ADAPTIVE INSTRUCTIONAL SYSTEMS:

SOME ATTEMPTS TO OPTIMIZE THE LEARNING PROCESS

by

Richard C. Atkinson

TECHNICAL REPORT NO. 240

November 20, 1974

PSYCHOLOGY AND EDUCATION SERIES

This research was supported by the Office of Naval Research, Contract No. N00014-67-A-0012-0054, by National Science Foundation Grant EC-43997, and by Grant MH-21747 from the National Institute of Mental Health.

INSTITUTE FOR MATHEMATICAL STUDIES IN THE SOCIAL SCIENCES

STANFORD UNIVERSITY

STANFORD, CALIFORNIA

ADAPTIVE INSTRUCTIONAL SYSTEMS:

SOME ATTEMPTS TO OPTIMIZE THE LEARNING PROCESS[1]

Richard C. Atkinson

Stanford University

INTRODUCTION

One cannot help but question the significance of psychology's contribution to the development of effective instructional procedures. On the one hand, psychology has been very influential in the field of education. In the last twenty-five years almost every major innovation in education--programmed textbooks, behavioral objectives, ungraded schools, individually prescribed instruction, computer managed and assisted instruction, token economies, and tailored testing to name a few--can be traced to psychology. In many cases these innovations have not been due to psychologists primarily identified with education, but rather to laboratory scientists whose research has suggested new approaches to instruction. Psychology can be proud of that record of accomplishment. But upon closer examination, it is evident that these accomplishments are not as closely linked to psychological research as many might believe. Psychology has suggested new approaches to education, but these suggestions have not led to sustained research programs that have the promise of producing a truly effective theory of instruction. Rather, psychology seems to provide the stimulus for innovation, but

---

[1]To be published in Klahr, D. (Ed.), Cognition and Instruction. Hillsdale, New Jersey: Erlbaum Associates, 1975.

innovation that has not in turn led to a deeper understanding of the learning process.

Why has psychology not had a more substantial impact? There are several reasons. The brightest and ablest young psychologists usually are not attracted to educational research, and the research that has been done tends to be piecemeal, not pursuing problems in real depth. This picture may change in the near future due to the limited number of jobs for new Ph.D.'s and to society's increasing emphasis on applied research. The more serious problem, however, is that psychologists know a great deal about the acquisition of individual facts and skills, but very little about how these combine to form a meaningful mental structure. Effective methods for acquiring skills and facts are important, but the major problem is the development of knowledge structures that are more than the sum of individual facts. In order to deal effectively with educational problems, we need theories that tell us how knowledge is represented in memory, how information is retrieved from that knowledge structure, how new information is added to the structure, and how the system can expand that knowledge structure by self-generative processes. The development of such theories is under way, and increasingly work in cognitive psychology is moving in that direction. The contributions of Anderson and Bower (1973), Newell and Simon (1972), Rumelhart and Norman (1973), and Schank (1974) are examples of substantial efforts to develop comprehensive theories of cognition, and it is already evident that this work will have implications for education. Such theories will not simply add another wrinkle to educational research, but will lay the foundations for research encompassing a larger set of educationally significant problems than has been considered in the past.

In this paper I want to review the ongoing work in my laboratory that has implications for instruction. Some of that work represents attempts to deal with the issue of complex knowledge structures, whereas some is more restrictive dealing with the acquisition of specific skills and facts. All of the work involves computer-based programs of instruction used on a daily basis in schools and colleges. These programs can best be described as adaptive instructional systems. By that term I mean two things: (1) the sequence of instructional actions taken by the program varies as a function of a given student's performance history, and (2) the program is organized to modify itself automatically as more students complete the course and their response records identify defects in instructional strategies.

Our work on adaptive instructional systems has three foci. One is the development of a course in computer programming for junior college and college students; the second is a course for teaching reading in the first three grades of elementary school; and the third is a foreign-language vocabulary program being used at the college level. This paper will review research on each of these projects.

INSTRUCTION IN COMPUTER PROGRAMMING

Our first efforts to teach computer programming involved the development of a computer-assisted instruction (CAI) curriculum to teach the AID programming language; this course has been used extensively in colleges and junior colleges as an introduction to computer programming

(Beard, Lorton, Searle, & Atkinson, 1973). However, it is a linear, "frame-oriented" CAI program and does not provide individualized instruction during the problem-solving activity itself. After working through lesson segments on syntax, expressions, etc., the student is assigned a problem to solve in AID. He must then leave the instructional program, call up a separate AID interpreter, perform the required programming task, and return to the instructional program with an answer. As the student writes his program with AID, his only sources of assistance are the error messages provided by the non-instructional interpreter.

An inadequacy of the AID course, especially for research purposes, is its limited ability to characterize individual students' knowledge of specific skills, and its inability to relate students' skills to the curriculum as anything more than a ratio of problems correct to problems attempted. The program cannot make fine distinctions between a student's strengths and weaknesses, and cannot present instructional material specifically appropriate to that student beyond "harder" or "easier" lessons. In order to explore the effects of different curriculum selection strategies in more detail, we developed another introductory programming course, capable of representing both its subject matter and student performance more adequately. The internal representation of programming skills and their relationships to the curriculum is similar in some ways to the semantic networks used in the "generative" CAI programs developed by Carbonell and others (Carbonell, 1970, and Collins, Carbonell, & Warnock, 1973).

