Detecting and Repairing Tutoring Failures

Sarah A. Douglas

CIS-TR-88-09
June 21, 1988

Department of Computer and Information Science
University of Oregon

Detecting and Repairing Tutoring Failures

Sarah A. Douglas
Department of Computer and Information Sciences
University of Oregon
Eugene, Oregon 97403
(503) 686-4408

CSNET: douglas@cs.uoregon.edu

> Any of the systems and contingencies implicated in the production and reception of talk - articulatory, memory, sequential, syntactic, auditory, ambient noise, etc. - can fail. Aspects of the production and analysis of talk that are rule-governed can fail to integrate. In short, the exchange of talk is indigenously and exogenously vulnerable to trouble that can arise at any time.
>
> -- E. Schegloff
> (From "The relevance of repair to syntax-for-conversation."
> In T. Givon (ed), *Syntax and Semantics*, 1979)

1. INTRODUCTION

During the course of studying a number of protocols of human tutors working with human students, I became aware of a complex process of interaction failure and repair. Although much ITS research has been devoted to the understanding and modeling of the detection and repair of student performance failure and misconception in learning curriculum concepts, there is little understanding of an equivalent self-detection and repair issue with tutor performance failure and misconception about what the student is taught. Indeed, there seems to have been a failure to examine the heart of intelligent tutoring systems, what Wenger (1987) terms *knowledge communication*. Communication is an inherent dyadic relation whose primary performance feature is *interaction*. In particular, interaction between humans, as all human performance, appears filled with both slips and bugs. Humans are highly tuned to the detection and repair of these problems. In the remainder of this paper, I present examples of some of the types of tutor interaction failure that I discovered in these protocols, discuss the detection and repair strategies used, and, finally, discuss the implications of these findings for ITS.

My conclusion about tutoring failures is that some types can possibly be reduced by use of an intelligent tutoring system, but that others, called model failures, are an inherent part of the teaching of complex domains. Knowledge and the communication of knowledge are inextricably intertwined. Since we cannot create error-free ITS, we should study in more detail the mechanisms of failure detection and repair that are inherent in human interaction. I believe that achieving this goal will require a much finer grain of analysis of the process of student response during both problem presentation and remediation and will place a greater

emphasis on the detailed design of the interface of ITS.

2. THE DOMAIN STUDIED

Since 1985 Russell Tomlin in the Linguistics Department at the University of Oregon and I have been studying the possible computer simulation of teaching beginning oral communication skills for second (natural) languages.*

Oral communication skills--the ability to comprehend and produce oral discourse--are crucial in nearly every educational, business, and scientific setting of language use. Yet the development of oral communication skills remains a difficult theoretical and practical problem, and traditional language teaching approaches regularly fail to help many learners.

From the point of view of language teaching theory our project draws on two important and innovative approaches to second language learning and teaching: the communicative and the comprehension approaches. Proponents of the communicative approach argue that successful language learning occurs when the student is provided the opportunity to solve non-language problems using the developing second language (Widdowson, 1978; Krashen and Terrell, 1983). They criticize traditional language teaching for focusing too much effort on the conscious discussion and manipulation of rules of language usage and not enough effort on the acquisition of the second language grammar through efforts to use that grammar to solve actual communication problems. This philosophy integrates well with the general spirit of ITS wherein learning is a problem-solving process.

Proponents of the comprehension approach argue that second language learning is enhanced when beginning stages of language learning are devoted to developing the ability to understand the second language. Obligatory oral production is delayed until the student is able to understand easily utterances in the second language. Delaying production improves student performance in other aspects of language acquisition (Asher, 1969; Postovsky, 1977, 1979; Winitz, 1981; Winitz & Reeds, 1973).

Our project embraces both of these complementary approaches to language learning and teaching. The instructional system we have created involves the student in solving communicative problems interactively with the system. The student participates in problem-solving simulations which allow manipulation of objects in a physical scenario or microworld. Information about the problem to be solved as well as information about the microworld is given in the second language. Meta-level commentary by the tutor is also in the second language. The teaching intervention in these simulations can vary from highly directed to coaching to purely student-controlled exploration.

During a typical tutoring session a microworld is used to construct problem solving tasks for the student. Information about the problem to be solved as well as information about the microworld is given in the second language. This approach believes that language learning is context dependent and basically interactive.

---

* This research is funded by FIPSE grant #84.116C from the U.S. Department of Education.

3. PROTOCOL STUDIES

In order to more fully understand the teaching task that we were modelling, we conducted a series of empirical studies of human tutor-student using the communicative/comprehension approach. Our original motivation in studying how human tutors worked with human students was to answer questions which are fundamental to the process of designing an ITS.

1) What does the tutor teach, e.g. the curriculum?

2) What kinds of problems does the tutor generate?

3) How does the tutor diagnose student misconceptions?

4) How does the tutor remediate misconceptions?

5) What is the control strategy of the tutor?

