JOHNSON GRANT
IN-61

(NASA-CR-188177) AN ENDORSEMENT-BASED APPROACH TO STUDENT MODELING FOR PLANNER-CONTROLLED INTELLIGENT TUTORING SYSTEMS Final Report, Jun. - 31 Dec. 1990 (Houston Univ.) 31 p CSCL 09B H1/61

N91-21739

Unclas
0007299

An Endorsement-Based Approach to Student Modeling for Planner-Controlled Intelligent Tutoring Systems

William R. Murray
FMC Corporation

December 31, 1990

Cooperative Agreement NCC 9-16
Research Activity No. ET.24

NASA Johnson Space Center
Mission Operations Directorate
Training Division

Research Institute for Computing and Information Systems
University of Houston - Clear Lake

T·E·C·H·N·I·C·A·L R·E·P·O·R·T

The RICIS Concept

The University of Houston-Clear Lake established the Research Institute for Computing and Information systems in 1986 to encourage NASA Johnson Space Center and local industry to actively support research in the computing and information sciences. As part of this endeavor, UH-Clear Lake proposed a partnership with JSC to jointly define and manage an integrated program of research in advanced data processing technology needed for JSC's main missions, including administrative, engineering and science responsibilities. JSC agreed and entered into a three-year cooperative agreement with UH-Clear Lake beginning in May, 1986, to jointly plan and execute such research through RICIS. Additionally, under Cooperative Agreement NCC 9-16, computing and educational facilities are shared by the two institutions to conduct the research.

The mission of RICIS is to conduct, coordinate and disseminate research on computing and information systems among researchers, sponsors and users from UH-Clear Lake, NASA/JSC, and other research organizations. Within UH-Clear Lake, the mission is being implemented through interdisciplinary involvement of faculty and students from each of the four schools: Business, Education, Human Sciences and Humanities, and Natural and Applied Sciences.

Other research organizations are involved via the "gateway" concept. UH-Clear Lake establishes relationships with other universities and research organizations, having common research interests, to provide additional sources of expertise to conduct needed research.

A major role of RICIS is to find the best match of sponsors, researchers and research objectives to advance knowledge in the computing and information sciences. Working jointly with NASA/JSC, RICIS advises on research needs, recommends principals for conducting the research, provides technical and administrative support to coordinate the research, and integrates technical results into the cooperative goals of UH-Clear Lake and NASA/JSC.

AN ENDORSEMENT-BASED APPROACH TO STUDENT MODELING FOR PLANNER-CONTROLLED INTELLIGENT TUTORING SYSTEMS

Preface

This research was conducted under the auspices of the Research Institute for Computing and Information Systems by William R. Murray of FMC Corporation. Dr. Glenn Freedman, Director of SEPEC, served as RICIS research representative.

Funding has been provided by Training Division, of the Mission Operation Directorate, NASA/JSC through Cooperative Agreement NCC 9-16 between NASA Johnson Space Center and the University of Houston-Clear Lake. The NASA technical monitor for this activity was Barbara Pearson, Systems/Elements Office, Space Station Training Office, NASA/JSC.

An Endorsement-based Approach to Student Modeling for Planner-controlled Intelligent Tutoring Systems

William R. Murray
FMC Corporate Technology Center
1205 Coleman Avenue, Box 580
Santa Clara, CA. 95052
(408) 289-0456
murray@ctc.fmc.com

RICIS Final Report
Ricis research activity #ET.24
June 1990 through December 31st, 1990
(NASA Cooperative Agreement NCC9-16)

Abstract

This report describes an approach to student modeling for intelligent tutoring systems based on an explicit representation of the tutor's beliefs about the student and the arguments for and against those beliefs (called *endorsements*). A lexicographic comparison of arguments, sorted according to evidence reliability, provides a principled means of determining those beliefs that are considered true, false, or uncertain. Each of these beliefs is ultimately justified by underlying assessment data.

The endorsement-based approach to student modeling is particularly appropriate for tutors controlled by instructional planners. These tutors place greater demands on a student model than opportunistic tutors. Numeric calculi approaches are less well-suited because it is difficult to correctly assign numbers for evidence reliability and rule plausibility. It may also be difficult to interpret final results and provide suitable combining functions. When numeric measures of uncertainty are used, arbitrary numeric thresholds are often required for planning decisions. Such an approach is inappropriate when robust context-sensitive planning decisions must be made. Instead, the ability to examine beliefs and justifications is required. This report presents a TMS-based implementation of the endorsement-based approach to student modeling, compares this approach to alternatives, and provides a project history describing the evolution of this approach.

1. Introduction - limitations of numeric student models

This report describes a symbolic (i.e., non-numeric) means of coping with uncertainty in student modeling. Rather than represent the uncertainty of the tutor's beliefs with numeric degrees of confidence the student model explicitly records arguments (called *endorsements* in [Cohen 85]) for and against each belief. No numeric combining functions or interpretation of numbers is required. Instead the different kinds of arguments are compared based on the reliability of their evidence to decide if belief or disbelief in a proposition is justified.

Previous research on the Blackboard Instructional Planner [Murray 90], a planner-controlled tutor for teaching troubleshooting for a complex hydraulic-electronic-mechanical device, illustrated some of the shortcomings of numeric student models. That research motivates the research presented here. Before reviewing the earlier research, we briefly consider the role and demands placed on the student model in both planning and non-planning (i.e., opportunistic) tutors.

In opportunistic tutors the student model may be used to decide what issues to discuss (e.g., WEST [Burton and Brown 82]) or what topics to explore (e.g., MENO-TUTOR [Woolf 84]). Other uses are problem selection (e.g., BIP [Barr 76]) or hint generation (e.g., WUSOR-II [Carr 77]). Frequently diagnostic student modeling is used to model a student's problem solving and its correctness (e.g., PROUST [Johnson 86]).