The BASIC Instructional Program

An important feature of a tutorial CAI program is to provide assistance as the student attempts to solve a problem. The program must contain a representation of the subject matter that is complex enough to allow the program to generate appropriate assistance at any stage of the student's solution attempt. The BASIC Instructional Program (BIP) contains a representation of information appropriate to the teaching of computer programming that allows the program both to provide help to the student and to perform a limited but adequate analysis of the correctness of his program as a solution to the given problem.

To the student seated at a terminal BIP looks very much like a typical timesharing BASIC operating system. The BASIC interpreter, written especially for BIP, analyzes each program line after the student types it, and notifies the student of syntax errors. When the student runs his program, it is checked for structural illegalities, and during runtime "execution" errors are indicated. A file storage system, a calculator, and utility commands are available.

Residing above the simulated operating system is the "tutor," or instructional program. It overlooks the entire student/BIP dialogue and motivates the instructional interaction. In addition to selecting and presenting programming problems to the student, the IP identifies the student's problem areas, suggests simpler "subtasks," gives hints or model solutions when necessary, offers debugging aids, and supplies incidental instruction in the form of messages, interactive lessons, or manual references.

At BIP's core is an information network whose nodes are concepts, skills, problems, sub-problems, prerequisites, BASIC commands, remedial lessons, hints, and manual references. The network is used to characterize both the logical structure of the course and our estimate of the student's current state of knowledge; more will be said about the network later. Figure 1 illustrates the interactions of the parts of the BIP program.

The curriculum is organized as a set of programming problems whose text includes only the description of the problem, not lengthy descriptions of programming structures or explanations of syntax. There is no fixed ordering of the tasks; the decision to move from one task to another is made on the basis of the information about the tasks (skills involved, prerequisites, subtasks available) stored in BIP's network.

A student progresses through the curriculum by writing, and running, a program that solves the problem presented on his terminal. Virtually no limitations are imposed on the amount of time he spends, the number of lines he writes, the number of errors he is allowed to make, the number of times he chooses to execute the program, the changes he makes within it, etc. The task on which he is working is stored on a stack-like structure, so that he may work on another task, for whatever reason, and return to the previous task automatically. The curriculum structure can accommodate a wide variety of student aptitudes and skills. Most of the curriculum-related options are designed with the less competent student in mind. A more independent student may simply ignore the options. Thus, BIP gives students the opportunity to determine their own "challenge levels" by making assistance available but not inevitable.