The major difficulty that we faced was that the tutoring approach, namely spontaneous natural language within a loosely structured task, actually discourages the tutor from extensive pre-planning and rote-like teaching and speaking. The goal is for the tutor to produce "natural" language appropriate to the situation. Lacking any texts or syllabi, we were forced to observe the actual teaching situation.

3.1. Procedure

The tutoring situation we studied, which we call Flatland, involved a mainly tutor-directed set of identification and movement tasks intended to teach the linguistic function of referring to objects in a physical context. During a tutoring session the tutor and student sat facing each other across a small table. The tutor had eight small square cardboard tiles, each with a picture of a colored geometric object. In keeping with the spontaneous, situated method of the communicative approach the tutor was given very general goals for the teaching session: Teach the student how to identify these objects by shape (square or circle), color (red or blue), size (large or small), or spatial relation (above, below, left of, right of, between). A typical task might be "Show me, which one is the large blue square?" (identification by deictic task) or "Pick up the small red circle and put it to the left of the large blue circle." (identification by change-of-state task).

Seven videotaped protocols were taken of two different tutors who work at the American English Institute at the University of Oregon. One tutor is highly experienced and is considered one of the foremost world experts in this particular teaching method. The other tutor had had six months of experience using this technique. They were both native English speakers and the language to be taught was English. The more experienced tutor did five protocols, while the novice tutor did two. The students were primarily of either Oriental or Arabic first language origin and were enrolled in a beginning course of study at the Institute. They are essentially nil-proficiency learners and had no prior familiarity with this problem domain.

3.2. Analysis

After collecting the protocols, we were faced with the problem of how to analyze the data in order to answer our major questions. We needed a classification system of constituent types for codifying the events. Lacking any other system we decided to analyze the rhetorical organization of the teaching discourse. There

is quite a bit of precedent for this approach as it attempts to capture the notion of speaker intention through a set of linguistic acts. Sinclair and Coulthard (1972) studied language in the classroom and proposed a hierarchical categorization of discourse units organized by level of abstraction. The highest level is the lesson, followed by transaction, exchange, move and act. The act corresponds most closely to the syntactic clause. Their model proposes twenty-one act types which can be variously combined into five move types according to specified rules. Exchanges are then comprised of moves, also according to specified structures. The model derived is thus a structural context-free form of classroom discourse.

Two applications of the Sinclair-Coulthard model described in the literature, Burton (1981) and Coombs & Alty (1985), take the basic Sinclair- Coulthard model and apply it to other domains. Burton looked at casual conversations and Coombs & Alty looked at conversations between computer center advisors and computer users. Both of these applications alter the original model to account for different observations in the particular domain studied by adding or subtracting types at the levels, but retain the problems of the original Sinclair and Coulthard model.

In addition to the work by Sinclair and Coulthard a similar attempt was made by Grimes (1975) to categorize rhetorical predicates. McKeown (1985) modeled these directly in a program for computer generation of natural language in response to database queries. Neither the categorization by Grimes nor that adopted and slightly modified by McKeown will suffice for the tasks which we are modeling. This is primarily because the classification does not include actions in the world such as a directive to the hearer ("Now take the small blue circle and put it on the right of the large red square."), but are limited to descriptive language. The difficulty is not simply adding more rhetorical predicates, but adding a complementary set of rhetorical actions.

Other computer implementations of discourse generation (Reichman, 1985; Woolf & McDonald,1984) adopt similar categorizations of constituents and use grammars implemented as transition nets to model the rules relating constituents. The work by Woolf and McDonald (1985) is particularly interesting in that it attempts to manage the discourse of an ITS for Pascal programming. The rhetorical organization is hierarchical consisting of three levels: pedagogic, strategic, and tactical. Each level successively refines the actions of the tutor. Later work by Woolf and Murray (1986) uses a transition network of predicates which describe the state of the tutoring discourse, such as "pose problem" or "teach by example", and conversational actions which the tutor can take to change the state of the discourse situation.

All of the above models propose a structure to discourse that is largely independent of the pragmatics of the particular discourse. Thus, the context, the student (hearer) model and the tutor's (speaker's) intentions are informally implied. The models are able to define a set of rules that determines well-formedness, i.e. *whether* a particular sequence of acts will occur, but are unable to explain *why* a particular sequence of rhetorical acts occurs at a particular time in the discourse. This makes them very difficult to apply to other domains where new descriptive categories might occur as well as leaving the analyst unsure how to apply the categorization.

Given these dissatisfactions, the reader should find it not at all surprising that we chose to develop our own approach to analysis. Our major motivation was to link the rhetorical organization of the discourse to the details of the the usual components of an ITS system, its data structures, (tutoring goals, curriculum and

student model), and processes (problem generation, diagnosis and remediation).

We decided on an hierarchical classification with the lesson as the basic context, followed by the exercise, and terminating with the episode. The lesson is organized around the teaching goals and curriculum. Each lesson is broken down into a number of exercises which are the structuring of the curriculum by topic. In the case of our domain, this can be represented by a partially ordered graph in which the first exercise is the identification of object(s) by either color, size, or shape as individual attributes; then the identification of object(s) by composite attributes; and finally the identification/movement of an object by its location relative to another object. This ordering follows the syntactic complexity of direct reference in the utterances: noun ("the square") > noun with adjective modifier ("the red square") > noun with phrase modifier ("the red square to the left of the small blue square"). It also follows the semantic complexity: entity > attribute > 2 entity relation > 3 entity relation.

