A GUIDE FOR THE APPLICATION OF PERFORMANCE STRUCTURE-ORIENTED CAI IN MILITARY JOBS

Dr. J. W. Rigney, Behavioral Technology Laboratories, U. Southern California
and
Dr. A. S. Blaiwes, Research Psychologist, Naval Training Equipment Center

INTRODUCTION

The University of Southern California and the Naval Training Equipment Center, under ARPA funding, are cooperating in an effort to develop a system which will serve to facilitate the production of appropriate Computer-Aided Instruction (CAI) for the Navy. This system, basically taking the form of a model or theory of CAI, is in part derived from and evaluated by empirical studies as described in other Naval Training Equipment Center technical reports by the present authors (Rigney, et al., 1973; Rigney, et. al., in preparation). Because this empirical aspect of the research exhausts major portions of the resources available for the project, explication of the system contained herein is in its earliest stages. It is published here mainly for heuristic reasons and is not intended to be used as a refined set of guidelines for developing CAI materials.

Even in this initial form, however, the system can suggest to course authors various components of CAI and gross steps associated with the development of these components that need to be considered in the process of course construction. Further development of the system will consist of endeavors to expand upon current capabilities by offering: (a) specific instructional approaches for the components forming the structure of CAI; (b) computer programs, capable of implementing the suggested instructional approaches, which are general enough to apply in a range of subject matter areas, and (c) computer capabilities for generating some portions of the CAI specifications for a given application. Thus, CAI program developers will be able to use the system as an aid to deciding upon instructional approaches appropriate for their particular teaching objectives. Further, they even will be able to obtain computer programs which essentially are ready for application in their training program.

The development of generalizable instructional approaches and the supporting computer programs is a prime reason for viewing the extensive CAI course construction activities as essential to the project. The empirical research, however, serves additional important functions, among which is the ancillary contribution of developing cost-effective means for teaching skills for critical Navy jobs. The Radar Intercept Officer's (RIO's) job was the first technical area addressed in this way. Developing CAI materials for teaching the utilization of the AWG-9 system for maintaining the F-14 aircraft is being considered for the continuation of the empirical aspect of the research on this project.

A GENERAL CHARACTERIZATION OF CAI: A MULTI-LEVEL INTERACTION BETWEEN STUDENT AND PROGRAM

Student-Program Interface Characteristics

Sources of information for the student must have sufficient capability and flexibility to allow satisfactory simulation of sources of information in the job performance environment. Similarly, student response-sensitive features in the interface must permit satisfactory representation of these features in the job performance environment.

The student-program interactive loop is completed at the interface. The student has built-in feedback loops that allow him to control his performance in relation to information inputs. The computer program must have at least enough feedback loops built into it to allow sensing and judging of student responses and to respond in turn.

Both student and program utilize memories during the interaction. The program must store a sufficient history of the student's responses in relation to the instructional sequence. This history is one of the essential requirements for adaptive control. The student must use short term memory as a working (or "scratch-pad") memory where he can temporarily store information from two sources: from his own long term memory and from the course in the interface.

The objective of the game is to so schedule the macro-interactions that the student will become capable of self-directed performance in the job environment. The program must contain the logic either to compose this schedule for some period in the future, or to assist the student in composing an effective schedule. In either case, scheduling operations become concerned with larger chunks of the instructional sequence, called macro-interactions in figure 1.

Program Surface and Deep Structures

The computer program communicates with the student via input-output subroutines which accept student responses and which generate visual and audio displays of information. These subroutines in turn interact with the "deep

