Student models, scratchpads and simulation.

Dick J. Bierman, Paul A. Kamsteeg & Jacobijn A. C. Sandberg
University of Amsterdam

1. Introduction

There appears to be a general feeling of impasse in the field of ITS. Some claim that radically different architectures are needed [1], others try to reconcile the field with more open-ended learning environments [2]. Special workshops are even organized [3]. We suspect that one of the underlying reasons for this feeling is the implicit recognition that research on complete and truthful cognitive student models, one of the key-components in the traditional ITS architecture, has shown hardly any progress and that serious doubt exists whether this progress can be expected in the short term or is possible at all. One can even doubt that for practical purposes such a student model is even necessary. This paper deals a.o. explicitly with these doubts.

I. Student Models: the emperors cloths?

Computer generated student (user) models which are complete and truthful representations of the student's cognitive state are an illusion. It takes cognitive scientists weeks of work with one single subject and with tools not available to ITS, like the analysis of thinking aloud protocols, before they might be able to infer the cognitive processes underlying subject's problem solving behaviour [4]. How could we ever expect the computer to perform this task in real time, given the restricted bandwidth of the man machine communication channel [5] and without interfering with the teaching process (Eg by questioning the student about his cognitive state)?

Even if we had an ideal student model, we wouldn't know how to choose proper teaching actions based upon the model. There are simply no theories of learning which are strong enough to infer the relation between the subtle aspects of such a student model and optimal teaching action [6].

Recent surveys on student models [eg. 7] generally discriminate between overlay, differential and perturbation models. However, these surveys fail to indicate that each of these models are around since the early days of the

field. Overlay and differential models are acknowledged to be essentially incorrect because they assume the knowledge of the student to be a subset of the expert's knowledge.
Also there is serious doubt if the use of perturbation models does increase the effectiveness of teaching. For instance it is highly questionable if teaching based upon correction of diagnosed malrules is better than teaching without explicit diagnosis of these malrules. Sleeman [8] compared Malrule Based Reteaching with Remedial Reteaching. Although in his study no difference was found, Sleeman, after emphasizing the effect of direct feedback (which is a feature appearing in traditional CAI systems and ITS alike), concludes ......*the subfield of ITS should not conclude that the task of building an ITS is impossible but it should conclude that the task is harder than the field initially thought ... because ... The target knowledge structures are generally far more subtle and more complex than the simple models used in current ITSs.....*
Apparently Sleeman still believes that, given better student models and better learning theories, it should be possible to get better teaching than teaching that is not based upon a detailed student model.

This belief might be based upon the assumption that human tutors outperforming ITS's, equipped with a malrule based student model [9], is due to the superior quality of these human tutors student models. But an analysis of thinking aloud protocols of human tutors suggests that, on the contrary, they *hardly* maintain a subtle and complex student model, their representation of the student seems to consists of a global classification augmented with a number of very localized diagnoses [10]. The reason that they outperform ITS possibly has to do with the non-verbal affective components of teaching rather than with the complexity of the student model. For instance human tutors might easily observe the amount of confidence that a student displays while solving a problem. Students answers given with confidence should be treated differently than answers given hesitatingly. Similarly, students might become bored or angry without the ITS noticing it.

So should we forget about increasing the complexity and subtlety of student models when it comes to developing an ITS?

Before answering this question it should be noted that while research indicates that human tutors do not invest much in building a model of their student, the educational practice is that students with evident learning problems are singled out for further diagnosis in special institutes which spend much effort to localize the source of the problems. It might be that deep diagnosis is useful for this and maybe other special subgroups. In our view the answer to the question on the use of student models furthermore depends on the goal of the ITS.
If the goal is to produce a practical educational environments for a normal student population in a cost effective way the answer seems to be: forget about detailed cognitive student models, at least for the time being, and focus on global classification of the student in a number of subgroups. This prototype matching not only is the behaviour of real human tutors, but it also appears to be the most successful way to model users of non educational systems [11]. These latter models do not pretend cognitive validity but they appear to be useful for the task at hand. We might label them "useful models".