The third level comprises the episodes of the tutor's discourse. These episodes are grouped into seven general classes. Figure 1 illustrates the classification scheme. The intentions (usually teaching goals) of the tutor, the assumed state of the student's knowledge, the context and focus of previous discourse, the context of the physical environment, and the expected result on the part of the student are used to identify a particular episode type according to a set of prescribed rules.

After all seven protocols were broken down into both exercises and episodes, we did several things. First, for each protocol, tutor and overall, we analysed the number of episode types for each protocol to give us some idea of the frequency and amount of time devoted to each. Second, for each protocol, tutor and overall, we analysed the transitions between episodes. This allowed us to get some sense of overall control.

Finally, we selected a number of episodes for more detailed analysis. The more detailed analysis involved the coding of the verbal actions (lexical as well as phonological aspects) of both tutor and student, non-verbal actions (pauses, gaze, and gestures), and mental states. For example, suppose we wish to explain the following discourse:

Tutor: "Now, take the small red circle"
<pause>
Student: <takes the card and moves it to the side of the table>
Tutor: "and put that below the large red circle."
Student: <hesitates>
Tutor: "below the large red circle."
Student: <pushes the small red circle to below the large red circle>

This segment consists of three episodes denoted by the three separate acts of the tutor with corresponding non-verbal responses by the student. The tutor's first two acts are diagnostic and the third is a remediation which is a repetition of the previous referent phrase. What is immediately apparent is that the episodes must be coded at a level below the sentence and clause, usually at the phrase or even single lexical item level. A second observation is that many of the interaction cues are non-verbal consisting of hesitations, intonations, physical actions, etc. We frequently observed that tutors broke the sentences into diagnostic units such as the above so that the complexity of identifying where the misconception occurs is reduced. Tutors

intently observed all student actions *during their performance* to ascertain if trouble was imminent. The grammar of representation is shown in Figure 2 and the coding of the first utterance from the example above is shown in Figure 3. As can be observed, this coding is quite tedious and was usually done for the episodes selected because they fell into the category of "Repair of Tutor Failure" (Figure 1).

4. TYPES OF TUTORING FAILURE AND REPAIR

Approximately 10-20% of all episodes in these protocols are tutor failures. Many of these failure situations are detected by the tutors and repaired. Tutors spend about 20% of their time repairing their failures. However, there are some failures which are never detected. We counted at least 10 episodes of tutor failure which the tutors failed to detect. The expert and novice tutors made about the same number of failures, but the expert was markedly better at detecting and repairing them.

These failures are not unusual in human behavior. Card, Moran and Newell (1983) in their study of expert human text editing concluded that though error behavior is far from infrequent or inconsequential, in experts the detection and correction of errors is mostly routine. They observed errors in 36% of the tasks under study and found that errors doubled the time to perform the tasks in which they occurred.

The types of failure that we compiled fall nicely into two categories, slips and bugs. The term *slip* comes from the compendium of data on verbal "slips of the tongue" phenomena, but was extended by Norman (1981) to include non-linguistic failures. A slip is defined by Norman (1981, p. 1) as "... a form of human error defined to be the performance of an action that was not what was intended." *Bugs* are the remaining failures which result from failure in formation of intention. Inappropriate goal determination, faulty knowledge, and inability to recognize context shift fall into this category (Brown & Burton, 1978).

4.1. Slips

Slips are performance as opposed to planning failures. For example, tutors sometimes cannot remember what they just said. This is often a particular word or concept: "...and put it between....did I say 'between'?..." Given the apparent simplicity of the language generation, why do slips occur in these protocols? A major cause is the difficulty of the task. Language tutors have limited time for planning and revision. They must concurrently attend to the response of the student while creating the next tutoring action. At any moment, depending on the behavior of the student, they may have to alter their plan mid-course. We discovered many monitoring strategies that tutors use to detect quickly trouble in the student's understanding of what has been said. Thus the complex interleaving of plan formation, plan execution, and online monitoring create frequent slips.

Another cause of slips is the conflict in tutors between their normal use of the language and the restricted subset they must use in a lesson. It is very difficult to restrict performance of routinized behavior. It requires a great deal of consistency in word choice which in fluent discourse would rarely be demanded. For example, tutors frequently taught "above" as a relation and then later used "over". Either word choice, used alone, would have been reasonable, but when the words were used together, the students became confused and the tutors had a hard time understanding this.

Language also has mechanisms which establish a context of referential objects; these objects may have priority over the immediate utterance. Thus we observed a tutor establish a focus of attention on an object and then refer to it with an obviously incorrect noun phrase. Since the student responded to what had to be meant (!) rather than what was said, the tutor never noticed the discrepancy between what was said and what the student did.