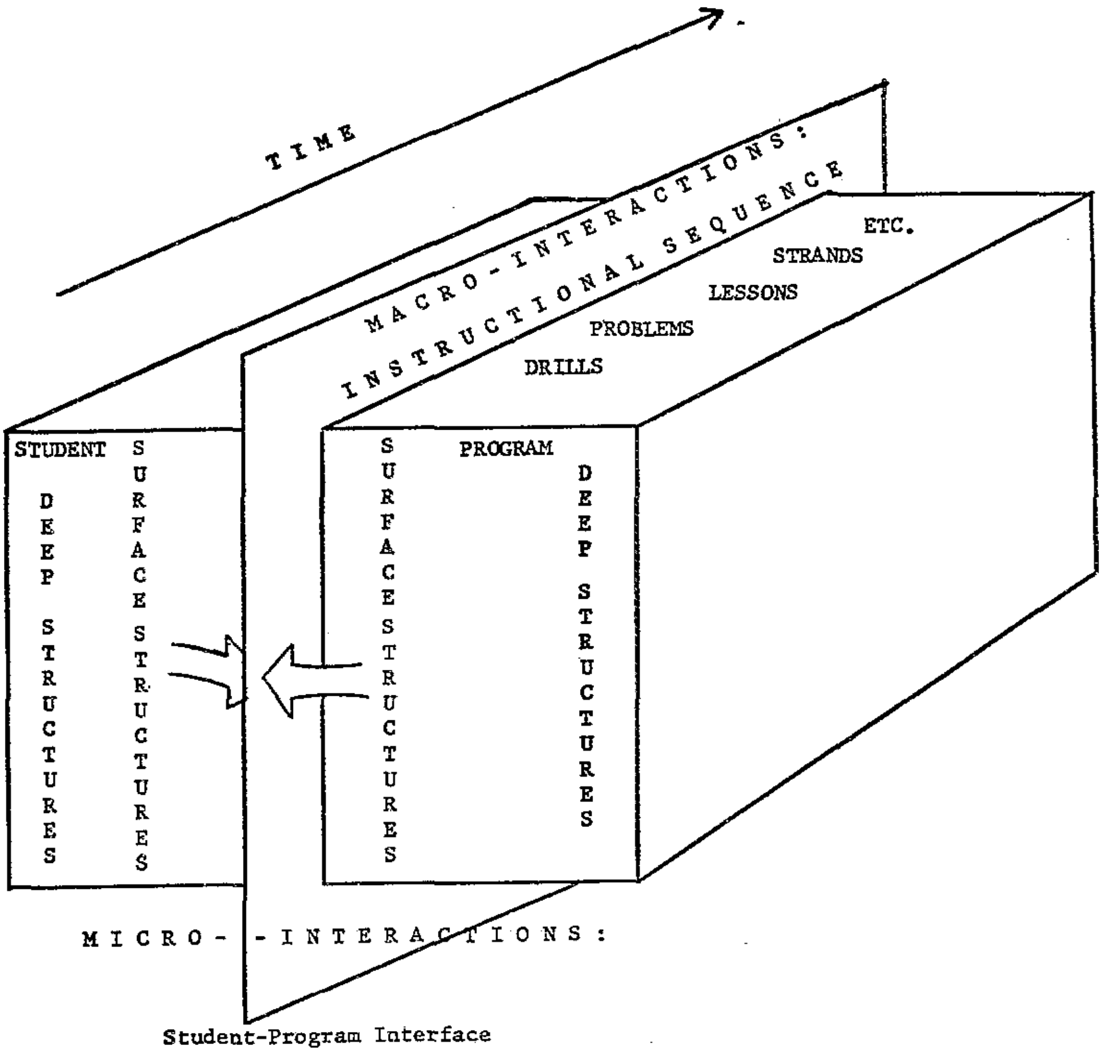

Figure 1. CAI as Interactive Communication Between Student and Computer Program Extending Over Time

structure" subroutines that perform the various functions required to sustain the micro-interaction and to vary it over extended periods of time according to the configuration established for the macro-interactions. (These relationships are depicted in figure 1.)

Both micro- and macro-interactions may be pre-established as a form of programmed instruction, in which case the computer program deep structure can be relatively simple. Alternately, one or both levels of interaction may be generated while the student is on-line with the system. In this case, the program deep-structure can become exceedingly complex. PSO CAI uses the latter type of program, as will be described in more detail later.

Student Surface and Deep Structures

The more important surface features on the student side of the interaction are the tasks he must learn to perform. Task description and analysis down to a suitable level is necessary to identify the job, tasks and skills that are to be taught. The primary requirement is to analyze down to a detailed enough level to uncover programming and data base requirements.

In the ideal case, it also would be possible to describe suitable deep structure that students should possess. Consider, for example, the instructional time that is wasted in teaching students "theory" in training courses that turns out not be be needed or even useful for their job performance because we do not know the true nature of deep structures. Gagne (1970) pioneered thinking about the nature of deep structures in education and training. Subsequently, there have been striking developments in modeling the structure of long-term memory, which are excellently summarized in Anderson and Bower (1973). Eventually, theorists may be able to represent deep structures stored in long-term memory and to operate upon these to produce surface structures in ways characteristic of human learning and memory. This would have enormously important implications for education and training in that instruction can be designed to be compatible with the deep structure of the student.

Currently, theoretical interest is centered in semantic information processing. Written or spoken sentences are said, since Chomsky (1964) originated the concept, to be the surface structure of language. This is what is seen, heard, or spoken. But this surface structure can vary a great deal and still convey the same meaning. Sentences can be paraphrased without destroying the essential information they communicate. There is, Chomsky said, a deep structure that conveys the meaning, and that is the basis for generating the surface structure of language.