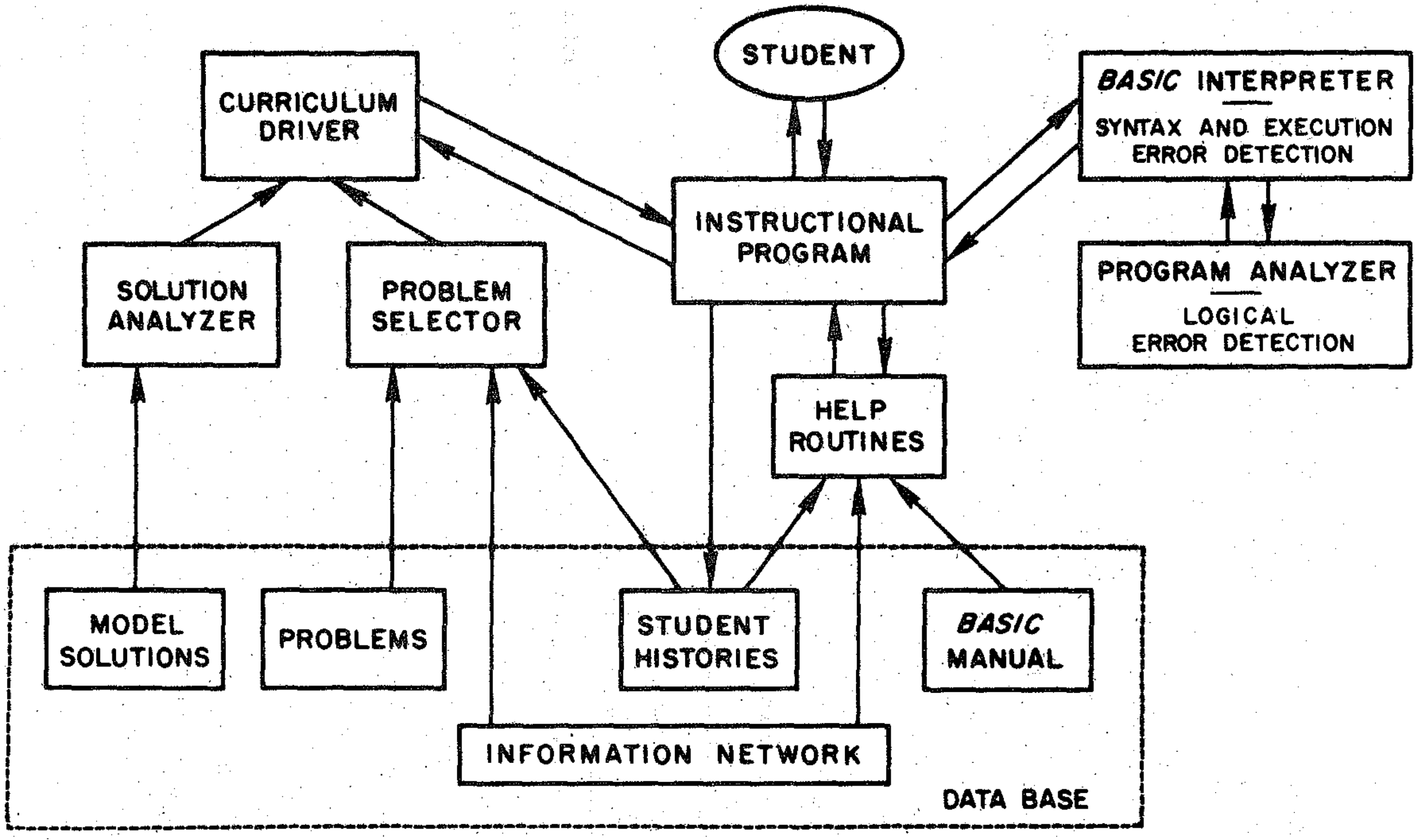

Figure 1. BIP's Information Flow Diagram

BIP offers the student considerable flexibility in making his own task-related decisions. He may ask for hints and subtasks to help him get started in solving the given problem, or he may ponder the problem on his own, using only the manual for additional information. He may request a different task by name, in the event that he wishes to work on it immediately, either completing the new task or not, as he chooses. On his return, BIP tells him the name of the again-current task, and allows him to have its text printed to remind him of the problem he is to solve. The student may request the model solution for any task at any time, but BIP will not print the model for the current task unless the student has exhausted the available hints and subtasks. Taken together, the curriculum options allow for a wide range of student preferences and behaviors.

BIP's Information Network

Task selection, remedial assistance, and problem area determination require that the program have a flexible information store interrelating tasks, hints, manual references, etc. This store has been built using the associative language LEAP, a SAIL sub-language, in which set, list and ordered triple data structures are available (Feldman, Low, Swinehart, & Taylor, 1972; Swinehart & Sproull, 1971; VanLehn, 1973). Figure 2 presents a simplified relationship among a few programming concepts, specific observable skills that characterize the acquisition of the concepts, and programming problems that require the use of those skills. The network is constructed using the associative triple structure, and is best described in terms of the various types of nodes:

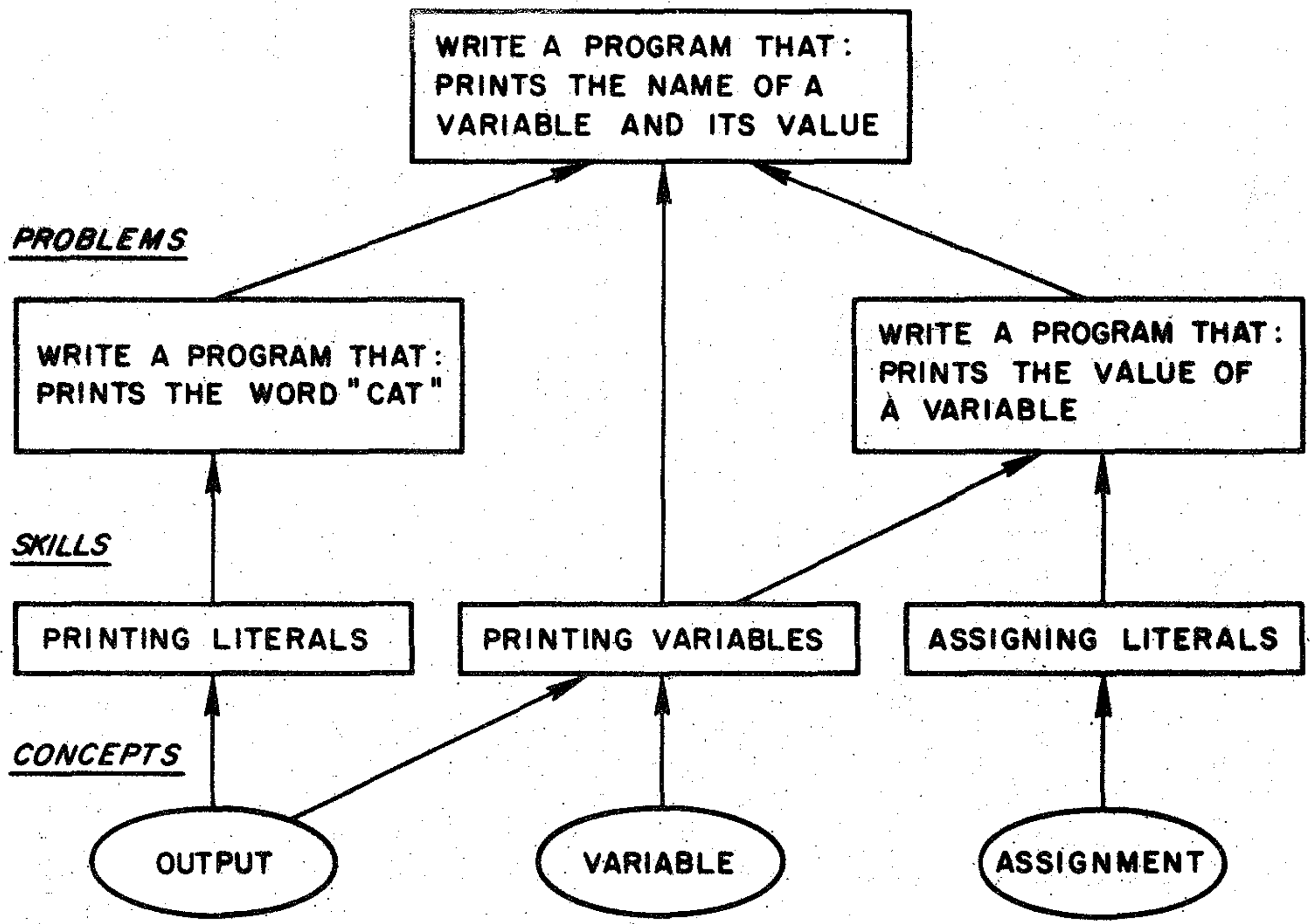

Figure 2. A Segment of BIP's Information Network

TASKS — All curriculum elements exist as task nodes in the network. They are linked to each other as subtasks, prerequisite tasks, or "must follow" tasks.