4.2. Bugs

For our tutors bugs constitute failures to diagnose properly a student's knowledge, selection of inappropriate teaching strategy, misjudgement about the difficulty of the curriculum and its sequencing and often a failure to coordinate what is said (as an intention of what is to be taught) with what is in the context.

Examples of these tutoring bugs abound. Tutors constructed tasks and failed to notice that the context was inappropriate until they had already uttered it. For example, one tutor said "Point to the square." when there were two on the table and "Take the circle..." when there was no circle. The novice tutor didn't anticipate the complexity of diagnosing a student misconception in a simple movement task and presented it before confirming the student's knowledge of the component elements. When unable to teach a concept, both tutors prematurely judged a concept learned when it wasn't. Often tutors ignored the needs of an exhausted student and pressed on through their planned curriculum.

Although these are difficult failures one of the most pernicious forms of failure is model failure where assumptions about states of knowledge are wrong.

For example, imagine a situation where there are congruent states of knowledge:

T= Tutor; S= Student

(1) T believes that p;
(2a) T believes that S believes that p;
(3a) S believes that p;
4a) S believes that T believes that p.

However, it might be the case that the states of knowledge are incongruent:

(3b replaces 3a) S believes that q;
(4b replaces 4a) S believes that T believes that q.

It is the tutor's task to detect and repair or bring about a congruent state of knowledge again:

(2b replaces 2a) T believes that S believes that q;
(5 added) T believes that S believes that T believes that q.

It is sometimes the case that both p and q can be right, caused by the ambiguity of the situation or the language. For example, tutors often did not anticipate the ambiguity of point of view in spatial relations, i.e. "on the left" as *your* left versus *my* left. In normal conversations this occurs regularly. Both people can be talking about two different things and yet appear to be talking about the same thing. How then can the tutor diagnose this failure? I will return to this topic again when I discuss the problem of detection and

repair of failure in detail.

4.3. An Extended Example

In order to more fully understand some of the issues in tutor failure and repair, an extensive example, shown in Figure 4a-d, will be used. The detailed coding has been omitted for sake of readability. This example is taken from the expert tutor working with a Japanese learner. (As is often the case with this teaching approach the tutor has no expertise in the student's native language.) The tutor and student had been working together for approximately 1.5 minutes. The tutor has decided that the student can discriminate deictically (e.g. by pointing to) the individual tiles by color, shape and size and a combination of features in response to requests like "Show me the small blue square." She now wants to test knowledge of spatial relations by a movement task.

This protocol segment has been divided into four exercises. In the first exercise the tutor first tests the student to verify that the referent small blue square is known (lines 1-5). Then the student performs a very confusing (to the tutor) series of actions in response to the tutor's request to put one object "under" (lines 6-17) another. All of these actions occur very quickly and the tutor comes to the conclusion that the student might understand the word "below" better than "under." This motivates Exercise #2 in which the tutor tries the same test with the word "below". The student still misunderstands. In Exercise #3, the tutor decides to demonstrate the spatial relation with the word "below" by analogy with "left" using the same two objects as in Exercise #1. In Exercise #4 she tests the student with the relation using "below" and the same two objects. The student mimics her behavior exactly. The tutor then decides that the problem is resolved and the student understands. The remainder of the protocol (not shown in Figure 4) confirms this understanding.

This sequence is representative of the complexity of tutoring failure that we observed. There is one slip where the tutor self-corrects "move" to "take" (Exercise #1 line 6) without any apparent feedback from the student. But most failures are bugs. For example, an ambiguous use of "blue square" (Exercise #1 line 9) causes the student to confuse the two blue squares. The tutor had assumed that the small blue square was in discourse focus. However, the major tutor bug is much more complicated and is a type of model failure. Rather than assuming a two dimensional plane surface for the spatial relation "under" as the tutor expects, the student interprets it as three dimensional, i.e. "underneath" (Exercise #1 line 14-17). This masks another tutor bug. The tutor fails to notice that the student reverses the two referents (Exercise #1 line 11 and 16). The reversal of the two referents would be possible in Japanese where the subject/object role in a relation are marked by particles. English requires word order to specify these roles.

In Exercise #2 these two issues are still confused. The tutor uses the word "below" but the student still interprets it three-dimensionally. She gives negative feedback to the student about his action (line 6) but the student interprets it as the need to reverse the two referents (line 7). The tutor still fails to see that the student has reversed the relation between the two referents. After tutoring by demonstration in Exercise #3 and testing in Exercise #4, the student serendipitously not only learns that the words "under" and "below" are used two-dimensionally in this context but that English syntax (word order) specifies which objects are marked in the relation.

5. DETECTION OF FAILURE

In the extended example above, we are struck by the microscopic level of detail that the tutor uses to interpret her own success or failure in communicating with the student. Thus one strategy aiding detection is that the tutor repeatedly breaks the presentation into small units so that detection of failure is simplified. It is important to stress that the form of the communication may be just as crucial as the content. In particular, we observe the tutor monitoring many non-verbal actions of the student while the tutor speaks: hesitations, gaze, and gesture. Some research on communication (Mehrabian, 1972) has suggested that in judging deception, a case where model congruence is intentionally pretended, vocal information contributes 38% of the information and facial expression 55%. Verbal information is only 7%. These data suggest that human interaction is highly attuned to the observation of the process of communication, particularly of non-verbal information.