Language is a form of performance; and an analogy can be drawn between other forms of human performance and language performance. Beginners must stumble along or be led by the hand until they learn a deep structure that will let them generate that desired performance. It gives the student the opportunity to organize what he already knows and to identify what he needs to know to generate the surface structure of whatever performance is required for the job.

It is proposed that there are powerful self-organizing processes in the central nervous system that function to develop deep structures in ways as yet very poorly known. These self-organizing processes have the opportunity to function during practice in performing surface structures.

It also should be pointed out that language is used in learning to perform tasks. As a representational process, as a means for communicating instructions, as the basis for creating shared contexts which allow humans to "stand aside and look" at what they have done, are doing, or are going to do, language is a universal and indispensable learning tool. In fact, it is possible that the same deep structures in long-term memory — what Anderson and Bower (1973) call the "strategy-free component of memory" — are the basis for all cognitively-controlled performance.

A MORE DETAILED LOOK AT CAI: ELEMENTS IN CAI SYSTEMS

Figure 2 illustrates the necessary elements in a CAI system. Each of these is described very briefly as follows.

Internal Processing

This box represents the student. There always is some definable population of students--some "target" population, for whom the CAI system is to be designed. We need to know characteristics of this population as a basis for the macro-interaction provided by the adaptive controller.

As a form of drill and practice, PSO CAI assumes the students already are familiar with the "theory of operation" or are learnit in a parallel course. This form of CAI normally would be embedded in the total curriculum. Therefore, the screening that is done in the broader context should suffice for prerequisite requirements.

Student-Program Interface

The problems here are to identify the stimulus displays, the response structures and the response records that will be required. It is probable that work on this part of the specification should be deferred until other parts of the system have been analyzed.

Student Data

This includes the student sufficient history computer and student records. The latter would include processed response records plus other information about the student, e.g., intelligence test scores.

A word or two about the concept of a student sufficient history computer is in order. Atkinson and Paulson (1972) stated that, "An index summarizing the information in a student's response protocol is a sufficient history if any additional information from protocol would be redundant in the determination of the student's state of learning. The concept is analogous to a sufficient statistic. . ." The word "computer" is added to indicate not a piece of hardware, but software required to compute the sufficient history from response records and student records.

The same remarks about priorities of analysis apply here: What constitutes a student sufficient history will be determined by