If the goal is to get more insight in individual learning we feel that the development of more subtle and complex cognitive student models is more or less necessary. In contrast with Sleeman we strongly doubt if computerized very intelligent educational environments with such cognitive student models will significantly increase teaching results compared to traditional (less intelligent) systems which use rough classification and proper localized diagnosis and adaptive feed-back. afzwakken

However, although in principle it is possible to think of models that might represent more subtle and complex cognitive structures, there remains the practical problem in diagnosing these subtle and complex structures. This bandwidth problem, which is most evident in the context of cognitive valid student models, is also present if one wants to have proper localized diagnosis in the context of LITS (less intelligent tutoring systems). So even outside of the research setting we have to face this problem. It is this problem which is addressed by the use of scratchpads.

II Scratchpads

2.1 Relation to educational environments

One of the ways to get around the problem of inadequate student models is to assume that students themselves know best what teaching they need, and just supply them with an environment in which they have the opportunity to explore the information that they are supposed to acquire. These discovery environments are reported to have the positive side effect that the knowledge, once acquired, would be remembered better and possibly transferred more easily to other domains. It appears, however, that these assumptions have hardly any empirical support. Although one can sympathize with the uneasy feeling which might arise from the asymmetric model of a teacher who knows and transfers knowledge to a student who does not know, it should also be realized that the relation between parent and child has similar asymmetries. Asymmetry does not necessarely result in authoritian behaviour: inequality in knowledge level can stil be resolved by negociation.

Combination of ITS's with a discovery environment should not be a strategy to hide and escape the underlying problem of inadequate student models. The environment might very well *look* like one that gives the student 'complete freedom' but, like Rousseau [as cited in 2] , we feel that this freedom should actually be completely under control of the teacher. In that case the problem of the inadequate student models just returns because it should be decided on the basis of this student model at what moment what constraint should be put upon the apparent freedom of the student.

Interestingly the combination of ITS with discovery environments results in a shift of focus on the internal representations of the ITS and the related discussion on architectures to an emphasis on the man-machine interface and the related discussion to visual representations. Elsom-Cook [2] states for instance: *...An issue which has rarely been considered in tutoring systems is that of the interface the student actually sees... ...it is no longer reasonable to build a system and 'tack an interface on' afterwards. The interface is in many ways the keystone of the design process....*

2.2 Theoretical background

The importance of the man-machine interface for ITS has been emphasized earlier [12, 13] in the context of increasing the bandwidth of

the communication channel, to enable better diagnosis. This increase of bandwidth facilitates local diagnosis and classification of the student but might even enable more subtle and complex student models.
Diagnozing students can be seen as inferring internal cognitive structures from external behaviour. More observable external behaviour potentially might result in better diagnosis. Compared to human teachers, artificial teachers are inferior in this respect, because the computer is neither able to observe non-verbal behaviour nor, at present, to interpret verbal behaviour (within reasonable processing times). The central premise for the approach of using scratchpads (and related tools) to increase the diagnostic power is, that observable acts of the student can be interpreted as externalizations of knowledge which is at that moment in 'working memory'. This is the very same premise as for the use of thinking aloud protocol analysis as a diagnostic tool [4]. In other words: the target knowledge of diagnosis, the internal cognitive structure, is only accessible via two intermediate information systems: the contents of working memory and the externalization thereof. The externalization, in the case of protocol analysis, is the verbalization of the contents of working memory.
The externalization task should be stimulated and be set-up in such a way that working memory contains the relevant information, in such a way that the externalization task does not interfere with the main task. The task can be approached in two ways: model-driven or data-driven. Or, put it simply, by asking questions or by monitoring.

2.2.1 Questioning

Asking questions is (should be) model-driven. The contents of the question should be chosen in such a way that exactly those parts of the cognitive structure in which one is interested, are brought into working memory. This approach is most suitable in cases where the student is not involved in a complex task or in a procedure which might be disturbed by the question.
Also, the question should not *result* in a complex task for the student. Thus, questions appear most suitable if they concern conceptual or simple procedural knowledge during the expository phase of the education.