The student model for a planner-controlled tutor must not only address these issues but others. A sophisticated student model is needed to track plans and allow customized plan generation based on an initial assessment of the student. It must interpret different kinds of *assessments* (student data) such as the student's background, any student self-assessment, test questions, any instructor assessment, student-initiated questions, and student problem-solving actions. Typically, the student model for opportunistic intelligent tutoring systems will handle a much more limited range of assessment data and have fewer responsibilities. For example, those tutors that act as problem-solving monitors (the most common paradigm) predominantly focus on assessing problem-solving actions for hint generation and future problem selection (e.g., IMTS [Towne et al 89]).

The student model of the Blackboard Instructional Planner illustrates some of the shortcomings of numeric student models and how they can limit tutor capabilities. That

student model is an overlay [Carr and Goldstein 77] of a semantic net representation of domain concepts. Associated with each concept is a number representing the tutor's confidence that the student has acquired the concept. The numbers are initialized from a pre-instruction questionnaire according to inferred cognitive stereotypes [Rich 79] and later adjusted according to the student's test and problem-solving performance.

With this numeric approach the tutor tended to either replan at the wrong times or not replan when it should. The problem was that planning decisions could only rely on these numbers, which were compared to threshold values. Replanning can easily go awry because of the difficulty of determining precisely how to adjust the numeric weights to integrate the different kinds of assessment data, and because of the arbitrary nature of the three planning thresholds that were used. One threshold measured when a concept was learned, another when it was forgotten, and a third when an instructional activity was making insufficient progress. When the thresholds and updates were adjusted conservatively the planner tended not to replan when it should. When they were adjusted less conservatively the planner tended to replan when it should not.

These problems led to the development of an *endorsement-based student model* (ESM). The remainder of this report describes the endorsement-based approach and its evolution, compares it to alternatives, and argues that it is particularly appropriate for planner-controlled tutors.

2. The endorsement-based approach to student modeling

The key aspects of the ESM are:

1. Explicit representation of tutor beliefs and their endorsements- propositions represent the tutor's beliefs about the student's skills along with arguments for and against those beliefs.

2. Inheritance of endorsements - an ISA hierarchy represents the subject matter. The ESM uses the hierarchy to represent the degree to which a student has generalized a skill. Endorsements for a *generic skill* (a skill that can be applied to all members of a class) are inherited down the hierarchy towards subclasses (or instances) representing more specific skills. Endorsements against a generic skill are propagated up towards superclasses representing more general skills.

3. *Wide variety of assessments* - several different kinds of information, varying both in specificity, source, and reliability are incorporated.

4. *Lexicographic comparison of arguments* - endorsements are sorted into equivalence classes according to reliability. This ordering allows lexicographic comparison of pro and con arguments. The result of the comparison is a label for each belief - **believed-true, believed-false, unknown** (no data), or **uncertain** - and an indication of the decisive argument, if any, that indicates how well justified a belief is.

5. *Consistency between endorsements and labels* - the student model explicitly represents the justification for each endorsement and tutor belief. All justifications are ultimately grounded in *assessments* (student data). If endorsements become invalid or labels change then consistency is maintained between derived endorsements and any labels that depend on them.

These features are best illustrated by examples.

2.1 Examples of endorsement-based student modeling

This section presents a scenario demonstrating the endorsement-based approach. Assume the student is learning to troubleshoot a device and must first learn how the device and its individual parts operate. Figure 1 shows a class hierarchy of parts of the device. Classes of parts are connected to subclasses by solid arrows. These in turn are connected to part instances by dotted arrows. The tutor's goal is to ensure that the student understands the operation of all of the device's hydraulic valves. This goal (a generic skill) is represented by the proposition SK(*op*, *hydraulic valves*).

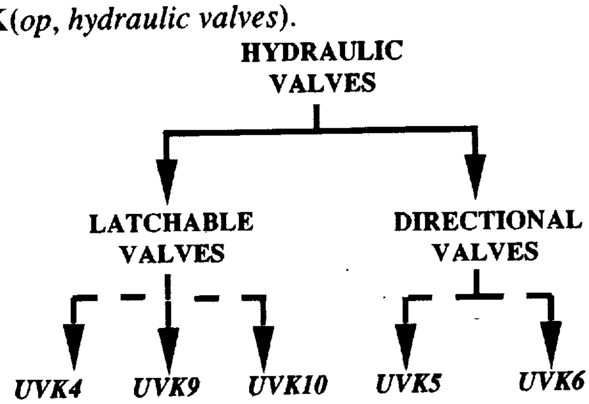

Figure 1. Class hierarchy of device parts

SK stands for "student knows" (a notation adopted from [Peachey and McCalla 86]). The general form is SK(*skill, node*) where *node* is either a class or instance. SK(*op, UVK4*) is believed true when the tutor believes the student understands the operation of the UVK4 valve. SK(*op, latchable valves*) is believed true when the tutor believes the student understands the operation of *all* the latchable valves - UVK4, UVK9, and UVK10. So, if SK(*op, UVK4*) was believed false then SK(*op, latchable valves*) would also have to be believed false.

The scenario below illustrates how an endorsement-based student modeling system can cope with several different kinds of assessments, can infer new beliefs based on inheritance (the links in Figure 1), and can retract beliefs that are no longer justified. It also shows how pro and con arguments are compared.