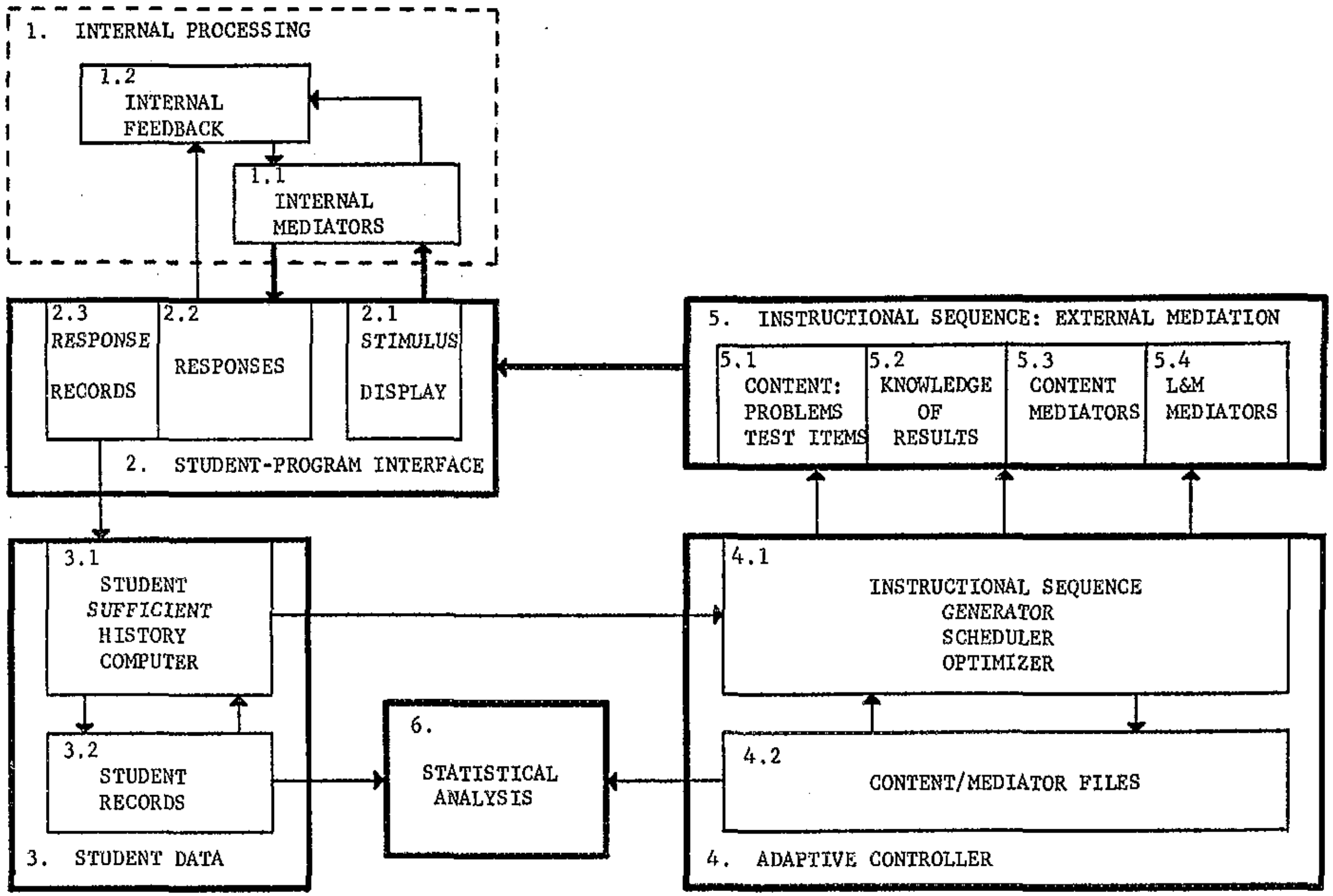

Figure 2. Outline of Major Elements in CAI

what is done with these data, which will depend on who will use the data and what they will be used for. Several different sufficient histories may be needed.

Adaptive Controller

This includes content/mediator files, essentially storage of course materials; an instructional sequence optimizer; an instructional sequence scheduler; and an instructional sequence (content) generator. These three instructional operators require some explanation.

An optimizer would be some method for improving the effectiveness of the CAI system, by optimizing learning rates under some set of constraints. Atkinson and Paulson (1972) have described general procedures for going about this. The optimizer would identify an optimal instructional sequence for each individual student. The composition and sequencing of instruction is the principal way, if not the only way, to influence learning and retention in the CAI system.

The optimizer would control an instructional sequence scheduler. If no optimizer is used, there still must be a scheduler. Instruction must be sequenced by some mechanism and that mechanism must be described in the specifications for the CAI system.

The instructional content generator could be either of two types; software that "makes up" the instruction on the spot from a data-structure, as in the TASKTEACH programs (see Rigney and Towne, 1970) or a team of instructors who write programmed instruction frames, and generate the entire sequence ahead of time. Unfortunately, the range of applications in which the instructional content can be dynamically generated by a computer program is currently quite narrow.

For the instructional content generator, the two major questions are: "What is the surface structure of the performance to be taught?" and "What are the content and learning and memory (L&M) mediators that will be required in the training?" Content mediators are the operations and concepts in the material to be taught that bridge between stimulus and response. For example, in algebra, the mental operations and concepts required to solve a quadratic equation are content mediators. L & M mediators are general operations the student may perform on broad categories of material to improve learning and retention of the material. Mental imagery is an example of this.

The content/mediator files in the adaptive controller store the material that is to be organized into an instructional sequence, or that is already organized into some "lesson" format.

The optimizer, scheduler, and generator mechanisms in the adaptive controller tend today to be relatively crude procedures. These are areas that should receive much more intensive research and development since they are at the core of the CAI system, and will determine its effectiveness.

Instructional Sequence

This sequence contains knowledge of results, subject-matter, and external mediators. It is the "input tape" to the student. Knowledge of results, something the system provides for the student, is distinguished from internal feedback, which is something the CNS provides using information that comes to it both from the external and the internal environments. Knowledge of results provides some information for internal feedback, but this information may or may not be used. The knowledge of results provided by the system sometimes may be superfluous. Two types of external mediators, content mediators and learning and memory mediators, are defined.

Since most material is at some intermediate level of difficulty in a roughly hierarchical structure, there usually are a number of different concepts, relations, and skills to be learned. These must somehow be organized into a sequence and presented to the student's very limited input system in serial order. Thus, the fundamental problem is how to schedule the instructional sequence in a way that will cover the different concepts, rules, and operations to be taught, and that will lead to optimum learning and retention rates.

As represented in figure 2, the instructional content would have been already composed by the generator and would be put into serial order by the optimizer scheduler, all in the adaptive controller. Thus, the instructional sequence is the output of these operators in the adaptive controller.

Feedback Loops

A CAI system contains information channels that could be used to learn about the adequacy of the instructional sequence, as well as to instruct and to provide knowledge of results to the student.

These feedback loops may be essentially open, because of long time delays, or essentially closed. For example, the feedback loop providing information about the adequacy of the instructional sequence usually is open. The very long time required to revise instructional materials usually means the revised version will be used on a different sample of students.