How then does the tutor detect her own failure? There are two cases: One in which there is self-detection and the other where some cue is given by the student. Self-detection can occur in at least two ways. The first is that the tutor imagines the act of hearing her own utterances. This can account for anticipating ambiguities on the part of the hearer. The second way is that the tutor can review her own memory of what she has just said and debug it.

The tutors often detected that trouble had occurred from cues given by the student. Non-verbal cues consisted of confused facial expressions, hesitation in performance, and hesitation in turn-taking. Verbal cues often consisted of the repetition of a confusing word or expression with a question intonation. In these tutoring protocols we observed most of the detection strategies documented by Clark and Wilkes-Gibbs (1986) in their study of referring.

All language interaction is essentially collaborative. Not only is the tutor's detection of trouble important but there is an equivalent detection by the student of the tutor's failure. This is apparent from the speaker's false- starts and interruptions, use of meta-language, intonation, gestures, and nervous laughter.

We also noticed that occasionally a tutor would fail to detect her own failure, for example, that the student's response was incorrect. Model failures are particularly difficult to detect because of inherent ambiguities in communication. A probable explanation is that the tutor expected a correct response and was already planning the next task. This failure to trigger an otherwise active schema indicates the expectation-driven nature of diagnosis and the demand on mental resources for planning.

6. REPAIR STRATEGIES

There is a preference in interaction for self-repair by the speaker or whoever is the active agent until the turn is passed (Schegloff et al., 1977). In the protocols we examined, turn-taking definitely existed although the student was not a "speaker" in the normal sense of the word. This notion of turn-taking is fundamental to all interaction and its control is very complex.

The tutor upon detecting trouble had three major repair strategies. Usually, the tutor would repeat the request or other utterance modifying only the corrected part. That is, the tutor didn't want to shift the context in any way. Similarly, after a failure was detected and a repair made, the next activity would often

maintain the focus of attention of the student on the same objects. This type of repair we call *repair reinforcement* and we will discuss it in more detail below. And finally, the tutor sometimes was forced to abandon the repair and would usually make a major context shift after that while retesting knowledge previously believed known. Students often aided the tutor in the repair. They would sometimes fill-in or complete the task according to their assumptions of what they thought the tutor intended.

The most common repair strategy of tutors for slips was to repeat the task with the corrected utterance, but only if it were appropriate within the context. Students had no difficulty comprehending that there had been a problem, probably given by the tutor's verbal intonations and the mismatch of the original utterance to what was context. Often we saw tutors take advantage of their failure to teach or diagnose a concept out of the planned sequence. The partial order of the curriculum allows an opportunism that creates local curriculum sequence differences among multiple protocols of the same tutor. Tutors often ignored slips they had made. We do not know if this was intentional or not.

Repair of bugs was much more difficult. Tutors often lapsed into meta-level discourse which was incomprehensible to the student, but which signaled that something was wrong. A particularly difficult repair for one tutor was the left-right concept. Halfway through the teaching episode, the tutor realized that the examples had all been reversed. (Since the tutor sat across from the student, she had inadvertantly taught left as *her* left.) The tutor attempted a repair by getting up and sitting next to the student and then repeating the tasks. We observed several episodes in which tutors were forced to abandon a task altogether because of increasing student confusion. Later they returned to it successfully. Tutor repair of failures in student misconception diagnosis involved careful monitoring of the next tasks and reverting to testing earlier, easier tasks. This indicates that tutors were checking their own verification techniques. We noticed that the complete failure of an initial diagnostic strategy in the first protocol of the novice tutor resulted in a radical rearrangement of the tutoring activities to resemble more closely that of the expert tutor.

As mentioned above, maintenance of the context of interaction and focus of attention is crucial since when failure occurs it is often impossible to know quite what is wrong. From two protocols, where 28 tutor failures were detected and repaired by the tutor, 15 had a continuity which we call repair reinforcement. Repair reinforcement is when an failure occurs, the repaired referent is used not only in the current problem generation but in the following as well. Figure 5 illustrates an example of repair reinforcement. In this example, the repaired referent large red square is used in the next exercise.

Repair reinforcement occurs usually after the first 10 mins. Repair reinforcement serves to maintain focus of attention on that referent and verify that student's performance is not confused by failure repair discourse. It is an example of the type of opportunism that failure can create in tutoring.