Table 1 summarizes the scenario. The top row lists the labels of the five left-most nodes in Figure 1. These nodes are the only ones whose labels change in this scenario. In the top row "Latch" and "Hydra" stand for "Latchable Valves" and "Hydraulic Valves" respectively. Below each node are two columns marked + and -. For each node x all pro arguments for SK(op, x) appear in the + column and all con arguments appear in the - column. The letters are abbreviations for different kind of arguments. For example, D stands for a default belief. The other kinds of arguments and their abbreviations are shown in Table 2; they will be explained as the scenario unfolds. Boldface arguments are the *deciding arguments* in determining the label of propositions, i.e., they cast the deciding vote for or against a proposition. If an argument is in boldface underneath a - column with label *node* then SK(*op, node*) is **believed-false**. Similarly, a boldface argument in the + column indicates a label of **believed-true**.

Initially the tutor assumes that the student does not know how the valves operate. These default assumptions are indicated by the three Ds in line 1. Since there are no arguments to oppose these each node[1] is labeled **believed-false**. The remaining two nodes receive the labels **unknown** as no arguments are recorded for them yet.

[1]Actually for each *node* the predicate SK(*op,node*) is assigned the label. Nodes are referred to instead of their corresponding SK predicates for succinctness.

Event	UVK4 +	UVK4 -	UVK9 +	UVK9 -	UVK10 +	UVK10 -	Latch +	Latch -	Hydra +	Hydra -
1. Defaults		D		D		D				
2. Self-assess		D		D		D	ST			
3. Inherit beliefs	IB	D	IB	D	IB	D	ST			
4. T/F question	IB	D T/F	IB	D	IB	D	ST			
5. M-C question	IB	D T/F	IB M-C	D	IB	D	ST			
6. S/A question	IB	D T/F	IB M-C	D	IB	D S/A	ST			
7. Trend - samples	IB	D T/F	IB M-C	D	IB	D S/A	ST	TR		
8. Retract inherited	~~IB~~	D T/F	~~IB~~ M-C	D	~~IB~~	D S/A	ST	TR		
9. Propagate disbelief		D T/F	M-C	D		D S/A	ST	TR		PR
10. Tutor presentation	TU	D T/F	M-C	D		D S/A	ST	TR		PR
11. Retract arguments	TU	~~D~~ ~~T/F~~	M-C	D		D S/A	ST	~~TR~~		~~PR~~
12. Inherit as before	TU IB		M-C IB	D	IB	D S/A	ST			
13. Tutor presentation	TU IB		M-C IB TU	D	IB	D S/A	ST			
14. Retract arguments	TU IB		M-C IB TU	~~D~~	IB	D S/A	ST			
15. Tutor presentation	TU IB		M-C IB TU		IB TU	D S/A	ST			
16. Retract arguments	TU IB		M-C IB TU		IB TU	~~D~~ ~~S/A~~	ST			
17. Trend - labels	TU IB		M-C IB TU		IB TU		ST LT			
18. Trend - labels	TU IB		M-C IB TU		IB TU		ST LT		LT	

Table 1. A summary of PRO and CON arguments for the scenario

Line 2 shows the student's self-assessment (ST) of his knowledge of the operation of latchable valves. This is recorded as a pro argument under Latch as the student claims to understand how this kind of valve operates. The node Latch now receives the label **believed-true**.

Line 3 represents three new endorsements inferred by inheritance. As shown in Figure 1, if the student understands how latchable valves operate then he should understand how UVK4, UVK9, and UVK10 operate. Each new inherited belief (IB) overrides the previous default (D) beliefs, changing the labels from **believed-false** to **believed-true**.

As shown in Table 2, each endorsement is classified into an *endorsement reliability class* according to the kind of endorsement and whether it is positive or negative. Table 2 lists the different kinds of endorsements used in the scenario, in order from most credible to least credible. Consistent data trends (TR) are considered the most reliable, followed by student claims of ignorance (ST-) and then specific counterexamples to generic skills (PR-). Tutor presentations are considered the next most reliable evidence (TU+), followed by arguments to label parent nodes the same as the majority of their children (LT). A student's claim to know some skill (ST+) is considered less reliable, but answers to individual questions are even more suspect. However, a given short answer question (S/A) is considered more reliable than a multiple choice question (M-C), which in turn is considered more reliable than a true false question (T/F). The weakest beliefs are those based on inheritance (IB+) or defaults (D).

Continuing the scenario, the tutor asks one question on each latchable valve in lines 4, 5, and 6. Only the second question is answered correctly. As arguments based on test data are more strongly believed than inherited beliefs or default beliefs the labels for UVK4 and UVK10 are now **believed-false** once more.

A new kind of argument, called a *data trend*, is inferred by the student model from these three questions. A data trend is only inferred based on test questions or other kinds of student performance, and only when a clear majority of the data is pro or con. A data trend is considered the most reliable kind of endorsment since it is based on multiple snap-shots of student performance. Individual questions (T/F, M-C, or S/A) are more liable to noise - lucky guesses, confusion, typos, etc.

A negative data trend is added as a con argument to the node Latch in line 7 as two out of three questions on latchable valves were missed. It overrides the student's self-assessment causing the label of Latch to become **believed-false**. The previous inherited beliefs, which depended on Latch being labeled **believed-true**, are now retracted as shown in line 8 by a strike through each retracted belief (IB).