As suggested in the internal processing box in figure 2, the organization of the human nervous system includes feedback loops. A recent conception of this organization by Powers (1973) makes a persuasive case for hierarchically controlled negative feedback loop circuits. Powers describes nine levels, from stimulus intensity to system concepts.

It is not unlikely that some kind of hierarchical control will be found, in the future, to control learning processes; e.g., storage and retrieval in long term memory.

A SPECIFIC APPROACH TO CAI

Performance-Structure Oriented (PSO) CAI Instructional Strategy

The section of figure 3 labeled "front-loading" refers to the points for students to enter the instructional system. It usually is the case that students need preliminary instruction before they can start to practice performing job skills. They need to absorb information specifically relating to, and supporting practice in performing job skills. One function of the front-loading section is to do this. Another purpose of this section is to bring students "up-to-speed" with respect to subskills in which they are deficient, before they begin practice on performing job skills. Generally speaking, the three sections of this diagram are coordinate with instructing, practicing, and testing functions in training. Of course, the instructional sequence may cycle through these functions over and over, and they may be scheduled in the sequence in different configurations, which are dynamic features not implied in figure 3.

The box labeled "instructional aids" represents instructional techniques which would be applied during practice to facilitate the development of proficiency. External feedback (knowledge-of-results), induction of mental imagery, visual analogies of invisible processes, "templates" to guide performance, fall into this category.

According to the conception of job performance as the integrated sequencing of a collection of subskills, students will require a certain amount of practice before they can sustain this integrated performance. Until then, they will have to function at lower levels. The chain of boxes labeled "subskill drills" is intended to indicate these levels of incomplete integration. They would be entered via remedial loops. Some of the subskills involved might also be scheduled as front-loading, particularly if they are common deficiencies in the population of students. In Section III of the diagram, coordinate with the testing function in instruction, the student's ability to sustain job task performance at criterion levels is established, and the student leaves the course. The RIO trainer is an example of the application of this general strategy (Rigney, et. al., 1973).

Behavioral Technology Laboratories Self-Standing CAI System

There already are general specifications for PSO CAI that serve to structure and to simplify the task of generating specifications for a particular application. PSO CAI is based on certain hardware capabilities that can be assumed to be available for any application:

1. Interactive, animated computer graphics.

2. Input-output interfacing for computer-controlled front-panel simulators, voice synthesizers, and image projectors.

3. Light pen (or the equivalent) and keyboard for student response inputs. It also must be possible to interface with the computer special response-input devices, e.g., control sticks, and analog and digital controls and switches on man-machine interfaces.

4. Economical random access disk storage.

5. CPU cycle-time sufficiently fast to permit real-time simulation and real-time animation.

6. Self-standing, portable system.

The problem for generating hardware specifications for specific applications, given the above general capabilities, is to identify the additional, special man-machine interfaces these applications would require, if any. Task analyses would be used to identify these special man-machine interface requirements.

PSO CAI also is based on a general structure for the software, i.e., the computer programs. The software capabilities generally required for PSO CAI can be characterized as follows:

1. Unique interaction with each student, via animated computer graphics and/or special student-program interfaces. The computer program must accept the student's input, calculate the values of variables required to respond to this input, and display these values in suitable format to the student.