2.2.2 Monitoring

When the student has to integrate knowledge, Eg. in solving problems, questioning is too disturbing and data can only be acquired more passively, by monitoring. Monitoring is data-driven: while the student is performing the task the diagnostic tool tries to tap working memory for as much information as possible. The diagnostic tool should unobtrusively stimulate the student to externalize results of sub-tasks which, then, hopefully are interpretable [15]. Of course, one could partition the complex task into subtasks by repeatedly asking questions, but in that case diagnosis of the student's cognitive *control* structures, which are important to solve the problem at hand, is impossible because the questions themselves tend to steer the student in some direction.

To be useful as diagnostic aids outside the research setting, that is in actual education, computerized tools which stimulate the student to externalize the contents of working memory should be useful to the student him/herself. The design should be such that the student can use the external working memory tools in a simple and natural way. If a piece of paper is a simpler alternative, the student is not going to use the electronic tools. For instance, unnecessary typing will frustrate the use of these tools.; it should be easy to copy information to the scratchpad. By having other tools which can perform non-relevant tasks like calculators or formula evaluators embedded in the scratchpad environment, the student will hopefully experience the extended scratchpad as alleviating his/her problem solving task and will use it. Of course the tools should only take over those subtasks that are not of interest for diagnosing the main task at hand. Typically, these subtasks are ones which may be assumed to be mastered, but are time-consuming and error-prone. An example of such a subtask might be 'calculating' in a domain of teaching of a more complex nature, Eg. thermodynamics.

In a research setting, however, it is possible to force the student to use certain tools (Eg by forbidding to make notes on paper).

2.3 Kinds of externalized working memory tools.

Basically, 3 types of monitoring can be effectuated:

1. Monitoring an action type by noticing that a certain tool (Eg. a calculator) is being used.
2. Monitoring of intermediate results on a scratchpad. A problem here is that, if the student does not jot down a particular intermediate result, it can not be inferred that the student didn't reach this result.

Another problem is that if some form of automatic logbooking is implemented, the student will usually not enter intermediate results on the scratchpad.

3. Monitoring that an intermediate result is used as input for a new action. If some intermediate result is used as input for a tool (Eg the calculator) this is easily recognizable. If the action itself is not recognizable, because there is no tool for it, the only thing that might be deduced is that the student used the result. This deduction might be made if the intermediate result was stored in a lookup table with one item visible at a time and the student looks the particular item up. This implementation of a part of the interface conflicts somewhat with the idea of a scratchpad but it can be used in a research-setting.

   The interpretation of an act of inspection of an intermediate result by the student is of course rather fuzzy. It is by no means certain that after inspection of a intermediate result the student will actually use this result in his/her next action. It can be compared with the interpretation of eye-movement recording data.

2.4 Level of structure and abstraction

One aspect of external working memory tools is the amount of pre-structuring that is required to enable parsing of the student's actions. For instance, it might be preferable to represent domain objects on a scratchpad in their canonical form rather than in concrete form (Eg. pointmass instead of fat-man), to allow for diagnosis of the student's capabilities of transforming the problem from its concrete to its symbolic form. However, forcing structure onto the scratchpad decreases the amount of a student's expression possibilities and also might give implicit hints to the student. Simultaneously the technical problems of parsing the student's actions decrease. The major design decisions concern the amount of pre-structuring and the level (or levels) of abstraction on which entities of the domain are represented. Structure generally shifts the task of the student from generating to selecting. Especially if it is unknown what the relevant cognitive structures are (like generally in a research setting) it seems advisable to have the scratchpad as open-ended as possible so that the student can make any note he/she wants. If the design requires selection, then all viable (correct and wrong) alternatives should be available to the student to select from.