Class	Symbol	Description
Data trends	TR	Consistent trends in student performance
Negative student self-assessment	ST-	The student says he does not know something
Propagated disbelief	PR-	Argue that skill x cannot be known for class y as it is not known for class (or instance) z and y includes z
Tutor presentation	TU+	Argue that skill is known as tutor has covered it
Label trends	LT	Assign class X the same label as most of its children
Positive student self-assessment	ST+	The student says he knows something
Short-answer	S/A	The student answers a single short-answer question
Multiple-choice	M-C	The student answers a single multiple-choice question
True-false	T/F	The student answers a single true or false question
Inherited belief	IB+	Argue that class (or instance) y is known as its superior class x is known
Default belief	D	Default belief

Table 2. Endorsement reliability classes, in order of believed reliability

If the student does not understand how latchable valves operate then he cannot understand how hydraulic valves operate. That is why a **PR** (for propagated disbelief) argument is added to the minus (con) column under Hydra in line 9. That causes Hydra to become labeled **believed-false.**

Now the planner decides to review the operation of the valves. Lines 10, 13, and 15 indicate these tutor presentations. After a tutor presentation prior test results or default beliefs indicating lack of the knowledge covered are no longer necessarily valid and are retracted. Such retractions occur in lines 11, 14, and 16. When the TR argument is retracted in line 11, the label for Latch is recomputed. It becomes **believed-true** again, which in turn causes the inherited endorsements (IB) for UVK4, UVK9, and UVK10 to be reintroduced in line 12.

After the final presentation a different kind of trend is inferred called a *label trend*. The earlier data trend depended on test data. This second kind of trend reflects a trend among the labels (not data) of the children of a node. The labels must be justified by arguments that are at least as strong as tutor presentations, which is why no label trend was inferred from the defaults in line 1. Lines 17 and 18 show label trends added to Latch and Hydra, assuming that Directional Valves (see Figure 1) was already labeled **believed-true** because of a sufficiently strong argument.

The label trend endorsement (LT) for Hydra causes SK(*op, hydraulic valves*) to become labeled **believed-true**. This completes the scenario as the tutor's goal is now achieved.

Note that the strength of a belief can be measured by the reliability of its deciding argument. For example, belief that the student knows how UVK9 operates increases from line 3 (IB) to line 5 (M-C) to line 13 (TU) as shown by the ordering in Table 2. If the planner had wanted stronger justification before believing its goal was achieved, it could have required a stronger deciding argument for SK(*op, hydraulic valves*), such as an argument of the data trend class. In that case further questioning of the student after the tutor presentation would be required to gather such data.

The key points illustrated in this scenario are:

1. Many different kinds of assessments are handled in the ESM - three different kinds of test questions were used along with default beliefs, inherited beliefs, student self-assessment, and changes inferred from tutor presentations.

2. No numeric degrees of belief are required for evidence - the ordering of endorsements according to their reliability is sufficient.

3. No numeric combining functions are required - all arguments are retained unless later retracted. Unlike numeric approaches, each argument's contribution to a label can always be determined.

4. Inferred beliefs reflect the inheritance hierarchy of the subject matter - the inheritance in Figure 1 is enforced by the ESM. The ESM uses the class hierarchy to represent the extent to which the student has generalized a skill.

The lexicographic comparison routine was only demonstrated in the scenario with simple cases. In general an arbitrary number of arguments can be compared. They are first sorted into equivalence classes of reliability, such as those shown in Table 2.[2] Then, starting with the most reliable class the pro and con arguments in that class are paired. If one or more pro arguments are left over then the label for an SK proposition in question

---

[2]Of course other kinds of assessments, evidence reliability classes, class orderings, and assessment to class mappings can be used in an ESM. Table 2 illustrates just one set of choices.

will be **believed-true**. If one or more con arguments are left over it will be **believed-false**. If all arguments can be paired then the next most reliable class is considered to break the tie. If a tie is never broken then the label is **uncertain**. If there are no arguments at all it is **unknown**.

2.2 Implementation

The ESM is implemented in a layered fashion over a justification-based truth[3] maintentance system (JTMS). It also uses a simple forward-chaining rule-based inference engine and assertional database called the JTRE (Justification-based Trivial Rule Engine) that makes use of the JTMS. These two systems were obtained from the documentation and code of [De Kleer et al 89] and were developed prior to the research described here.

The role of the JTMS is to ensure consistency between inherited and propagated beliefs and those they depend on, and to notify the lexicographic comparison routines that ESM labels need to be recomputed when such beliefs are retracted or previous endorsements are un-OUTed (i.e., reintroduced). The assertional database (JTRE) stores propositions representing SK predicates, their ESM labels, and the pro and con arguments that justify the labels. Forward-chaining JTRE rules carry out the propagation and inheritance of endorsements and invoke the lexicographic comparison routines when new arguments should be considered.

3. Related work in student modeling and uncertain reasoning

Now we consider related work in student modeling and uncertain reasoning. Numeric and symbolic approaches to uncertainty are discussed for both ITS and non-ITS applications.

3.1 Numeric approaches

Possible numeric approaches to representing uncertainty include certainty factors [Shortliffe and Buchanan 75], Dempster-Shafer theory [Shafer 76], fuzzy logic [Zadeh 78], or use of Bayes' Rule. These approaches are discussed in [Bonissone 87], along with the following problems:

---

[3]Justification-based truth maintenance systems are distinguished from other kinds of TMS by having nodes that are either IN (believed) or OUT (not believed). The only kind of constraints that can be expressed are logical implications. In contrast, an ATMS (assumption-based TMS) has labels indicating when nodes will be believed (i.e., what sets of assumptions must be true) and an LTMS (logic-based TMS) allows even more general logical constraints (e.g., either x is true or y but not both) [De Kleer et al 89].

1. Inability to distinguish uncertainty from lack of evidence - if a single number is used to represent degrees of belief then typically 0 will represent both a complete lack of data and uncertainty due to a balance of conflicting data.

2. Normalizing PRO and CON evidence - if on the other hand two numbers are used so the distinction above can be made, then the amount of evidence for and against a belief may be normalized. This results in disproportionate weighting of a single piece of evidence that contradicts several other pieces of evidence.

3. Difficulty of assigning numbers - all of these approaches require numbers to be assigned to indicate the reliability of each piece of evidence.