7. COMPUTATIONAL IMPLICATIONS FOR INTELLIGENT TUTORS

There are some interesting control issues which spring from the problem of failure and repair that we observed in these protocols. Classical ITS control modeling is very top-down, contolled by the structure of the curriculum and the tutoring task cycle, i.e. problem generation, diagnosis, and remediation. What we observed in these protocols is much more bottom-up, data-driven, or opportunistic control determined by student modelling issues and trouble repair. Curricula must be structured as partial orders to allow this

flexibility. Repair, including repair reinforcement, tended to require control which maintained a focus of attention within the tutoring context. When trouble occurs in communication, it is always the case that the repair action ruptures the normal ongoing activity and establishing another context. Resuming the on-going tutoring is often a delicate and difficult task. We believe that there are many perceptually observable activities which tutors use to create context shifts. We need to understand these better.

A second interesting issue we observed with computational implications is the balancing of planning versus execution versus monitoring. Tutors were planning under real-time constraints, and achieving multiple goals. Slips and bugs are no doubt created because of these demands (Birnbaum, 1986). In general, we saw very opportunistic styles of planning. The kind of opportunism observed in repair involves the ability to suppress currently active goals and recognize goals which are not currently active but could be achieved. This type of computation has not generally been applied to intelligent tutoring systems. It requires much more flexible systems with fast pattern matching.

It is obvious from our protocols that meta knowledge, knowing what you do and know is crucial for repair. Some sort of tutoring context maintenance must be done and used for evaluation. Strategies for detection and repair of failure have to be thought through. My general feeling is that they are domain independent.

Finally, if human-human interaction is any measure then we need to pay attention to more micro-level feedback from the interface, for example, the duration of pauses, the path of mouse selections, or the short verbal feedback signifying the questioning of a referent. Our systems are without visual input which often carries crucial information for face-to-face interaction. One question we can ask is, "How do human tutors adapt to situations without visual input?" Protocol studies we are doing now using the same tutoring task may give us some answers to this question which we can possibly model. In the longer range, we are developing theoretical models of interactive communication and, thus, interactive teaching.

8. CONCLUSIONS

Are these tutoring failures peculiar to just this particular domain of natural language discourse? If anything the tutoring situation modeled here, very closely approximates the type of human-computer interaction that is common today with verbal instruction from the tutor and pointing and movement requested of the student. Rather, I observe the observations made in this paper are common to any interaction between knowledge-based systems, whether human or machine. All such interactions occur within a context where reference must be established and maintained, and attention allocation becomes important. It is impossible for any programmer to anticipate all future situations between the program and a human, and to even guarantee that the program will perform without failure. It is also the case that because tutoring must occur in real-time it may be computationally impossible to derive an exact solution in many tutoring cases. Thus, the machine tutor, like the human may be forced to adopt approximate judgments which are more susceptible to bugs and in need of repair strategies.

Although the position I am taking here may appear radical, it is founded on the hard lessons of our ITS experience. We know that so-called errors cannot be eliminated from an interface. Thus we must accommodate the problem in the initial design and build systems which are robust in repair. But where do

we learn how to do such things? More intensive study of human conversation could prove very useful. Some of the collaborative processes that speakers and hearers use to detect and repair misunderstandings are just being studied (Clark & Wilkes-Gibbes, 1986; Cohen, 1985; Suchman, 1987). These concepts, which are basic to human interaction and not just natural language, may help us better model the complex interaction process between tutor and student. In short, the interface design may be crucial the critical Achilles heel building effective intelligent tutoring systems.

9. ACKNOWLEDGEMENTS

This paper has resulted from many painstaking hours of videotape analysis and computational modeling. Three graduate students in the Computer Science Department, David Novick, Sue Olivier, and Sharon Collins contributed enormously to that task. To them I am indebted for many hours of discussion and insights of interpretation.

10. REFERENCES

Asher, J. J. (1969). The total physical response approach to second language learning. *The Modern Language Journal, 53*, pp. 3-17.

Birnbaum, L. (1986). *Integrated Processing in Planning and Understanding.* Unpublished doctoral dissertation, Yale University, Department of Computer Science.

Burton, M. (1981). Analysing spoken discourse. In Coulthard, M. & Montgomery, M. (Eds.) *Studies in Discourse Analysis.* London: Routledge and Kegan Paul.

Brown, J.S. and Burton, R. (1978). Diagnostic models for procedural bugs in basic mathematical skills. *Cognitive Science, 2*, 155-192.

Card, S., Moran, T.P. and Newell, A. (1983). *The Psychology of Human- Computer Interaction.* Hillsdale, N.J.: Erlbaum.

Clark, H. and Wilkes-Gibbes, D. (1986). Referring as a collaborative process. *Cognition, 22*, 1-39.

Cohen, P.R. (1985). The pragmatics of referring and the modality of communication. *Computational Linguistics, 10*, 97-146.

Coombs, M.J. and Alty, J.L. (1985). An Application of the Birmingham Discourse Analysis System to the Study of Computer Guidance Interactions. *Human-Computer Interaction*, Vol. 1, 243-282.

Grimes, J.E. (1975) *The Thread of Discourse.* The Hague: Mouton.

Krashen, S. and Terrell, T. (1983). *The Natural Approach.* San Francisco: Alemany Press.

McKeown, K.R. (1985). *Text generation: Using discourse strategies and focus constraints to generate natural language text.* Cambridge: Cambridge University Press.

