*. Instructional Science, **14/3-4,** 293-326.

Papert, S., (1980), *Mindstorms: Children, Computers and Powerful ideas*. New York: Basic Books.

Woodroffe, M.R. (1985). FITS, a framework for an Intelligent Tutoring Proc. of the 2nd Int. Conf. on AI & Education, University of Exeter, Sept. 2&3 1985.