4. Difficulty of interpreting numbers - with the exception of approaches based on Bayes' Rule, it can be hard to provide consistent and meaningful semantics to the numbers assigned to derived beliefs.

5. Obscuring the source of derived beliefs - no records are maintained showing how numeric degrees of belief have been accumulated from different sources of evidence.

6. Arbitrary combining functions - there may be several consistent ways of combining conflicting data reflecting conservative, optimistic, or moderate viewpoints.

7. Stringent assumptions - Bayes' Rule can be simplified given strong requirements regarding the mutual independence of each piece of evidence and the exhaustivity and disjointness of the hypotheses. Unfortunately, these requirements, or the need for a large number of conditional probabilities (if the simplifying requirements are lifted), often render the approach impractical.

Formal approaches to handling uncertainty are infrequently used in intelligent tutoring systems, with some exceptions. Certainty factors have been used in GUIDON [Clancey 87] but the initial assignment and subsequent updating within tutorial rules is somewhat arbitrary. A different approach, based on fuzzy logic, is being applied to the TAPS intelligent tutoring system [Derry 89] to handle imprecision in measuring the correctness of student inputs.[4]

---

[4]In contrast, there is no uncertainty in the assessments the ESM receives. Instead there is uncertainty in deciding which tutor beliefs are justified when there are conflicting assessments.

Frequency of use measures or parameter adjustment approaches, neither based on probability theory, are the most commonly used numeric approaches to uncertainty in ITS. WEST [Burton and Brown 79] and WUMPUS [Stansfield 76] rely on the frequency of use approach. They measure how often a skill was used compared to the numbers of times it could have been used. Examples of the parameter-adjustment approach include the Blackboard Instructional Planner (discussed earlier), Kimball's integration tutor [Kimball 82], MENO-TUTOR [Woolf 84], and the user modeling system GRUNDY [Rich 79].

3.2 Non-numeric approaches

Typical non-numeric symbolic student models used to represent student problem-solving strategies or knowledge include

1. *Procedural networks* - such as BUGGY's [Burton 82] procedural network to represent subtraction skills.
2. *Rules and mal-rules* - such as the rules of LMS [Sleeman 83] representing correct and incorrect linear algebra simplifications.
3. *Plan and bug libraries* - such as the loop plans and bug recognizers of PROUST [Johnson 86] used to understand Pascal programs.
4. *Rule application heuristics*- such as ACM's [Langley et al 84] representation of production rules for subtraction. The heuristics the student uses in choosing which rule to apply next are induced from student solutions.

These student models go beyond overlays by representing incorrect beliefs a student may have. However, except for ACM, they typically do not address issues of uncertainty other than by applying averaging or other statistical techniques to reduce the effects of noise in data [Wenger 87]. The kind of knowledge they focus on is primarily the representation of subskills required to perform an algorithmic, procedural, or problem-solving task.

As mentioned earlier, the ESM is built over a truth maintenance system (TMS) to maintain consistency between endorsements and labels. In general, TMSs and nonmonotonic logics can be used to represent tutor assumptions about the student, and detect contradictions that arise when tutor expectations do not match student performance (as in [Fum, Giangrandi, and Tasso 90]). The faulty assumptions can then be retracted and the consistency of the student model restored. [Huang 90] adopts this kind of approach to enforce default cognitive stereotypes and switch stereotypes when expectations are contradicted.

The difficulty with TMSs (without extensions) are the restricted labels of TMS nodes. As there will frequently be conflicting justifications for and against any particular belief about the student the TMS will have to resolve or tolerate many contradictions. Resolving the contradictions may require too much student interrogation at an inappropriate time. Alternatively, the beliefs can just be considered unknown, but that is not much use to the planner.

Cohen first presented endorsement theory in a portfolio recommendation program called FOLIO [Cohen 85]. That program weighed pro and con arguments for various investments and intermediate conclusions, such as whether a client would accept high risk investments, in making its recommendations.

CYC [Guha and Lenat 90] uses a similar approach called *argumentation*. In this approach alternative defaults are compared and specific preference relationships between defaults (e.g., assumption A is preferred to assumption B) are used to decide which is the most compelling. The endorsement based approach is similar except it uses a less flexible means of weighing arguments.

4. Project history

We briefly review this project's history here; a more detailed discussion appears in the appendix. As noted in the introduction, this project evolved from shortcomings of the Blackboard Instructional Planner arising from the numeric student model it used. The original proposal submitted to RICIS and AFHRL proposed investigating the application of TMSs to improve the student model. Once the project began it became apparent that a TMS alone was insufficient and further extensions to support weighing conflicting evidence were required. This led to the endorsement-based approach discussed in the design document submitted to RICIS and AFHRL.

Once implementation began, five prototype ESMs were implemented. Their major differences are shown in Table 3. The first prototype used a heuristic measure of the weight of pro and con arguments. It did not use the JTMS or JTRE. The second prototype switched to a lexicographic comparison to weigh evidence. It also incorporated the JTMS and JTRE, but only for use in explaining label assignments and to provide an assertional database. It did not use the TMS to track dependencies. The third prototype distinguished between *performance samples* (individual test questions) and data trends drawn from performance samples. It also placed evidence superseded by tutor

presentations in a special *shadowed* class to discount its reliability. The next ESM clarified the semantics of the knowledge base, which had been unclear in the previous prototypes. It changed the level at which teaching and assessing was done from concepts to attributes of concepts. It also defined generic skills. The fifth and final ESM used the TMS to maintain dependencies between endorsements and other endorsements that were propagated or inherited, and any labels depending on those endorsements. In this final ESM there is no special class of shadowed data. Instead once data is superseded by tutor presentations it is withdrawn (retracted). The TMS ensures that dependent inferences are also withdrawn. Special JTRE rules recompute labels when endorsements change in this process. For more details of the five ESM prototypes see the appendix.

