May 1982
Also numbered: HPP-82-8

Report. No. STAN-G-82-910

Exploration of Teaching and Problem-Solving Strategies, 1979–1982

by

William J. Clancey & Bruce Buchanan

Department of Computer Science

Stanford University
Stanford, CA 94305

EXPLORATION OF TEACHING AND PROBLEM-SOLVING STRATEGIES, 1979-1982

William J. Clancey
Bruce G. Buchanan

Department of Computer Science
Stanford University, Stanford CA 94305

Contract No. N000C14-79-0302, effective March 15, 1979.
Expiration Date: March 14, 1982
Total Amount of Contract -- $396,325
Principal Investigator, Bruce G. Buchanan (415) 497-0935
Associate Investigator, William J. Clancey (415) 497-1997

Sponsored by:
Office of Naval Research,
Personnel and Training Research Programs,
Psychological Sciences Division.
Contract Authority No. NR 154-436
Scientific Officers: Dr. Marshall Farr and Dr. Henry Halff

1. Introduction

This is the final report for Contract N-00014-79-C-0302, covering the period of 15 March 1979 through 14 March 1982. The goal of the project was to develop methods for representing teaching and problem-solving knowledge in computer-based tutorial systems. One focus of the work was formulation of principles for managing a case method tutorial dialogue; the other major focus was investigation of the use of a production rule representation for the subject material of tutorial program. The main theme pursued by this research is that representing teaching and problem-solving knowledge separately and explicitly enhances the ability to build, modify and test complex tutorial programs.

Two major computer programs were constructed. One was the tutorial program, GUIDON, which uses a set of explicit "discourse procedures" for carrying on a case method dialogue with a student. GUIDON uses the original MYCIN knowledge base as subject material, and as such, was an experiment in exploring the ways in which production rules can be used in tutoring. GUIDON's teaching knowledge is separate from and compatible with any knowledge base that is encoded in MYCIN's rule language. Demonstrations of GUIDON were given for two medical and one engineering application. Thus, the generality of this kind of system goes beyond being able to teach about any problem in a "case library"--it also allows teaching expertise to be transferred and tested in multiple problem domains.

The second major program is the consultation program, NEOMYCIN. This is a second generation system in which MYCIN's knowledge has been reconfigured to make explicit distinctions that are important for teaching. Unlike MYCIN, the program uses the hypothesis-oriented approach and predominantly forward-directed reasoning. As such, NEOMYCIN is consistent with and extends psychological models of diagnostic problem-solving. The program differs from other knowledge-based AI systems in that reasoning is completely controlled by a set of explicit meta-rules. These meta-rules are domain independent and constitute the diagnostic procedure to be taught to students: the tasks of diagnosis and heuristics for attending to and confirming relevant diagnostic hypotheses.

Separation of problem knowledge and diagnostic strategy in this way allows a tutorial program to present them separately to a student, as well as to look for them in his behavior.

Constructing these programs for representing teaching and problem-solving knowledge has provided us with new methods for building complex tutorial programs and advanced our theoretical understanding of the process of diagnostic reasoning.

Theoretical work focused on two aspects of diagnosis. First, diagnostic knowledge can be characterized on multiple levels corresponding to heuristics that link data to hypothesis, structural relations for indexing heuristics, strategies for attending to appropriate structural relations, and theoretical knowledge for justifying (and remembering) heuristics. Second, the *meta-strategy* of diagnosis can be characterized in terms of constructing, extending, and testing a set of hypotheses with respect to a hierarchical problem space of previously learned diagnoses.

At the conclusion of this contract, NEOMYCIN was serving as the core for many projects related to teaching diagnosis. A prototype student modelling program called IMAGE has been implemented; it will be the basis for offering strategical help to students in the new GUIDON2 tutorial program. An explanation system for NEOMYCIN is under development; it will be a testbed for producing explanations on multiple levels of detail that will be part of GUIDON2. A small project was investigating how NEOMYCIN's diagnostic strategy would apply to computer failure problems. Another project was investigating the advantages of representing NEOMYCIN in a predicate calculus language like MRS, to enhance the program's knowledge about itself. Finally, a project had just begun in interactive knowledge acquisition that takes advantage of NEOMYCIN as a psychological model.