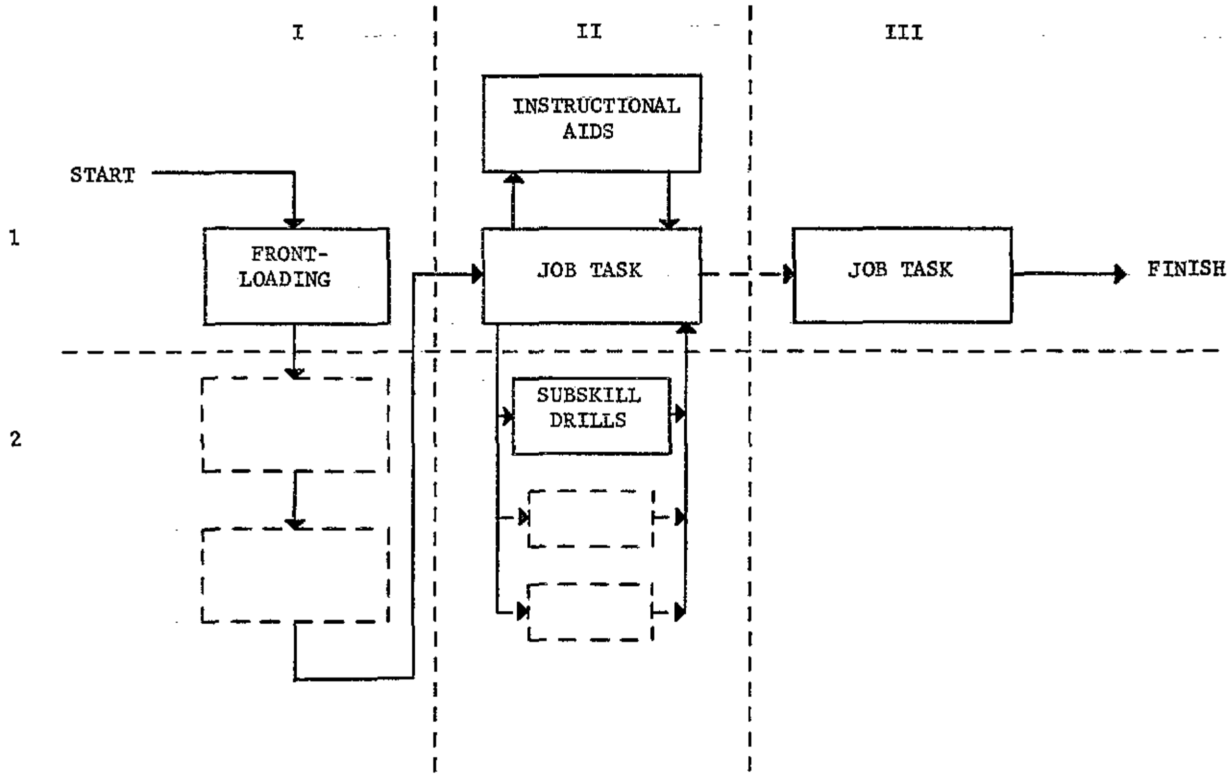

Figure 3. General Flowchart for PSO CAI Instructional Sequence

2. Continuous updating of the simulated environment that is the context of the training. (In the case of the RIO trainer, which was a real-time problem, this updating occurred every tenth of a second and every time a student responded.)

3. Continuous response-by-response tracking of student progress. This is necessary for the interaction with the student, for external feedback, and for data for student sufficient histories.

4. Instructional functions that can be placed under either student or program control.

5. Response-sensitive adaptive control over the instructional sequence.

6. Multiple dependent variable recording and analysis.

7. Generation of the instructional sequence and the interaction with the student from program logic operating on simple data-structures.

TECHNIQUES FOR GENERATING PSO CAI SPECIFICATIONS

A process for generating specifications for PSO CAI might be as illustrated in figure 4. The analyst starts with available information about what is to be done and how, implied by the four arrows at the top of the diagram, and derives hardware and software specifications, using procedures implied by the interconnected boxes in the diagram. When he is finished, he will have generated and organized these two categories of specifications for each necessary element of a CAI system: hardware and software.

In figure 4, contextual structures refer to the "subject matter" of the training. If it is a tactical or operational job, the contextual structure may be air intercepts. The geometry of air intercepts would have to be analyzed, so that the air intercept environment could be simulated. If the job is maintenance of devices, then the structure of the device that the student is to learn to maintain must be analyzed. Generally, tactical jobs require quite different data bases and program logic than maintenance jobs.