Mehrabian, A. (1972). *Nonverbal Communication.* Chicago: Aldine- Atherton.

Norman, D.A. (1981). Categorization of action slips. *Psychological Review, 88*, 1, 1-15.

Postovsky, V. 1977. Why not start speaking later? In M.Burt et al. (Eds), *Viewpoints on English as a Second Language.* New York: Regents.

Postovsky, V. 1979. Effects of delay in oral practice at the beginning of second language learning. *Modern Language Journal*, 58:229-239.

Reichman (1985). *Getting Computers to Talk Like you and Me.* Cambridge, MA: MIT Press.

Schegloff, E.A., Jefferson, G. and Sacks, H. (1977). The Preference for Self- Correction in the Organization of Repair in Conversation. *Language*, Vol. 53, No. 2.

Sinclair, J. McH., and Coulthard, R.M. (1972). *Toward an Analysis of Discourse.* London: Oxford University Press.

Suchman, L.A. (1987). *Plans and Situated Actions: The Problem of Human-Machine Communication.* Cambridge: Cambridge University Press.

Wenger, E. (1987). */Artificial Intelligence and Tutoring Systems.* Los Altos, CA: Morgan Kaufman.

Widdowson, H.G. (1978). *Teaching Language as Communication.* Oxford: Oxford University Press.

Winitz, H. (Ed). (1981). *The Comprehension Approach to Foreign Language Instruction.* Rowley, Massachusetts: Newbury House.

Winitz,H. and Reeds, J. (1973). Rapid acquisition of a foreign language by the avoidance of speaking. *International Review of Applied Linguistics*, 11, 295-317.

Woolf, B. and McDonald, D.D. (1984). Building a Computer Tutor: Design Issues. *IEEE Computer*, September 1984.

Woolf, B. and Murray, T. (1987). Discourse Transition Networks for Intelligent Tutoring Systems. *Technical Report*, Computer and Information Science Dept., University of Massachusetts.

Figure 1: Protocol analysis classification

- Lesson
  - Exercise (by curricular topic)
    - Episode
      - Introduction
        - Introduce topic
      - Closure
        - Close topic
      - Testing/Diagnosis
        - Problem generation
        - Affirmation demonstration
      - Demonstration
      - Remediation
        - Descriptive referral
        - Repetition
        - Remediation demonstration
        - Opportunistic remediation
      - Repair of Tutor Failure
        - Disambiguation
        - Self-correction
        - Repair-reinforcement
      - Reinforcement
        - Acknowledge
      - Reassurance
        - Reassure

Figure 2: Coding grammar

<illocutionary-act> ::= <request-act> | <assert-act> |
<change-of-state-act> | <deictic-ref>

<request-act> ::= request(Actor,{<illocutionary-act>}+)

<assert-act> ::= assert(Actor,<value>({<illocutionary-act>|
<state>}+))

<change-of-state-act> ::= know(Actor,{<illocutionary-act>|
<state>}+) |
take(Actor,{<state>}+) |
put(Actor,{<state>}+) |
has(Actor,{<state>}+)

<deictic-ref> ::= physically-refer-to(Actor,{<state>}+)

<state> ::= prop(<referent>,property+) |
spatial-rel(<relation>,{<referent>}+)

<referent> ::= name(Obj) | descriptor(Obj) | deictic-ref(Obj) |
anaphora(Obj)

<value> ::= True | False | Ambiguous | Unknown

<relation> ::= Above | Below | Right-of | Left-of | Between | Under

<meta-act> ::= demarcate-focus-boundary(Actor) |
reassure(Actor,Hearer)

In the grammar variables are denoted ?name, for example ?truth-value is used for <value>, and constants are denoted Name. Tile configurations are from the student's perspective. Other factors that influence classification of episodes are the student model, the curriculum, and the context. The student model includes information pertaining to the tutor's understanding of the student's level of comprehension within the context of the lesson. Thus history is absent from this model, as well as all information that the tutor assumes the student knows. The curriculum states how far into the curriculum the lesson has progressed. Context gives the tile configuration on the table. This factor circumscribes, to a certain extent the moves available to the tutor.

The student model can be viewed as the tutor learning about the student. The tutor can therefore learn by example (the student performs a task either correctly or incorrectly), by inference (the student fails to perform the task), and by inferred analogy (the student demonstrates knowledge of A therefore infer they have knowledge of B) Fairly simple rules can be associated with building up this student model, for example:

R1: (S requested to perform task) & (S performs task correctly) ->
(S knows (action AND referents))

R2: (S requested to perform task) & (S performs task incorrectly) ->
(S ~know aspect in which S performed incorrectly)

R3: (S identifies A) & (A orthogonal to B) -> (S knows (A & B))

R4: (S requested to perform task) & (S ~performs task) ->
(S ~know (action OR referents))