2. Background: Expert Systems and Tutoring

Recent advances in our understanding of human problem solving will soon make it possible to "rewrite the textbooks" in diverse fields of scientific and medical problem solving. The mounting evidence from Cognitive Science studies of the past 5 years--schema-based understanding [Rumelhart 80], opportunistic planning [Hayes-Roth 80], knowledge compilation [Anderson 80], failure-driven memory [Schank 81], hypothesis and revise plan recognition [Schmidt 80], to name a few examples--promises a new kind of instruction that will be based on theories of how people organize knowledge and manage the task of problem solving. In short, psychological studies have found that computational metaphors--the theoretical concepts of Artificial Intelligence in particular--are providing a new language for understanding how experts think, and a resulting new understanding of what students need to be taught.

The idea of directly teaching students "how to think" goes back at least to Polya [Polya 57], but it reached a new stage of development in Papert's laboratory [Papert 70]. In the LOGO lab, young students were taught AI concepts such as hierarchical decomposition, opening up a new dimension by which they could take apart a problem and reason about its solution. In part, Polya's heuristics have seemed vague and too general, too hard to follow in real problems. But progress in AI programming, particularly Expert System' design, has suggested a vocabulary of *structural concepts* that we now see must be conveyed along with the heuristics to make them intelligible [Clancey 81 b].

Developing in parallel with Papert's educational experiments and capitalizing even more directly on AI technology, a series of complex programs called Intelligent Tutoring Systems were constructed in the 1970's. In contrast with the CAI programs of the 1960's, these programs used new AI formalisms to separate out what they intended to teach from their program or logic for interacting with the student. This approach has several advantages: it becomes possible to keep records of what the student knows; the logic of teaching can generalized and applied to multiple problems in multiple

[1] **A glossary appears at the end of [Clancey 81 a]**

problem domains; and a model of student knowledge can be inferred from student behavior and used as a basis for tutoring. The well-known milestones in ITS research include:

- interacting with the student in a mixed-initiative dialogue [Carbonell 70]
- tutoring by the Socratic method [Collins 76]
- evaluating student hypotheses for consistency with measurements taken [Brown 75]
- enumerating bugs in causal reasoning [Stevens 78]
- interpreting student behavior in terms of expert knowledge ("overlay model") [Burton 79, Carr 77, Clancey 79a]
- codifying discourse procedures for teaching [Clancey 79b]
- constructing incorrect plans or procedures [Genesereth 81, Brown 78]
- relating incorrect procedures to a generative theory [Brown 80].

The record of ITS research reveals a few recurring questions:

1. ***Nature of Expertise:*** What is the knowledge we want to teach a student?
2. ***Modelling***: How can we determine what the student knows?
3. ***Tutoring:*** How can we improve the student's performance?

Almost invariably, researchers have backed off from initially focusing on the last question--"How shall we teach?" --to reconsider the second question, building a model of the student's knowledge. This follows from the assumption that student errors are not random, but reflect misconceptions about the procedure to be followed or facts in the problem domain, and the best teaching strategy is to directly address the student's misconceptions.

In order to extend the research in building models of misconceptions in well-understood domains such as subtraction to more complex domains such as physics, medicine and electronic troubleshooting, we need a sounder understanding of the nature of knowledge and expertise. Comparison studies of experts and novices [Chi 80, Feltovich 80, Lesgold 81] are revealing that how the expert structures a problem, the very concepts he uses for thinking about the problem, distinguish his reasoning from the student's often formal, bottom-up approach. These studies suggest that we might directly convey to the student the kinds of quick associations, patterns and reasoning strategies that experts build up tediously over long exposure to many kinds of problems--the kind of knowledge that tends not be to written down in basic text books.