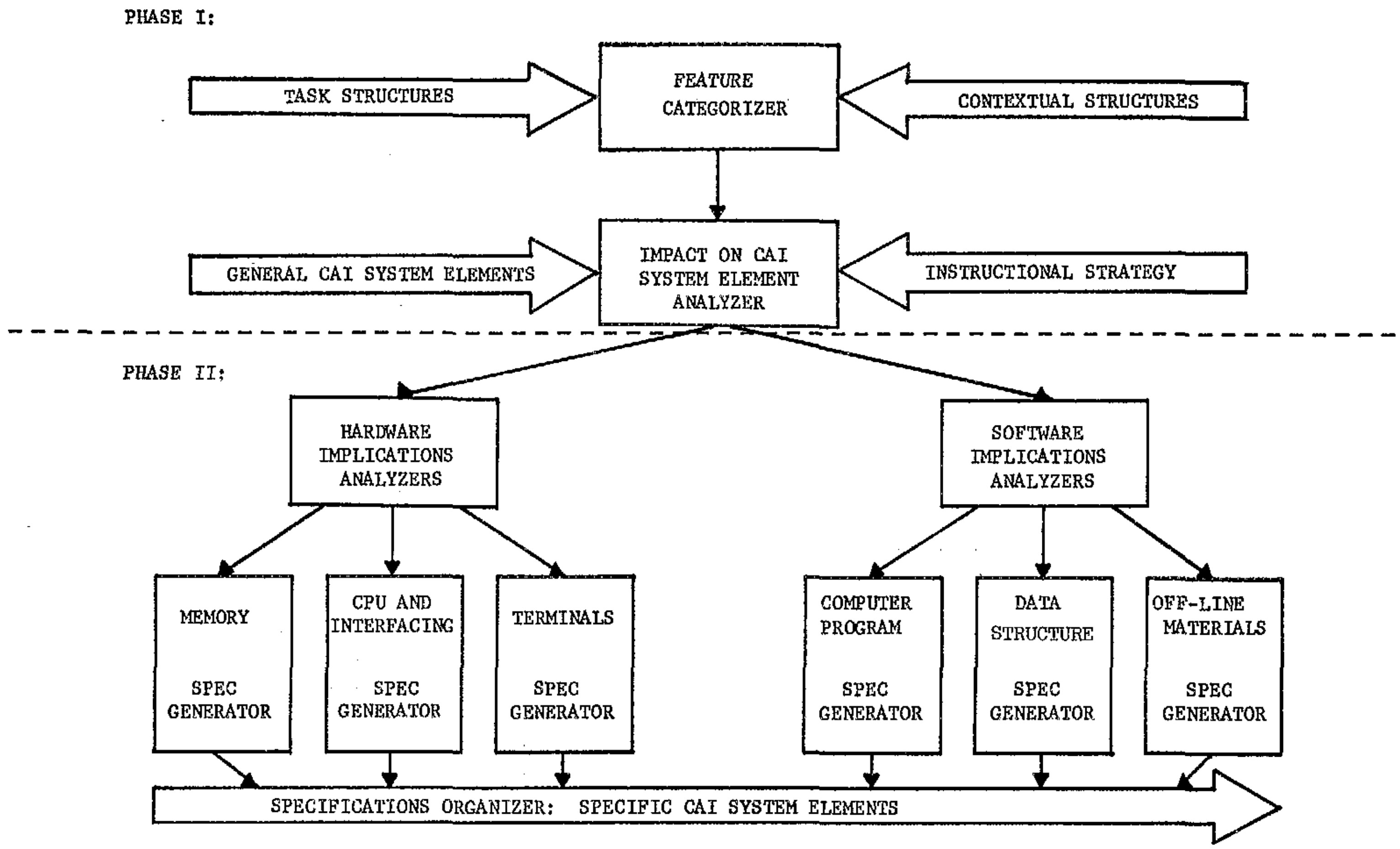

Figure 4. Tentative Outline of Procedure for Generating CAI System Specifications from an Analysis of the Performance to be Taught

The feature categorizer in figure 4 abstracts from task and contextual structures, information for the way CAI systems elements are designed. Some examples illustrate this. In the RIO trainer, the air intercept environment, specifically interception problems, required that there be a situational display showing fighter and bogey headings, speeds, turns, etc. This could be met by real-time animation using computer graphics. In turn, real-time animation requires a local processor and memory (in the terminal). As another example, in AWG-9 maintenance training, an important contextual structure is the fact that build-in-test (BIT) sequences exist to test different parts of the system. This is done under program control. Each BIT sequence is composed of a number of tests and subtests, which are performed automatically and in sequence, in contrast to simpler devices where the technician selects the next test to perform. The implication of this is that there must be an executive routine in the trainer to sequence these tests.

The feature categorizer looks at these structures and identifies features that would impact on one or more of the six necessary CAI system elements (shown in figure 2). The implementation of these elements by hardware and/or software then is determined and specifications for this implementation are generated in one or more of the six boxes at the bottom of the diagram. Specifications for computer programs, for example, would be in the form of program flowcharts.

The instructional strategy is concerned primarily with the content and sequencing of instruction. We can draw an analogy, albeit a loose one, between a tape being fed into a Turing machine and an instructional sequence being "fed into" a student. The input tape is the principal means for changing the behavior of the machine in one case and of the student in the other case. The instructional strategy will impact on the composition of the instructional sequence in the way of determining the items, problems, drills, external feedback, and other instructional aids to be used, and the order of their use. The instructional sequence, in turn, will require particular kinds of data structures and storage. The scheduling of the instructional sequence will require, if done adaptively, a student monitor program, and this program will require student sufficient histories and some type of optimizer.

Phase I and Phase II in the diagram refer to the fact that it usually is convenient to divide the development of specifications into two phases. In Phase I, operations will be mostly concerned with information gathering and integration. It may be necessary to perform task, equipment, and tactical analysis as part of the development, or some of this information may be available in documentation. In either case, a subject matter expert is a practical necessity to select and to interpret documents and to fill in gaps in information.

A good way to integrate Phase I information is to construct an instructional flowchart. This will be an instructive exercise, because it forces planners to put down the surface structure of the student-program interaction in detail. The instructional flowchart will serve as the principal guide for Phase II operations, although it must be supplemented by other sources of information. Usually, several iterations are required before a flowchart can be produced at a sufficiently detailed level to be useful.