SKILLS — The skill nodes are intermediaries between the concept nodes and the task nodes (Figure 2). Skills are very specific, e.g., "concatenating string variables" or "incrementing a counter variable." By evaluating success on the individual skills, the program estimates competence levels in the concept areas. In the network, skills are related to the tasks that require them and to the concepts that embody them.

CONCEPTS — The principal concept areas covered by BIP are the following: interactive programs; variables and literals; expressions; input and output; program control - branching; repetition - loops; debugging; subroutines; and arrays.

OPERATORS — Each BASIC operation (PRINT, LET, ...) is a node in the network. The operations are linked to the tasks in two ways: either as elements that must be used in the solution of the problem, or as those that must not be used in the solution.

HINTS — The hint nodes are linked to the tasks for which they may be helpful. Each time a new skill, concept or BASIC operator is introduced, there is an extra hint that gives a suitable manual reference.

ERRORS — All discoverable syntax, structural, and execution errors exist as nodes in the network, linked to the relevant "help" messages, manual references and remedial lessons.

Clearly, in some cases, a hierarchy among skills or problems is implicit; more frequently, however, such a relationship cannot be assumed. By imposing only a very loose hierarchy (e.g., requiring that all students begin the course with the same problem), it is possible to select curriculum and provide assistance on the basis of a student's demonstrated competence level on specific skills, rather than on the basis of a predetermined, nonindividualized, sequence of problems. Students who acquire competence in skills in some manner other than that assumed by subject-matter experts to be standard should benefit most from this potential for individualization.

Upon completion of a task, the student is given a "post task interview" in which BIP presents the model solution stored for that problem. The student is encouraged to regard the model as only one of many possible solutions. BIP asks the student whether he has solved the problem, then asks (for each of the skills associated with the task) whether he needs more practice involving that skill. In addition to the information gained from this student self-analysis, BIP also stores the result of a comparison between the student's program and the model solution, based on the output of both programs when run on a set of test data. The student's responses to the interview and the results of the program comparison are used in future BIP-generated curriculum decisions. BIP informs the student that he has completed the task, and either allows him to select his next task by name (from an off-line printed list of names and problem texts), or selects it for him.

An example of the role of the Information Network in BIP's tutorial capabilities is the BIP-generated curriculum decisions mentioned above. By storing the student's own evaluation of his skills, and by comparing his solution attempts to the stored models, BIP can be said to "learn" about each student as an individual who has attained a certain level of competence in the skills associated with each task. For example, BIP might have recorded the fact that a given student had demonstrated competence (and confidence) in the skill of assigning a literal value to a variable (e.g., N = 1), but had failed to master the skill of incrementing a counter variable (e.g., N = N+1). BIP can then search the network to locate the skills that are appropriate to each student's abilities and present tasks that incorporate those skills. The network provides the base from which BIP can generate decisions that take into account both the subject matter and the student, behaving somewhat like a human tutor in presenting material that either corrects specific weaknesses or challenges and extends particular strengths, proceeding into as yet unencountered areas.

The BIP program has been running successfully with both junior college and university students. However, the program is still very much in an experimental stage. From a psychological viewpoint, the principal research issues deal with (1) procedures for obtaining on-line estimates of student abilities as represented in the information network, and (2) alternative methods for using the current estimates in the information network to make instructional decisions. Neither of these issues is restricted to this particular course, and a major goal in the development of BIP is to provide an instructional model suitable to a variety of different subject areas. Two topics must be discussed in relation to this goal: the nature of appropriate subject areas and the general characteristics of the BIP-like structure that makes it particularly useful in teaching such subjects.

A subject well-suited to this approach generally fits the following description: it has clearly definable, demonstrable skills, whose relationships are well-known; the real content of the subject matter is of a problem-solving, rather than a fact-acquiring, nature; the problems presented to the student involve overlapping sets of skills; and a student's solution to a given problem can be judged as adequate or inadequate with some degree of confidence. The BASIC language, as taught by BIP, is one such subject, but the range of appropriate curriculums goes well beyond the area of computer science. For example, elementary statistics could be taught by a similar approach, as could algebra, navigation, accounting, or organic chemistry. All these subject areas involve the manipulation of information by the student toward a known goal, all involve processes that can be carried out or simulated by a computer, and all are based on a body of skills whose acquisition by the student can be measured with an acceptable degree of accuracy.

Because they require the development of problem-solving skills, rather than the memorization of facts, these subject areas are frequently difficult to master and difficult to tutor, especially using standard CAI techniques. One limitation of such standard techniques is their dependence on a "right" answer to a given question or problem, which precludes active student participation in a problem-solving process consisting of many steps, none of which can be evaluated as correct or incorrect except within the context of the solution as a whole. In addition, standard CAI techniques usually consist of an instructional facility alone--a mechanism by which information is presented and responses are judged. This facility can be linked to a true problem-solving facility that allows the student to proceed through the steps to a solution, but the link does not allow the transfer of information between the instructional and the problem-solving portions of the program. The complete integration of the two parts is a key feature of BIP, making it appropriate to instruction in subject areas that have been inadequately treated in CAI.