ESM #	TMS	Clear semantics	Data trends	Comparison method	Retraction
1	NO	NO	NO	Heuristic	NO
2	YES	NO	NO	Lexicographic	NO
3	YES	NO	YES	Lexicographic	Shadowed
4	YES	YES	YES	Lexicographic	Shadowed
5	YES	YES	YES	Lexicographic	YES - TMS retraction

Table 3. ESM prototypes developed during project.

5. Conclusion

This report has described problems with numeric approaches to representing uncertainty in student models. These problems have motivated the development of an endorsement-based approach. An endorsement-based student model (ESM) is particularly suitable for planner-controlled tutors due to the greater demands they place on the student model. These tutors rely on the student model to generate, track, and revise instructional plans. They must query the student model and interpret the results to decide if a current activity has achieved its objective, if a previous objective needs to be reachieved, or if a pending objective has already been achieved. The endorsement-based approach supports these kind of queries by allowing context-sensitive planning decisions to be made that rely on an examination of tutor beliefs and the evidence that justifies them.

The key research contribution of this work is the symbolic approach to uncertainty of the ESM. In this approach the tutor's beliefs about the student's knowledge are represented explicitly. Arguments for and against these beliefs are recorded, and justified in terms of underlying assessments. The ESM weighs these arguments by sorting arguments according to evidence reliability and then performing a lexicographic comparison.

Acknowledgements

I would like to thank Dr. Kurt Steuck of the Air Force Human Resources Laboratory for sponsoring this work under subcontract #063, RICIS research activity #ET.24 (NASA Cooperative Agreement NCC9-16).

Appendix - A more detailed history of the project

This appendix describes the project's history in more detail, focusing on how the ideas presented in this report have evolved. We review changes from the original research proposal, to the design document, and then through the four prototypes leading to the final implementation. The ideas have evolved from applying TMS to student modeling, to applying endorsements, and then to clarifying the representation of the student model, the meaning of the endorsements, and the underlying implementation.

Research proposal

The original research proposal (titled "A Research Proposal: Applying Machine Learning Techniques to Student Modelling and Diagnosis") discussed possible broad applications of truth maintenance systems or algorithmic debugging methods [Shapiro 83] to different components of the Blackboard Instructional Planner (BB-IP). The most specific approach discussed was to represent part-state change rules with JTRE rules that made explicit assumptions that parts were operating correctly. Then if a later observation contradicted a result predicted by the rules then the set of assumptions underlying the contradiction would indicate the possibly faulty parts. The approach would be extended to a student modeling application by adding two different kinds of assumptions: first, that the student knew a rule, and second, that he applied it. Then if the tutor made a prediction that differed from the student's the set of underlying assumptions would indicate the rules the student might not know or might not have applied.

Design document

The design document (titled "Complex Student Modeling for Planner-controlled tutors") proposed replacing the TMS approach with the use of endorsements. The TMS approach was abandoned because of the reasons discussed earlier: first, plausible not purely logical reasoning is required and second, there must be some way of distinguishing different kinds of uncertainty in a more refined way than IN or OUT; or TRUE, FALSE, or UNKNOWN labels. Furthermore, the focus on only identifying the student's knowledge and application of rules that predict device operation appeared too narrow.

The design document proposed compiling a subject matter representation into a student model with multiple links to represent possible propagation paths of endorsements. Part of the complexity would arise from the variety of different kinds of things that could be learned (facts, rules, principles, and procedures). Additional complexity was introduced by allowing several different kinds of links in the subject matter representation such as ISA, PART-OF, INSTANCE, REFINES, CAUSES, and PREREQUISITE. The student model also attempted to represent to what degree a student had learned a concept. Three stages were proposed, based on [Brecht 90] (in turn based on [Bloom 56]), to indicate whether a concept was known factually, analytically, or synthetically. A means of interpreting assessment data was proposed whereby endorsements would be propagated along links according to the student's stage of learning and whether the endorsements were pro or con. A set of rules called *conflict resolution rules* was proposed to weigh conflicting pro and con evidence. A heuristic measure of utility to choose new assessments was also proposed.

Prototypes

Not surprisingly, what was implemented was less complex and did not address all of the issues regarding the different kinds of things that can be learned and their different stages of learning. The compilation of representations and the different levels of knowing a concept were not implemented. It was first necessary to clarify the semantics of the knowledge base, the propagation and weighing of endorsements, and the underlying implementation. The clarification occured through the implementation of five endorsement-based student model prototypes that will be referred to as ESM 1 through ESM 5. ESM 5 is the final implementation discussed in this paper. The differences between these implementations are summarized in Table 3 and discussed in more detail below.

ESM 1: Using heuristics to weigh evidence

The first prototype did not use any truth maintenance system. Rather than explicitly represent propositions a semantic network of concept nodes was created. Each concept node was a record that not only indicated the other concept nodes that it was linked to, but also the pro and con arguments for believing the student had acquired the concept. Each

argument was itself a different kind of record with slots indicating the kind of assessment the argument was based on, when the assessment occurred, what node was originally assessed, and how many links separated the two nodes (source and destination) in the conceptual network. A heuristic evaluation function was used to compute the strength of the pro and con arguments for comparison:

$$\text{Weight} = \sum_{i} \frac{\textit{priority}\,(\text{arg[i]})}{\textit{delay} * \textit{distance} * \textit{direction}}$$

Priority is a number indicating the strength of the underlying evidence. *Delay* is proportional to how long ago the argument's assessment occurred and is at least 1. *Distance* is proportional to how far away in the conceptual network the node originally assessed was and is also at least 1. *Direction* is either 1 or 2 to measure the plausibility of the direction of propagation within the network. It is 1 for pro evidence propagated downward, or for con evidence propagated upwards, as this is consistent with the semantics of inheritance. It is 2 for pro evidence propagated upward as the evidence is weaker that the student knows a parent concept given only that he knows a subordinate concept. It is also 2 for con evidence propagated downwards as the fact that the student does not know some parent concept does not necessarily imply that he does not know any of the parent's children concepts.