Information supplementing the instructional flowchart would consist largely of the results of examining implications for (a) each of the general CAI system elements, in turn; (b) task and contextual structures; and (c) the instructional strategy. This exercise will force the analysts to make at least provisional decisions about how the operations required for each element will be implemented. During this exercise, the instructional flowchart will serve as a roadmap of features that must be considered. In Phase II, the objective is to describe the specific elements in the CAI system that will be designed to do the training. Given the outputs of Phase I (the instructional flowchart and the preliminary delineations of element operations) the analytical problem can be divided into hardware and software sections, and the effort can be concentrated on developing specifications in each of the six boxes at the bottom of the diagram.

In summary, the whole process consists of relating specific applicational informations (task and contextual structures) to general requirements of CAI system elements and to an instructional strategy to derive the specifications for specific implementations of CAI system elements that will do the job of training

REFERENCES

Anderson, J. R. and Bower, G. H. Human Associative Memory. Washington, D. C.: V. H. Winston and Sons, Inc., 1973.

Atkinson, R. C. and Paulson, J. A. An approach to the psychology of instruction. Psychological Bulletin., 1972, 78(1), 49-61.

Campbell, D. T. and Stanley, J. C. Experimental and Quasi-Experimental Designs for Research. Chicago: Rand McNally College Publishing Co., 1963

Chomsky, N. Current issues in linguistic theory. In J. A. Fodor and J. J. Katz (Ed.), The Structure of Language: Readings in the Philosophy of Language. Englewood Cliffs, N. J.: Prentice-Hall, 1964.

Gagne, R. M. The Conditions of Learning. New York: Holt, Rinehart and Winston, 1970, 2nd Edition.

Hays, W. L. Statistics for Psychologists. New York: Holt, Rinehart and Winston, 1963.

Powers, William T. Behavior, The Control of Perception. Chicago: Aldine Publishing Co., 1973.

Pribram, K. H. A review of theory in physiological psychology. In VOL. XI of Annual Review of Psychology. Palo Alto, Calif.: Annual Review, Inc., 1960.

Rigney, J. W., Morrison, D. K., Williams, L. A. and Towne, D. M. Description and Initial Evaluation of a Computer-Based Individual Trainer for the Radar Intercept Officer. Los Angeles: Behavioral Technology Laboratories (NAVTRAEQUIPCEN 71-C-0219-1), 1973.

Rigney, J. W., Morrison, D. K., Williams, L. A. and Towne, D. M. A Guide for the Application of Performance Structure Oriented CAI in Naval Training. Los Angeles: Behavioral Technology Laboratories (NAVTRAEQUIPCEN TR, 1974) (in press).

Rigney, J. W., Morrison, D. K., Williams, L. A. and Towne, D. M. Field Evaluation of Model II of the Computer-Based Individual Trainer for the Radar Intercept Officer. Los Angeles. Behavioral Technology Laboratories (NAVTRAEQUIPCEN TR) (in preparation).

Rigney, J. W. and Towne, D. M. TASKTEACH: A method for computer-assisted performance training. Human Factors, 1970, 12 (3), 285-296.

ABOUT THE AUTHORS

DR. JOSEPH W. RIGNEY, since 1967, has been Director of the Behavioral Technology Laboratories and Adjunct Professor of Psychology at the University of Southern California at Los Angeles. He received B.S. (Biology) and M.S. (Physiology) degrees from New Mexico State University and the University of New Mexico, respectively, and the Ph.D. (Experimental Psychology) from the University of Southern California. He has published over 67 reports on the subject of training with an emphasis on CAI, to which his group is now fully committed. He has engaged in a number of special assignments with the Navy including: (a) Special Consultant on training to CNO (1954); (b) Technical Director, NATO Advanced Study Institute, Royal Naval College, Greenwich, England (1970); (c) Member, Laboratory Advisory Board for the Naval Research Advisory Committee, Washington, D. C. (1969-1971); and (d) Chairman, Laboratory Advisory Board for Personnel Laboratories, Naval Research Advisory Committee, Washington, D. C. (1971 to present).

DR. ARTHUR S. BLAIWES is a Research Psychologist in the Human Factors Laboratory at the Naval Training Equipment Center. He received a B.S. (1962) from Ohio State University and M.A. (1966) and Ph.D. (1967) in Experimental Psychology from the University of Kentucky. He is currently engaged in a number of research and development programs dealing with learning, retention and transfer problems in military training. He has published and presented a number of papers on the subject of training. He is a member of the Institute of Electrical and Electronic Engineers.