The most general characteristics of the "network" structure include a representation of the curriculum in terms of the specific skills required in its mastery and a representation of the student's current levels of competence in each of the skills he has been required to use. Individual record-keeping relates each student's progress to the curriculum at all times, and any number of schemes may be used to apply that relationship to the selection of tasks or the presentation of additional information, hints, advice, etc.

An important element of our network structure is the absence of an established path through the curriculum, providing the built-in flexibility (like that of a human tutor) to respond to individual students' strengths and weaknesses as each student works with the course. This can only be accomplished through a careful analysis and precise specification of the skills inherent in the subject matter, the construction of a thorough curriculum providing in-depth experience with all the skills, and a structure of associations among elements of the curriculum that allows for the implementation of various instructional strategies. Instructional flexibility is complemented by research flexibility in such a structure, because the nature of the associations can be modified for different experimental purposes. Once the elements of the network have been established, it is easy, for example, to change the prerequisite relationship between two problems, or to specify a higher level of competence in a given skill as a criterion measure.

The considerable complexity involved in programming this kind of flexible structure imposes a certain limitation. Standard CAI "author languages" are not appropriate to this network approach, and constructing a CAI course on BIP's pattern is not a task to be undertaken by the educator (or researcher) who has no programming support. The usefulness of author languages is their simplicity, which allows subject-matter experts to prepare course material relatively quickly and easily. Most author languages provide for alternative paths through a curriculum, for alternative answer-matching schemes, and so forth; considerable complexity is certainly possible. However, the limits, once reached, are real, and the author simply cannot expand the sophistication of his course beyond those limits.

The programming support required by the network approach, on the other hand, implies (1) the use of a general, powerful language allowing access to all the capabilities of the computer itself, and (2) a programming group with the training and experience to make full use of the machine. It has been our experience that the flexibility of a general purpose language, while expensive in a number of ways, is worth the costs by virtue of the much greater freedom it allows in the construction of the curriculum and the implementation of experimental conditions. For a more complete description of BIP and a review of our plans for further research see Barr, Beard, and Atkinson (1974).

INSTRUCTION IN INITIAL READING (GRADES 1-3)

Our first efforts to teach reading under computer control were aimed at a total curriculum that would be virtually independent of the classroom teacher (Atkinson, 1968). These early efforts proved reasonably successful, but it soon became apparent that the cost of such a program would be prohibitive if applied on a large-scale basis. Further, it was demonstrated that some aspects of instruction could be done very effectively using a computer, but that there were other tasks for which the computer did not have any advantages over classroom teaching. Thus, during the last four years, our orientation has changed and the goal now is to develop low-cost CAI that supplements classroom teaching and concentrates on those tasks in which individualization is critically important. A student terminal in the current program consists only of a Model-33 teletypewriter with an audio headset. There is no graphic or photographic capability at the student terminal as there was in our first system, and the character set of the teletypewriter includes only uppercase letters. On the other hand, the audio system is extremely flexible and provides virtually instantaneous access to any one of 6,000 recorded words and messages.

Reading Curriculum

Reading instruction can be divided into two areas which have been referred to as "decoding" and "communication." Decoding is the rapid, if not automatic, association of phonemes or phoneme groups with their respective graphic representations. Communication involves reading for meaning, aesthetic enjoyment, emphasis, and the like. Our CAI program provides instruction in both types of tasks, but focuses primarily on decoding. The program is divided into eight parts or strands. As indicated in Figure 3, entry into a strand is determined by the student's level of achievement in the other strands. Instruction begins in Strand 0, which teaches the skills required to interact with the program. Entry into the other strands is dependent on the student's performance in earlier strands. For example, the letter identification strand starts with a subset of letters used in the earliest sight words. When a student reaches a point in the letter identification strand where he has