The strength of the pro and con arguments was compared to assign node labels. This approach was not very satisfactory as it still relied on numbers and there was no more refined explanation for label assignments other than the results of comparing two numbers.

Other disadvantages were the coarse-grained and ill-defined knowledge representation and the unclear semantics of the propagation of endorsements. These deficiencies led to the next ESM.

ESM 2: Using the JTMS to infer and explain labels

The next prototype added the JTMS to provide improved explanations for label assignments. Propositions were used to represent the conceptual network and its relationships. A lexicographic comparison of pro and con arguments was used for the first time. Each proposition also had a second label (either **low**, **medium**, or **high**) indicating the tutor's confidence in its belief based on the amount of pro and con arguments and the

degree of conflict between the two sets of arguments. JTRE inference rules were now used for propagating endorsements. To simplify matters PRO arguments could only propagate downwards and CON arguments could only propagate upwards.

One problem remaining was how to classify test data. Although test data is more reliable than other kinds of data when clear trends emerge, *individual* test questions are not so reliable due to noise. Thus it was difficult to determine exactly where endorsements based on test questions should be classified. For example, should the student's performance on a particular true/false question be given more or less weight than a student's self-assessment for the same skill? The next ESM addressed this problem.

ESM 3: Distinguishing between weak and strong evidence

ESM 3 created two separate classes of endorsements for data. One was based on data trends obtained from performance samples. The second was based on the performance samples themselves. It included multiple-choice, true-false, or short-answer questions. The advantage of this distinction is that the first class is less susceptible to noise, and thus more reliable, than the second class.

In ESM 3 classes of endorsements are first subdivided into two major classes, one for weak evidence and one for strong evidence. The strong evidence class includes both data trends and performance samples, along with any other arguments directly based on assessment data without propagation. The weak evidence class includes everything else - endorsements based on propagation and *shadowed* endorsements (discussed next).

Shadowed endorsements are endorsements that are considered dated and only marginally relevant now. An endorsement becomes shadowed if it is a con argument and a subsequent tutor presentation covers the same material. The rationale behind shadowing is that the tutor's presentation has substantially increased the likelihood that the student has learned the material so previous assessments to the contrary are no longer relevant. But student learning is not guaranteed by tutor presentations so prior endorsements are not discounted completely. They remain relevant, but are demoted to the class of weak evidence even if they were previously strong evidence.

ESM 4: Clarifying the semantics of the knowledge base

The next prototype clarified the semantics of the knowledge base. Previously the finest-grained item a student could learn was a concept, such as UVK4. That grain size is unsatisfactory as there are many aspects of a concept that can be learned. For example, the student can learn the operation of UVK4, the common faults of UVK4, or the role which UVK4 plays in the operation of the device. Thus it does not really make sense to say that the student knows the concept UVK4 or does not know that concept. Instead we would like to be able to say, for example, that the student has learned how UVK4 operates, but not yet learned what role UVK4 plays or what its common faults are.

A second problem with the previous semantics of the knowledge base was in determining what it means for the student to know a particular skill for a higher-level concept, such as knowing the generic skill *operation* for the class *hydraulic valves*. On the one hand it could mean that the student knows how hydraulic valves operate in general but not that he can necessarily *apply* this knowledge to any particular valve (e.g., UVK10). Or it could mean that the student can apply this knowledge to each hydraulic valve in addition to understanding the common principles of hydraulic valve operation.

To address these ambiguities the grain size of the knowledge base was changed and its semantics clarified. Now each object in a hierarchy could have one or more attributes and these attributes were target skills to be learned associated with domain objects. The class hierarchy of domain objects could then be used to represent to what extent the student had generalized different skills. So SK(*attribute*, *class*) was defined to mean the generic skill in which the student knows SK(*attribute*, *instance*) for each *instance* of *class* (the second of the two meanings given above).

ESM 4 also dropped the second label used to measure the confidence of the tutor's belief as **low**, **medium**, or **high**. Instead, **believed-true** and **believed-false** label assignments were amended to include the determining arguments used to decide lexicographic comparisons. The strength of a belief could then be measured by the endorsement reliability class of the determining argument as discussed at the end of Section 2.1.

ESM 5: Implementing retraction of endorsements & labels

One failing of the last ESM was that when arguments were shadowed any propagated or inherited arguments based on them were not. ESM 5 uses the TMS to maintain consistency rather than adding special rules to ensure that all derived arguments are also shadowed. The advantage of this approach is that all derived arguments depending on superseded assessments are automatically retracted. Special JTRE rules detect when a label needs to be recomputed because one of its endorsements has been retracted.

So in this ESM version there is no shadowing, instead once a tutor presentation teaches attribute *a* of class *c* then all prior assessments showing that the student did not know *a* of *c* are retracted along with any derived conclusions and labels. Labels are recomputed as necessary.

This version of the ESM is the one presented in this paper.

Conference paper

A conference paper describing the final ESM was submitted to IJCAI-91 under the Intelligent CAI subarea of the Principles of AI Applications topic. Acceptance or rejection will not be known until March 20, 1991. This technical report is based upon the conference paper. The only difference is that the paper did not include either the project history contained in Section 4 or this more detailed appendix.

References

[Barr 76] Barr, A.; Beard, M; and Atkinson, R. The Computer as a Tutorial Laboratory. *International Journal of Man-Machine Studies* (8):567-596. 1976.