It is with this premise--that we will be better teachers by better understanding expertise--that Expert Systems research becomes of keen interest to the educator. These knowledge-based *programs* contain within them a large amount of facts and rule-like associations for solving problems in restricted domains of medicine, science and engineering. While these programs were developed originally just for the sake of building systems that could solve difficult problems, they have special interest to Cognitive Science research as simulation models that can be used as a "laboratory workbench" for experimenting with knowledge structures and control strategies. By altering the "program as a model," one can test hypotheses about human performance (for example, see [Johnson 81]).

Another natural application for Expert Systems in education is to use them as the "knowledge foundation" for an Intelligent Tutoring System. Brown pioneered this technology in the SOPHIE3 system which took a student through the paces of debugging a circuit. Brown, Collins and Goldstein pioneered the use of production rules to express knowledge about how to interact with a student and how to interpret his behavior. The first tutor built on top of a complex expert system was GUIDON [Clancey 79c], using MYCIN's 400 production rules and tables for teaching medical diagnosis by the case method. GUIDON's teaching expertise is represented cleanly and independently of the domain rules; it has been demonstrated for both medical and engineering domains.*

Complementing the studies in expert/novice differences as well as our own work at Stanford on systems that explain their reasoning, we have recently shown that Expert Systems must represent knowledge in a special way if it is to be used for teaching [Clancey 81 b]. First, *the program must*

---

[2] **In GUIDON teaching knowledge is treated as a form of expertise. That is, GUIDON has a knowledge base of teaching rules that is distinct from MYCIN's knowledge base of infectious disease rules.**

convey organizations and approaches that are useful to the *student;* this argues for a knowledge base that reflects ways of thinking used by people (the hypothesis formation approach). Second, *various kinds of knowledge must be separated out and made explicit so reasoning steps can be carefully articulated--the* expert's associations must be decomposed into structural and strategic components. Under our current contract with ONR, such an expert system, called NEOMYCIN, has been constructed [Clancey 81c]. It is being readied for use with students both through active development of its knowledge base and construction of modelling programs that will use it as a basis for interpreting student behavior.

The ultimate goal of our work in the past few years has been to use NEOMYCIN for directly teaching diagnostic problem solving to students. Students will have the usual classroom background, but will be exposed in this tutoring system to a way of thinking about and organizing their textbook knowledge that is usually taught only informally in apprenticeship settings. That is, we are capturing in an Expert System what we deem to be the essential knowledge that separates the expert from the novice, and teaching it to the novice in practice sessions in which its value for getting a handle on difficult, confusing problems will be readily apparent. Empirical studies will be a key part of this research.

We view our work as the logical "next step" in knowledge-based tutoring. Just as representing expert knowledge in a simulation program provides a vehicle for testing hypotheses about how people reason, using this knowledge in a tutoring system will enable us to see how the knowledge might be explained and recognized in student behavior. The experience with GUIDON1, as detailed further in the next section, illustrates how the tutoring framework provides a "forcing function" that requires us to clarify what we want to teach and how we want to teach it.

Based on our experience, this research will find application in several areas of Expert Systems research. Our previous research has demonstrated that work in knowledge-based tutoring yields results that feed back to expert systems problems such as how to represent knowledge, how to interact with experts to acquire new knowledge, and how to explain reasoning to expert system clients. Application can be expected in all areas of expert system research relevant to the Navy, including, but not limited to: C^3I [Baciocco 81, Bechtel 81], computer-aided decision making in restricted environments (such as aboard submarines-- [Henderson 81]), and of course training in complex diagnostic and interpretative skills.

3. Overview of Progress

GUIDON was first conceived as an extension of the explanation system of the MYCIN consultation program. This previous research provided the building blocks for a teaching program:

- modular representation of knowledge in production rules
- English translation of the internal rule representation
- a developed "history trace" facility for recording reasoning steps
- representation in the system of the grammar of its rules so they can be parsed and reasoned about by the system itself
- an explanation subsystem with a well-developed vocabulary for the logical kinds of questions that can be asked about the MYCIN's reasoning ("Why didn't you ask X?" "How did you use X to conclude about Y?").