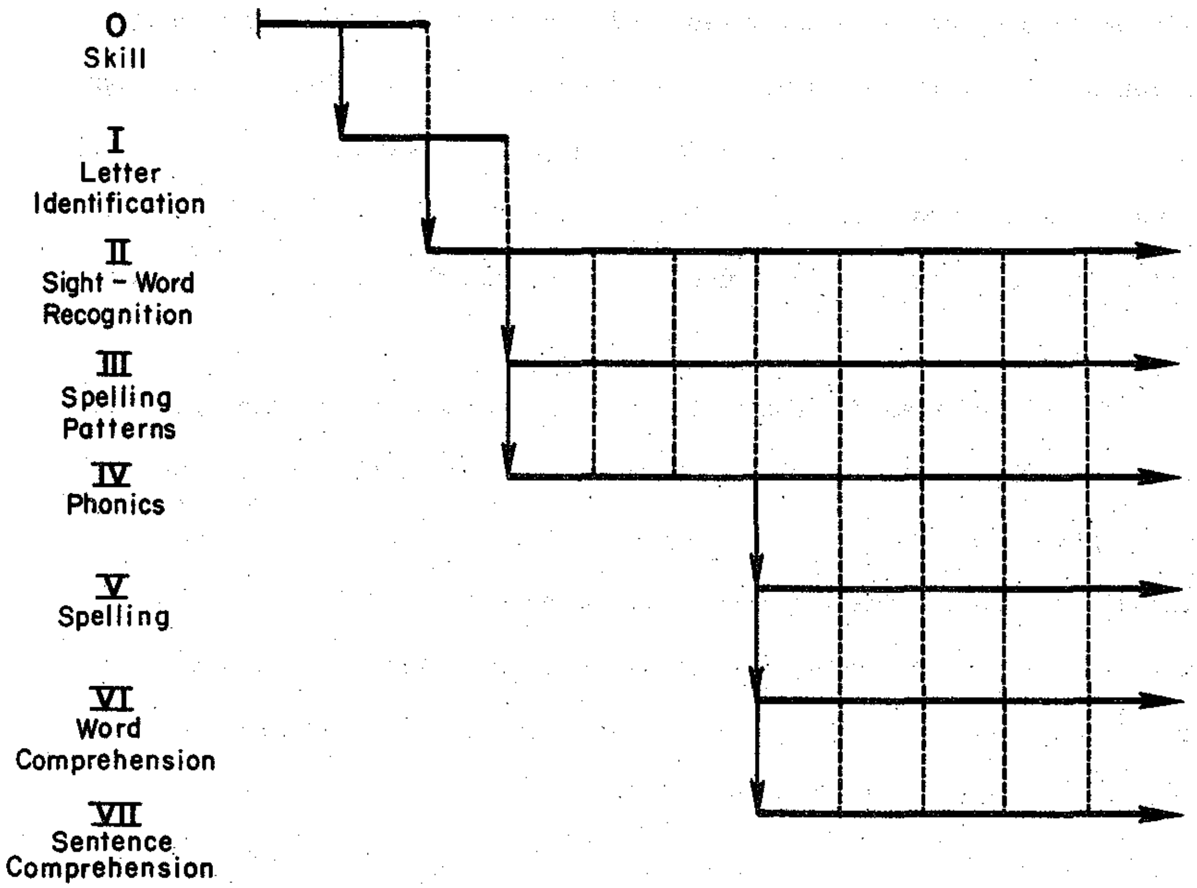

Figure 3. Schematic presentation of the strand structure. (Entry into each strand depends on a student's performance in earlier strands. The vertical dotted lines represent maximal rate contours which control the student's progress in each strand relative to the other strands.)

exhibited mastery over the letters used in the first words of the sight-word strand, he enters that strand. Similarly, entry into the spelling pattern strand and the phonics strand is controlled by the student's placement in the sight-word strand. On any given day, a student may be seeing exercises drawn from as many as five strands. The dotted vertical lines in Figure 3 represent "maximal rate contours," which control the student's progress in each strand relative to his progress in other strands. The rationale underlying these contours is that learning particular material in one strand facilitates learning in another strand; thus, the contours are constructed so that the student learns specific items from one strand in conjunction with specific items from other strands.

The CAI program is highly individualized so that a trace through the curriculum is unique for each student. Our problem is to specify how a given subject's response history should be used to make instructional decisions. The approach that we have adopted is to develop mathematical models for the acquisition of the various skills in the curriculum, and then use these models to specify optimal sequencing schemes. Basically, this approach is what has come to be known in the engineering literature as "optimal control theory," or, more simply, "control theory." Precisely the same problems are posed in the area of instruction, except that the system to be controlled is the human learner rather than a machine or group of industries. If a learning model can be specified, then methods of control theory can be used to derive optimal instructional strategies.

Some of the optimization procedures will be reviewed later, but in order for the reader to have some idea of how the CAI program operates, let me first describe a few of the simpler exercises used in Strands II, III, and IV. Strand II provides for the development of a sight-word vocabulary. Vocabulary items are presented in five exercise formats; only the copy exercise and the recognition exercise will be described here. The top panel of Table 1 illustrates the copy exercise, and the lower panel illustrates the recognition exercise. Note that when a student makes an error, the system responds with an audio message and prints out the correct response. In earlier versions of the program, the student was required to copy the correct response following an error. Experiments demonstrated that the overt correction procedure was not particularly effective; simply displaying the correct word following an error provided more useful feedback.

Strand III offers practice with spelling patterns and emphasizes the regular grapheme-phoneme correspondences that exist in English. Table 2 illustrates exercises from this strand. For the exercise in the top panel of Table 2, the student is presented with three words involving the same spelling pattern and is required to select the correct one based on its initial letters. Once the student has learned to use the initial letter or letter sequence to distinguish between words, he moves to the recall exercise illustrated in the bottom panel of Table 2. Here he works with a group of words, all involving the same spelling pattern. On each trial the audio system requests a word that requires adding an initial consonant or consonant cluster to the spelling pattern mastered in the preceding exercise. Whenever a student makes a

Table 1

Examples of Two Exercises Used in Strand II
(Sight-Word Recognition)

	Teletypewriter display	Audio message
Copy exercise		
The program outputs:	PEN	(Type pen.)
The student responds by typing:	PEN	
The program outputs:	+	(Great!)
The program outputs:	EGG	(Type egg.)
The student responds by typing:	EFF	
The program outputs:	////EGG	(No, egg.)
Recognition exercise		
The program outputs:	PEN NET EGG	(Type pen.)
The student responds by typing:	PEN	
The program outputs:	+	
The program outputs:	PEN EGG NET	(Type net.)
The student responds by typing:	NET	
The program outputs:	+	(Fabulous!)

Note: The top panel displays the copy exercise and the bottom panel the recognition exercise. Rows in the table correspond to successive lines on the teletypewriter printout.