Figure 3: Coded example

Utterance: "Now, take the small red circle" <pause>

Context: SBC SRC
LBC LRC LRS SBS SRS
LBS

(In focus: SBC, LRC from the previous utterance)

note: C is Circle, S is Square, L is Large, S is Small, B is Blue, R is Red

Curriculum: Teach(spatial-rel(Below))
Goal: know(T,know(S,spatial-rel(Below,descr(SRC),descr(LRC))))
OR know(T,~know(S,spatial-rel(Below,descr(SRC),descr(LRC))))
Subgoal pursued: know(T,know(S,prop(descr(SRC), Small Red Circle)))
OR know(T,~know(S,prop(descr(SRC), Small Red Circle)))

Tutor's Illocutionary Act: request(T,take(S,prop(descr(SRC), Small Red Circle)))

Student Model: know(T,know(S,prop(descr(SRC), Small Red Circle)))
(assumed for this example having been confirmed earlier)

Tutor's Intended (Perlocutionary) Effect: take(S,prop(descr(SRC), Small Red Circle))

Episode Class -> DIAGNOSIS (Student Model & Curric & Goals)
Type -> PROBLEM GENERATION (Intended Perlocutionary Effect)

Apply rule:
R1: (S requested to perform task) & (S performs task correctly)
-> (S knows (action AND referents))

Thus Student Model: know(T,know(S,prop(descr(SRC), Small Red Circle)))
AND know(T,know(S,take(S,<state>)))

Figure 4a

Exercise #1

Note: C is Circle, S is Square, L is Large, S is Small, B is Blue, R is Red

01:34 [Tutor places all the tiles in a vertical arrangement at the left side of the table with the SBS placed in the center of the table.]

```
?
?
?
LBS
LRC                SBS
?
?
```

1) TUTOR: "Show me the SBS."
2) STUDENT: [quizzical look]
3) TUTOR: "Show me the SBS."
4) STUDENT: [points to the SBS]
5) TUTOR: "Okay."
6) TUTOR: "Move ... take the LRC."
7) STUDENT: [takes LRC in hand]
8) TUTOR: "Uh-huh."
9) TUTOR: "and put it under the BS." [tutor has SBS in focus and uses anaphoric]
10) STUDENT: [leaves LRC below the SBS]

```
?
?
?
LBS
                   SBS
?
?                  LRC
```

11) STUDENT: [moves SBS to previous LRC slot (which is below the LBS)]

```
?
?
?
LBS
SBS
?
?                  LRC
```

12) TUTOR: "UhhHmm..." [ambiguous as to whether positive or negative feedback]
13) STUDENT: "Under....under" [touches LBS then SBS]
14) STUDENT: [slides SBS to the center of the table]
15) TUTOR: "Wait, wait one minute."
16) STUDENT: [physically places LRC on top of SBS. looks at her and laughs.]
17) TUTOR: "Ah-hah, that's very nice." [Laughs, gestures]
18) STUDENT: [removes LRC and moves SBS back to slot below LBS]

19) TUTOR: "Okay good, okay."

20) TUTOR: "Let's start again." [Exchanges SBS with LRC.]

Figure 4b

Exercise #2

2:25

1) TUTOR: "Let's start with this one, the SBS." [points to SBS]

2) TUTOR: "Now, take the LR, take the LRC,"

3) STUDENT: [takes LRC]

4) TUTOR: "and put it below, put it below the BS, below the BS."

5) STUDENT: [picks up SBS places it on top of LRC]

6) TUTOR: "Okay, okay don't worry, don't worry."

7) STUDENT: [exchanges SBS and LRC.]

8) TUTOR: "Almost, almost." [Laughs]

9) STUDENT: [puts LRC back in slot.]

Figure 4c

Exercise #3

1) TUTOR: "Let's try again." [moves the tile back.]

2) TUTOR: "Here we've got the, okay, SBS."

3) TUTOR: "Now, take the LRC."

4) TUTOR: "Put the LRC, the LRC - _below - the LRC _below the SBS."

5) STUDENT: [points to LRC, then the SBS. picks out LRC and puts it to the left of the SBS.]

6) TUTOR: "Oh....kay, okay. Now this, this is, this is on the left, that's on your left, that one's on the left."

7) TUTOR: "So below" [picks up LRC] "and yeah, under, below." [puts it below SBS]

8) TUTOR: "Okay." [places LRC back in the slot.]

Figure 4d

Exercise #4

03:53 [All tiles are placed at the side of the table again with SBS in center.]

1) TUTOR: "First, take the SBS."

2) STUDENT: [takes SBS in hand]

3) TUTOR: "Okay."

4) TUTOR: "Now, take the LRC and put it below the SBS."

5) STUDENT: [takes LRC in hand and places it below the SBS]

6) TUTOR: "Good, okay, okay."

Figure 5: Repair Reinforcement Example

Exercise #1
TUTOR: "Now, move the small, no, the _large red square"
STUDENT: [picks up LRS]
TUTOR: "and put that between the small blue and small red circles."
STUDENT: [puts LRS between SBC and SRC]

Exercise #2
TUTOR: "Large blue circle above large red square."
STUDENT: [puts LBC above LRS]