With this foundation, we constructed an tutoring program that would take MYCIN's solution to a problem, analyze it, and use it as the basis for a dialogue with a student trying to solve the same problem. Several hundred tutoring rules were developed, organized into "discourse procedures" for carrying on the dialogue (offering advice, deciding whether and how to interrupt, etc.) [Clancey 79a]. Student modelling rules were used to interpret student's partial problem solutions in terms of MYCIN's knowledge, and then the model fed into tutoring decisions concerning how much to tell the student and when to test his understanding.

The 1978 proposal to ONR for GUIDON research outlined investigation of both problem solving and teaching strategies. With the program so well-developed, it was expected that early experimentation could be done with alternate teaching approaches. However, during preliminary discussions with other reseachers in this field, a key question was repeatedly raised. To paraphrase John Brown (August 2, 1978 at Stanford University):

> "What is the nature of the expertise to be transmitted by this system (GUIDON)? You are not just unfolding a chain of inferences; there is also glue or a model of process.... What makes a rule click?"

Following this lead, we began to concentrate on the nature of the expertise to be taught. GUIDON'S interactions were studied, particularly the kind of feedback it was able to provide to incorrect partial solutions. The inability of the program to provide strategical guidance--advice about what to do next--revealed that the "glue" that was missing had something to do with the system of rules as a whole. With *400* rules to learn, there had to be some kind of underlying logic that made them fit together: the idea teaching a set of weakly structured rules was now seriously in question. Significantly, this issue had not arisen in the years of developing MYCIN, but was now apparently critical for teaching, and probably had important implications for MYCIN's explanation and knowledge acquisition capabilities as well.

During 1979-80 a study was undertaken to determine how an expert remembered MYCIN's rules (the "model of process" glue) and how he remembered *to use them.* This study (outlined in [Clancey 81a]) utilized several common AI methods for knowledge acquisition, but built upon them significantly through the development of an *epistemological framework* for characterizing kinds of knowledge (detailed in [Clancey 81 b]). The expert's explanations were characterized in terms of: strategy, structure, inference rule, and support (see Section 3 of [Clancey 81b]). With this kind of framework, discussions with the expert were more easily focussed and experiments devised for filling in the gaps in what we were told.

By the end of 1980, we had formulated and implemented a new, comprehensive psychological model of medical diagnosis [Clancey 81c]. NEOMYCIN is a consultation program in which MYCIN's rules are reconfigured according to our epistemological framework. That is, the knowledge representation separates out the inference rules (simple data/hypothesis associations) from the structural and strategic knowledge: *we separate out what a heuristic is from when it is to be applied.* Moreover, the strategies and structure we have chosen model how an expert reasons. We have attempted to capture his forward-directed inferences, his "diagnostic task structure," and the types of focussing strategies he uses. This explicit formulation of diagnostic strategy in the form of meta-rules is exactly the material that our original proposal only mentioned as a hopeful aside. Throughout 1981 we have been fine tuning NEOMYCIN, investigating its applicability to other domains, and exploiting it as the foundation of a student model.

Our 1978 proposal anticipated this possible shift in emphasis from teaching strategies to the study of the knowledge to be taught. Referring to the original objectives, by March 1982 we had accomplished the following specific points:

- explore the structural and strategical meta-knowlege that is important for MYCIN/GUIDON to have... (epistemological study)
- construct a student model from this meta-knowledge... (IMAGE)
- determine what meta-strategies are useful for diagnostic reasoning... characterize the structural meta-knowledge for a large knowledge base... (NEOMYCIN)
- characterize the degree of domain independence of the program and its planning knowledge for doing diagnosis... (investigation of computer failure diagnosis).

In addition, an explanation system for NEOMYCIN now under construction will illustrate how the structural and strategic knowledge enters into explanations. The techniques we are developing will be used directly in GUIDON2, the version of GUIDON that tutors from NEOMYCIN's knowledge base.