[Bonissone 87] Bonissone, P. Plausible Reasoning. Appears in Encyclopedia of Artificial Intelligence, Volume 1. pp. 854 - 863. Edited by Stuart C. Shapiro. John Wiley & Sons, Inc. 1987.

[Burton 82] Burton, R.R. Diagnosing Bugs in a Simple Procedural Skill. Appears in Intelligent Tutoring Systems, pp. 157 - 183. Edited by D. Sleeman and J. S. Brown. Academic Press. London. 1982.

[Burton and Brown 82] Burton, R. R. and Brown, J. S. An Investigation of Computer Coaching for Informal Learning Activities. Appears in Intelligent Tutoring Systems, pp. 79 - 98. Edited by D. Sleeman and J. S. Brown. Academic Press. London. 1982.

[Carr 77] Carr, B. Wusor-II: a Computer-Aided Instruction Program with Student Modeling Capabilities. AI Lab Memo 417 (Logo Memo 45). Massachusetts Institute of Technology, Cambridge, Massachusetts. 1977.

[Carr and Goldstein 77] Carr, B. and Goldstein, I. P. Overlays: a Theory of Modeling for Computer-aided Instruction. Technical Report AI Lab Memo 406, Massachusetts Institute of Technology. 1977.

[Clancey 87] Clancey, W. J. Knowledge-based Tutoring. The MIT Press. Cambridge, Massachusetts. 1987.

[Cohen 85] Cohen, Paul R. Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach. Pitman. London, England. 1985.

[De Kleer et al 89] De Kleer, J.; Forbus, K.; and McAllester, D. Truth Maintenance Systems. (Notes for tutorial SA5.) Eleventh International Joint Conference on Artificial Intelligence in 1989. Notes available from AAAI. Menlo Park, California. 1989.

[Derry 89] Derry, S.; Hawkes, L. W.; Kegelmann, H.; and Holmes, D. Fuzzy Remedies to Problems in Diagnostic Modeling. Proceedings of the 4th International Conference on AI and Education. pp. 81 - 85. IOS. 1989.

[Fum, Giangrandi, and Tasso 90] Fum, Danilo; Giangrandi, Paolo; Tasso, Carlo. Backward Model Tracing: An Explanation-based Approach for Reconstructing Student Reasoning. Proceedings Eight National Conference on Artificial Intelligence. pp. 426 - 433. 1990.

[Guha and Lenat 90] Guha, R. V.; and Lenat, D. B. Cyc: A Mid-term Report. AI Magazine 11:3, pp. 32 - 59. 1990.

[Huang 90] Huang, Xueming. Student Model Revision: Evolution and Revolution. CSCSI-90. 1990.

[Johnson 86] Johnson, W. L. Intention-based Diagnosis of Novice Programming Errors. Research Notes in Artificial Intelligence. Morgan Kaufmann. Los Altos, California. (copublished with Pitman, London). 1986.

[Kimball 82] Kimball, R. A Self-Improving Tutor for Symbolic Integration. Appears in Intelligent Tutoring Systems. Edited by D. Sleeman and J. S. Brown. Academic Press. London. 1982.

[Langley et al 84] Langley, P; Ohlsson, S; and Sage, S. Machine Learning Approach to Student Modeling. Technical Report CMU-RI-TR-84-7. Robotics Institute, Carnegie-Mellon University. Pittsburgh, Pennsylvania. 1984.

[Murray 90] Murray, William R. A Blackboard-based Dynamic Instructional Planner. Proceedings Eighth National Conference on Artificial Intelligence. pp. 434 - 441. 1990.

[Peachey and McCalla 86] Peachey, D. R. and McCalla, G. I. Using Planning Techniques in Intelligent Tutoring Systems. *International Journal of Man-Machine Studies* 24:77 - 98. 1986.

[Rich 79] Rich, E. User Modeling via Stereotypes. *Cog Sci* 3:355 - 366. 1979.

[Shafer 76] Shafer, G. A Mathematical Theory of Evidence. Princeton University Press. Princeton, New Jersey. 1976.

[Shapiro 83] Shapiro, E. Algorithmic Program Debugging. MIT Press. MIT. 1983.

[Shortliffe and Buchanan 75] Shortliffe, E. H. and Buchanan, B. G. A Model of Inexact Reasoning in Medicine. *Mathematical Bioscience* 23:351 - 379. 1975.

[Sleeman 83] Sleeman, D. Inferring Student Models for Intelligent Computer-Aided Instruction. Machine Learning: An Artificial Intelligence Approach, Volume One. pp 483 - 510.

[Stansfield 76] Stansfield, J. C.; Carr, B.; and Goldstein, I. P. *Wumpus Advisor I: a First Implementation of a Program that Tutors Logical and Probabilitistic Reasoning Skills.* Technical Report AI Lab Memo 381, Massachusetts Institute of Technology, 1976.

[Towne 89] Towne, D. M.; Munro, A.; Pizzini, Q.A., Surmon, D.S., and Wogulis, J. ONR Final Report: Intelligent Maintenance Training Technology. Technical report number 113, Behavioral Technology Laboratories. University of Southern California. 1989.

[Wenger 87] Wenger, Etienne. Artificial Intelligence and Tutoring Systems. 1987. Morgan Kaufmann Publishers, Inc. Los Altos, California.

[Woolf 84] Woolf, Beverly P. Context Dependent Planning in a Machine Tutor. Ph.D. dissertation. University of Massachusetts. 1984.

[Zadeh 78] Zadeh, L. A. Fuzzy Sets as a Basis for a Theory of Possibility. *Fuzzy Sets Sys.* 1:3-28. 1978.