2.5 Research Findings

2.5.1 previous research

Formal research concerning the design and the use of electronic scratchpads in the framework of diagnosis is not abundant.
In expository teaching, scratchpads might look like grids upon which a semantic net can be built by the student. Major concepts, which they detect in the educational material presented to them, can be placed upon the scratchpad and attributes can be filled in. Relations that they detect can be made explicit by connecting the relevant concepts. Such a scratchpad should be quite helpful in tracing the cognitive development of the student. To our knowledge this approach hasn't been used yet.
However, in the context of problem solving (once the expository phase is over) some research has been done. The scratchpad in this research was embedded in a thermodynamics coach as a separate window with a number of canonical objects from the domain of thermodynamics (like pistons and containers). These objects could be dragged on to the scratchpad. In this way intermediate representations of the problem, which was presented in text on another window, could be built. It was shown that with a proper design of the scratchpad (and the explicit exclusion of paper near the machine) the vast majority of procedural steps that the student takes during problem solving can be detected [12]. The amount of structure of the scratchpad was rather large: the primitive elements were canonical objects in the domain, which had (invisible) hooks that allowed them to interconnect selectively to prespecified places on prespecified other canonical objects [13]. A parser was able to semantically interpret the drawings made by the student. Since the environment was intended to teach the procedures to solve problems rather than the domain knowledge in the domain of thermodynamics, emphasis in the diagnosis was on detection of (premature) solution steps. The diagnostic power for this type of cognitive structures was greatly improved by the scratchpad. Although the research only used problems in the domain of thermodynamics we feel that the results can be generalized to all domains where intermediate representations of a problem are required to solve it.

2.5.2 Present research goals

Present research is aimed at getting at a general design strategy for these scratchpads, in different domains .

2.5.3 Method

As a first approach, it was decided to start with an analysis of natural notes. As new domain, the use of psycho-diagnostic skills was chosen. This domain was expected to be quite different from the domain of thermo-dynamics, but it might be typical for other domains where diagnosis is essential.

A traditional CAI program has been developed which presents students with case material (verbal report) from which the students have to infer a preliminary diagnosis and additional required tests. After the data for these additional tests are supplied the student has to select the final diagnosis. 32 students participated in the course while making notes on completely blank pieces of paper. The course had a log-book already built in so that the students could look up all raw data that had been presented.

To score the notes a scoring scheme was designed on the basis of a task analysis of the task that the students performed (appendix I). Each utterance on the note would be classified according to this scheme while simultaneously classifying the form in which the utterance appeared (Eg. free text, drawing, numbers, arrows etc) . Utterances which could not be scored were the primary focus of our attention because they might represent a natural need which we did not anticipate.

2.5.4 Results (Thus far[1])

In total the notes on 82 problems were scored. There is an enormous intersubject variability and also intra subject variability when it comes to types of information written down. Some students wrote down all raw data in some cases while in other cases they only wrote down the transformed and partially interpreted data. Mostly text was used (95%), even for data that subjects had been trained to analyze graphically. In total, 38 utterances were found which could not be scored. From these, a few concerned non relevant tasks like the use of the CAI-program and messages that the student intended to mail to the teacher. The remaining concerned tasks that were relevant but were not anticipated in the task analysis like the evaluation of the final diagnosis and the task of writing a

[1] At the moment of this writing 70% (57 notes) of the material has been scored.

response to the (hypothetical) doctor that had sent the patient. Further annotations were found which had to do with the certainty of intermediate conclusions or the severity or importance of the symptoms / conclusions. Finally control-notes were observed. These notes concern elements of the control structure that steers the diagnosis. For instance, there are rules which require a minimum number of indications before a certain conclusion can be reached. Students might write down (parts of) these rules. These can be seen as reminders.

Currently design consequences of these findings are considered. The annotations that students make concerning the results of certain sub-tasks suggest that, for each result which is jotted down on the electronic scratchpad, it should be possible to put an electronic "sticky note" on it. The annotations dealing with the control-knowledge suggest that the students might need a look-up table where they can easily find this knowledge (and which would help the program to diagnose that the student is considering to use certain control knowledge).

The results further show that the vast majority of utterances can be anticipated by doing a task-analysis. However this empirical approach does not provide us with specific new suggestions how to implement the scratchpad. This will be the focus for subsequent research.

III Simulation

If the student, while solving a problem, builds a (proper) description of the problem on the scratchpad, this description could in principle be used in a simulation. For instance, if the student has to solve a problem in the domain of electricity - say Ohm's law applied to parallel and serial resistors - he/she might make a drawing of the circuitry on the scratchpad. If the problem at hand is to find the potential over one of the resistors the student might want to check his/her answer by having the circuitry model which was built in the process of problem solving actually simulate the simple laws of electricity. Thus there appears to be an intrinsic relation between the scratchpad and simulation. One could consider a simulation environment to be a 'runnable' scratchpad.