4. Conclusions

A well-structured knowledge base can be used in multiple ways. It can be used in a consultative mode to solve problems. It can also be used as a store of subject material to be conveyed to a student. However, good problem-solving performance alone does not ensure that the knowledge base is suitable for application to teaching. Implicit knowledge cannot be explained. It is advantageous to represent hierarchical subtype and causal relations explicitly. It is advantageous to separate domain specific knowledge from the procedures for using this knowledge. *Articulated representation* of knowledge enhances its use in multiple ways.

A computer program that keeps records of its reasoning steps and has knowledge about its own representation can be adapted to provide explanations of its reasoning. However, complete and clear explanations are not necessarily adequate for teaching a student. People do diagnosis by a hypothesis-directed approach. It is advantageous to teach diagnosis in terms of forming and testing hypotheses: the knowledge base and reasoning procedures should approximate a model of human reasoning. A psychological model provides a basis for interpreting student behavior and providing useful, timely advice in a tutorial program.

Our interest in using a knowledge base for teaching has changed our ideas about how expert systems should be built. In constructing MYCIN, we were *programming.* with an AI language, filling in the production rule formalism. As for most programmers, our interest was mainly in getting the right answers, with only secondary attention given to structuring our system to make our methods clear. This design knowledge turns out to be essential for explanation and teaching.

In constructing NEOMYCIN, we started with a theory of diagnosis and of knowledge relations. We used and extended suitable AI languages so as to model the theory. From this theoretical perspective, we look suspiciously upon scoring functions and weights that are written for computational reasons, combining and selecting because *some* decision needs to be made. Instead, we constantly ask, "What do people do? What principles underlie that decision?" We then choose a representation that not only computationally produces the answer we desire, but also makes the principles clear.

5. Acknowledgments

As graduate students, Bob London and Reed Letsinger contributed significant ideas and programming effort to the work reported here. Other students associated with the project are: Conrad Bock, Diane Warner, Glenn Rennels, Paula McKenzie, and Vlad Rutenburg. Discussions with Ted Shortliffe and Jim Bennett have always been helpful. Computing resources have been provided by the SUMEX-AIM facility (Contract RR-00785). The first section of this report is based on a schema extracted from the introduction of Norman's final report [Norman 79].

References

[Anderson 80] Anderson, J. R., Greeno, J. G., Kline, P. J., and Neves, D. M.
Acquisition of problem-solving skill.
Technical Report 80-5, Carnegie Mellon University, 1980.
To appear in J. R. Anderson (Ed.), Cognitive Skills and their Acquisition.

[Baciocco 81] Baciocco, RAdm. A. J. Jr., USN.
Artificial Intelligence and C^3I.
Signal 36(1):23-28, Sept, 1981.

[Bechtel 81] Bechtel, R. J.
Selected Artifical Intelligence systems and their relevance to command and control.
1981.
Code 8242, Naval Ocean Systems Center.

[Brown 75] Brown, J. S., and Burton, R.
Multiple representations of knowledge for tutorial reasoning.
In D. G. Bobrow and A. Collins (editor), *Representation and Understanding,* pages 31 l-349. Academic Press, 1975.

[Brown 78] Brown, J. S. and Burton, R. R.
Diagnostic models for procedural bugs in basic mathematical skills.
Cognitive Science 2(2):155-192, 1978.

[Brown 80] Brown, J. S. and VanLehn, K.
Repair theory: a generative theory of bugs in procedural skills.
Cognitive Science 4(4), 1980.

[Burton 79] Burton, R. R.
An investigation of computer coaching for informal learning activities.
The International Journal of Man-Machine Studies 11:5-24, 1979.

[Carbonell 70] Carbonell, J. R.
Mixed-initiative man-computer instructional dialogues.
Technical Report 1970, Bolt, Beranek and Newman, 1970.

[Carr 77] Carr, B. and Goldstein, I.
Overlays: A theory of modeling for computer-aided instruction.
Al Memo 406, Massachusetts Institute of Technology, 1977.

[Chi 80] Chi, M. T. H., Feltovich, P. J., Glaser, R.
Representation of physics knowledge by experts and novices.
Technical Report 2, University of Pittsburgh, Learning Research and Development Center, 1980.