Table 2

Examples of the Recognition and Recall Exercises Used in Strand III (Spelling Patterns)

	Teletypewriter display	Audio message
Recognition exercise		
The program outputs:	KEPT SLEPT CREPT	(Type kept.)
The student responds by typing:	KEPT	
The program outputs:	+	
Recall exercise		
The program outputs:		(Type crept.)
The student responds by typing:	CREPT	
The program outputs:	+	(That's fabulous!)

correct response, a "+" sign is printed on the teletypewriter. In addition, every so often the program will give an audio feedback message; these messages vary from simple ones like "great," "that's fabulous," "you're doing brilliantly," to some that have cheering, clapping, or bells ringing in the background. These messages are not generated at random, but depend on the student's performance on that particular day.

When the student has mastered a specified number of words in the sight-word strand, he begins exercises in the phonics strand; this strand concentrates on initial and final consonants and consonant clusters in combination with medial vowels. As in most linguistically oriented curricula, students are not required to rehearse or identify consonant sounds in isolation. The emphasis is on patterns of vowels and consonants that bear regular correspondences to phonemes. The phonic strand is the most complicated one of the group and involves eight exercise formats; two of the formats will be described here. The upper panel of Table 3 illustrates an exercise in which the student is required to identify the graphic representation of phonemes occurring at the end of words. Each trial begins with an audio presentation of a word that includes the phonemes, and the student is asked to identify the graphic representation. After mastering this exercise, he is transferred to the exercise illustrated in the bottom panel of Table 3. The same phonemes are presented, but now the student is required to construct words by adding appropriate consonants.

Optimal Sequences for Individual Students

This has been a brief overview of some of the exercises used in the curriculum; a more detailed account of the program can be found in

Table 3

Examples of Two Exercises from Strand IV (Phonics)

	Teletypewriter display	Audio message
Recognition exercise		
The program outputs:	-IN -IT -IG	(Type /IG/ as in fig.)
The student responds by typing:	IG	
The program outputs:	+	(Good!)
The program outputs:	-IT -IN -IG	(Type /IT/ as in fit.)
The student responds by typing:	IT	
The program outputs:	+	
Build-a-word exercise		
The program outputs:	-IN -IT -IG P--	(Type pin.)
The student responds by typing:	PIN	
The program outputs:	+	(Great!)
The program outputs:	-IG -IN -IT F--	(Type fig.)
The student responds by typing:	FIN	
The program outputs:	////FIG	(No, we wanted fig.)

Atkinson, Fletcher, Lindsay, Campbell, and Barr (1973). The key to the curriculum is the optimization schemes that control the sequencing of the exercises; these schemes can be classified at three levels. One level involves decision making within each strand. The problem is to decide which items to present for study, which exercise formats to present them in, and when to schedule review. A complete response history exists for each student, and this history is used to make trial-by-trial decisions regarding what to present next. The second level of optimization deals with decisions about allocation of instructional time among strands for a given student. At the end of an instructional session, the student will have reached a certain point in each strand and a decision must be made about the time to be allocated to each strand in the next session. The third level of optimization deals with the distribution of instructional time among students. The question here is to allocate computer time among students to achieve instructional objectives that are defined not for the individual student but for the class as a whole. In some global sense, these three levels of optimization should be integrated into a unified program. However, we have been satisfied to work with each separately, hoping that later they can be incorporated into a single package.

Optimization within a strand (what has been called Level 1) can be illustrated using the sight-word strand. The strand comprises a list of about 1,000 words; the words are ordered in terms of their frequency in the student's vocabulary, and words at the beginning of the list have highly regular grapheme-phoneme correspondences. At any point in time a student will be working on a limited pool of words from the master list; the size of this working pool depends on the student's ability level and is usually between 5 and 10 words. When one of these words is mastered, it is deleted from the pool and replaced by the next word on the list or by a word due for review. Figure 4 presents a flow chart for the strand. Each word in the working pool is in one of five possible instructional states. A trial involves sampling a word from the working pool and presenting it in an appropriate exercise format. The student is pretested on a word the first few times it is presented to eliminate words already known. If he knows the word, he will pass the pretest and the word will be dropped from the working pool. If the student does not pass the pretest, he first studies the word using the recognition exercise. If review is required, he studies the word again in what is designated in Figure 4 as Exercises 4 and 5.

As indicated in Figure 4, a given word passes from one state to the next when it reaches criterion. And this presents the crux of the optimization problem, which is to define an appropriate criterion for each exercise. This has been done using simple mathematical models to describe the acquisition process for each exercise and the transfer functions that hold between exercises (Atkinson & Paulson, 1972). These models are simple Markov processes that provide reasonably accurate accounts of performance on our tasks. Parameters of the models are defined as functions of two factors: (1) the ability of the particular student, and (2) the difficulty of the particular word. An estimate of the student's ability is obtained by analyzing his response record on all previous words, and an estimate of a word's difficulty is obtained by analyzing performance on that particular word for all students on the