Therefore one of the things to keep in mind in designing scratchpads is whether the design is usable as a simulation environment. For simulations too the question of the abstraction level at which objects and their relations should be represented is relevant. Eg. in the electricity model the simulation could be very concrete with electrons running through wires. It can be argued that the level of abstraction should not be fixed but should adapt itself to the actual student model. (Adherents of the complete freedom approach would make different levels of abstraction accessible and let the student choose).

Running simulations in the domain of for instance electronics requires a deep model about how electronic components work. So, if a scratchpad is designed in connection to electronic trouble-shooting and this scratchpad has to be runnable the deep model should be implemented too.

It appears rather problematic to formulate a proper deep model for the domain of psycho-diagnosis. Like electronic trouble shooting, psycho-diagnosis is a classification task which might be trained by showing a simulation of the mind-circuitry. Experts in the domain of psycho-diagnosis however do not agree on this 'circuitry'. Therefore this approach of letting the possible simulation environment guide the scratchpad design fails for this domain.

Conclusion

The call for a totally different approach to 'intelligent' tutoring systems in which the traditional split in three modules (a diagnostic module, a tutoring module and an interface module), all centered around the student model, is discarded, becomes louder and louder. There might be many reasons for this. A central one is that current ITS's do not allow for the complex and subtle representations of student's cognitive state. There are several related issues, like the lack of specific theories of learning and the empoverished communication channel, which might justify giving up on student models at least for practical purposes. In this paper we described the underlying ideas and some initial results of a research program aimed at the enrichment of the communication channel in order to enable more detailed cognitive diagnosis. It obviously is still uncertain that even with a far better communication channel, far better student models and a far better theory of learning, ITS could live up to our original expectations. The present research however promises also to offer a framework for the practical design of simulation environments. Future research on scratchpad design for domains where simulation is relevant like electricity and heat theory is in preparation.

References (incomplete)

[1] Self, J.& Gilmore, D. (1989) The application of machine learning methods to intelligent tutoring systems, In J. Self (ed.) AI and Education.

[2] Elsom-Cook, M. (1990) Guided Discovery Tutoring in M. Elsom-Cook (Ed.) Guided Discovery Tutoring, Paul Chapman Publ., London, pp.14.

[2b] Bierman, D.J. (1986). 'Intelligent' Simulation Environments in Education. *Proc. of the 7th European Conference on Artificial Intelligence,* Part II- 84-90. Brighton, july 1986.

[3] New Directions for Intelligent Tutoring Systems (Portugal, 6-10 October 1990)

[4] Erickson & Simon, _

[5] Goldstein,I.P. (1982). The Genetic Graph: a representation for the evolution of procedural knowledge. In D. Sleeman & J.S. Brown (Eds). Intelligent Tutoring systems. Academic Press, New York, pp.69.

[6] Ohlson

[7] Kass, 1987. The role of User Modelling in ITS. Technical Report MS-CIS-86-58. Dept. of Computer and Information Science, Univ. of Pennsylvania, Philadelphia, PA.

[8] Sleeman,D. , Kelly, A.E., Martinak, R., Ward, R.D., Moore, J.L. (1989). Studies of diagnosis and remediation with High School Algebra Students. Cognitive Science, 13, 551-568.

[9] Anderson, J.R., Conrad, F.G. & Corbett, A.T (1989) Skill acquisition and the LISP Tutor. Cognitive Science, **13**, 467-505.

[10] Kamsteeg, P.A. & Bierman, D.J. (1989). *Teacher thinking, student thinking: Think aloud protocol analysis in tutoring systems research.* In: Peter Goodyear (Ed.) Teaching Knowledge and Intelligent Tutoring. Ablex Corporation, Norwood, NJ. chapter 9.

[11] Nessen, E. (1989) SC-UM: User modelling in te SINIX consultant. Applied Artificial Intelligence, 3: 33-44.

[12] Bierman, D.J. & Anjewierden, A.A.(1986). The use of a graphic scratch pad for students in 'Intelligent' CAI., *Proc. of the 27th Conf.*

of the Association for the development of Computer Based Instructional Systems, New Orleans, feb. 1986, 68-71.

[13] Barnard & Sandberg. (1989) Where is the interaction in ITS?, aera paper.

[xx] van Lehn (in Polson & Richardson)

[