[Clancey 79a] Clancey, W. J.
Tutoring rules for guiding a case method dialogue.
The International Journal of Man-Machine Studies 11:25-49, 1979.

[Clancey 79b] Clancey, W. J.
Dialogue management for rule-based tutorials.
In *Proceedings of the Sixth IJCAI.* 1979.

[Clancey 79c] Clancey, W. J.
Transfer of Rule-Based Expertise through a Tutorial Dialogue.
PhD thesis, Stanford University, August, 1979.
STAN-CS-769.

[Clancey 81 a] Clancey, W. J.
Methodology for building an intellient tutoring system.
1981.
To appear in *Problems of Methodology in Cognitive Science,* Kintsch (Ed.).

[Clancey 81 b] Clancey, W. J.
The epistemology of a rule-based expert system: a framework for explanation.
1981.
Submitted to Artificial Intelligence.

[Clancey 81 c] Clancey, W. J. and Letsinger, R.
NEOMYCIN: Reconfiguring a rule-based expert system for application to teaching.
In *Proceedings of the Seventh IJCAI,* pages 829-836. 1981.

[Collins 76] Collins, A.
Processes in acquiring knowledge.
In R. C. Anderson, R. J. Spiro, and W. E. Montague (editor), *Schooling and the acquisition of know/edge,* pages 339-363. Erlbaum Associates, 1976.

[Feltovich 80] Feltovich, P. J., Johnson, P. E., Moller, J. H., and Swanson, D. B.
The role and development of medical knowledge in diagnostic expertise.
1980.
Presented at the 1980 Annual meeting of the American Educational Research Association.

[Genesereth 81] Genesereth, M.
The role of plans in intelligent tutoring systems;
In D. Sleeman and J. S. Brown (editor), **Intelligent** *Tutoring Systems,* . Academic Press, 1981.

[Hayes-Roth 80] Hayes-Roth, B.
Human Planning Processes.
Technical Report R-2670-ONR, The Rand Corporation, December, 1980.

[Henderson 81] Henderson, J. V., Ryack, B. L., Moeller, G., Post, R., and Robinson, K. D.
Use of a computer-aided diagnosis system aboard patrolling FBM submarines.
Technical Report 938, Naval Submarine Medical Research Laboratory, April, 1981.

[Johnson 81] Johnson, P. E., Duran, A., Hassebrock, F., Moller, J., Prietula, M., Feltovich, P. J., and Swanson, D. B.
Expertise and error in diagnostic reasoning.
Cognitive Science 5(3):235-283, 1981.

[Lesgold 81] Lesgold, A. M., Feltovich, P. J., Glaser, R., Wang, Y.
The acquisition of perceptual diagnostic skill in radiology.
Technical Report PDS-1, University of Pittsburgh, Learning Research and Development Center, sept, 1981.

[Norman 79] Norman, Donald A.
Studies of learning and self-contained educational systems, 1976- 1979.
Technical Report 7902, University of California, San Diego, June, 1979.

[Papert 70] Papert, S.
Teaching children programming.
North Holland, 1970.

[Polya 57] Polya, G.
How to Solve It: a new aspect of mathematical method.
Princeton University Press, 1957.

[Rumelhart 80] Rumelhart, D. E. and Norman, D. A.
Analogical processes in learning.
Technical Report 8005, University of California, San Diego, Sept, 1980.

[Schank 81] Schank, R. C.
Failure-driven memory.
Cognition and Brain Theory 4(1):41-60, 1981.

[Schmidt 80] Schmidt, C. F.
Plan recognition and revision: understanding the observed actions of another actor.
Technical Report CBM-TR-115, Rutgers University, Laboratory for Computer Science, Sept, 1980.
Presented at the 88th Annual Convention American Psychological Association.

[Stevens 78] Stevens, A. L., Collins, A., and Goldin, S.
Diagnosing student's misconceptions in causal models.
Technical Report 3786, Bolt Bernek and Newman, 1978.