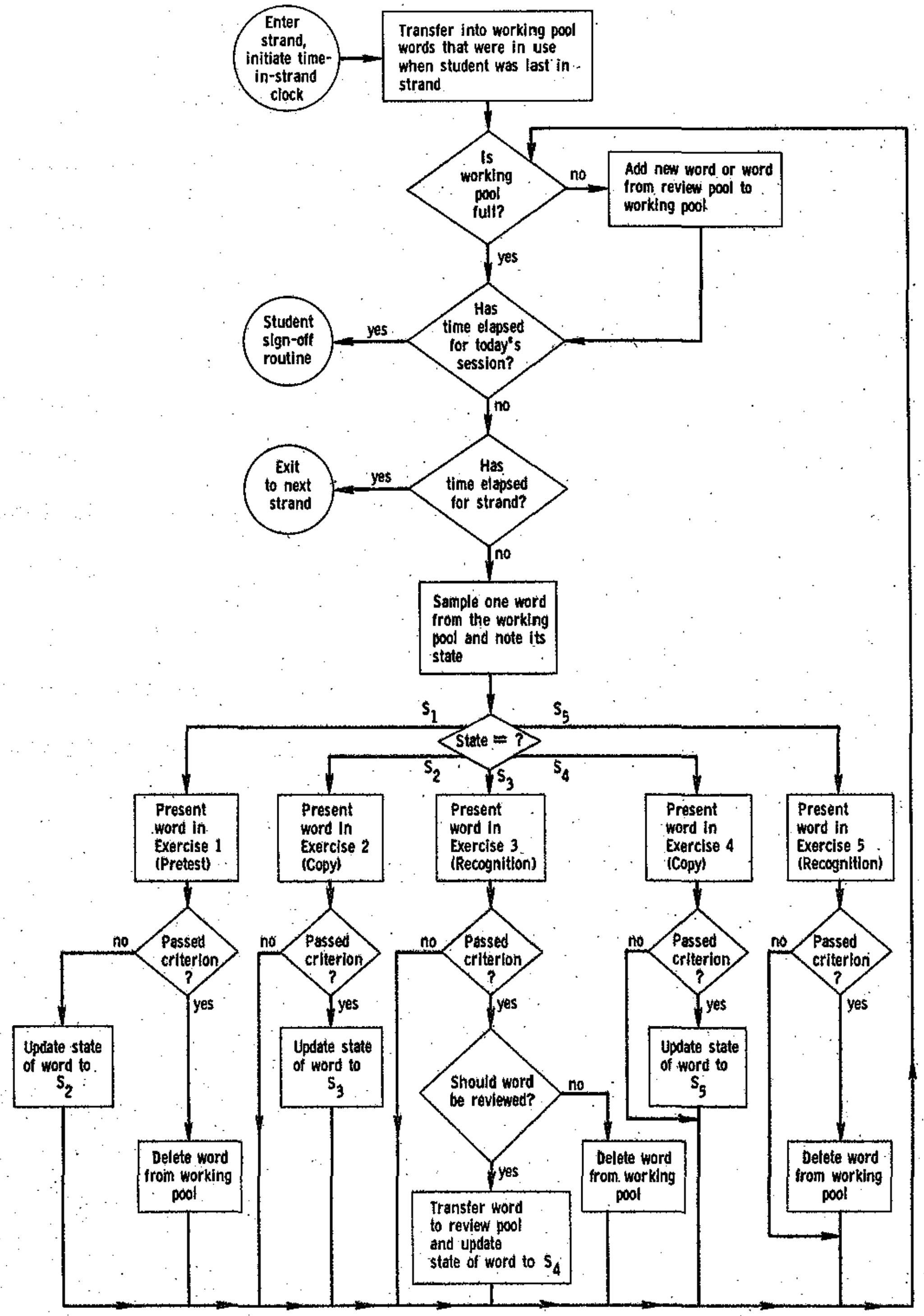

Figure 4. Partial flow chart for Strand II (sight-word recognition). The various decisions represented in the bottom part of the chart are based on fairly complicated computations that make use of the student's response history. The same recognition exercise is used in both state S_3 and S_5.

program. The student records are continually updated by the computer and are used to compute a maximum likelihood estimate of each student's ability factor and each word's difficulty factor. Given a well-defined model and estimates of its parameters, we can use the methods of control theory to define an optimal criterion for each exercise. The criterion will vary depending on the difficulty of the item, the student's ability level, and the precise sequence of correct and incorrect responses made by the student to the item. It is important to realize that the optimization scheme is not a simple branching program based on the student's last response, but depends in a complicated way on his complete response history.

Optimization between strands (what has been called Level II) was mentioned earlier in the description of maximum-rate contours. In some respects this optimization program is the most interesting of the group, but it cannot be explained without going into considerable mathematical detail. In essence, a learning model is developed that specifies the learning rate on each strand as a function of the amount of material that has been mastered in each of the other strands. Using mathematical methods of control theory, an optimal instructional strategy is determined based on the model. This strategy defines a closed-loop feedback controller that specifies daily instructional allocations for each strand based on the best current estimate of how much the student has mastered in each strand. An account of the theoretical rationale for the program is presented in Chant and Atkinson (1973).

Optimizing Class Performance

Next let us consider an example of optimization at what has been called Level III. The effectiveness of the CAI program can be increased by optimally allocating instructional time among students. Suppose that a school has budgeted a fixed amount of time for CAI and must decide how to allocate that time among a class of first-grade students. For this example, maximizing the effectiveness of the CAI program will be interpreted as meaning that we want to maximize the class performance on a standardized reading test administered at the end of the first grade.

On the basis of prior studies, the following equation has been developed to predict performance on a standardized reading test as a function of the time a student spends on the CAI system:

$$P(t;i) = A(i) - B(i)\exp[-tC(i)] \text{ .}$$

The equation predicts Student i's performance on a standardized test as a function of the time, t, spent on the CAI system during the school year. The parameters $A(i)$, $B(i)$, and $C(i)$ characterize Student i, and vary from one student to another. These parameters can be estimated from scores on reading readiness tests and from the student's performance during his first hour of CAI. After estimates of these parameters have been made, the above equation can be used to predict end-of-year test scores as a function of the CAI time allocated to that student.

Let us suppose that a school has budgeted a fixed amount of time T on the CAI system for a first-grade class of N students; further, suppose that students have had reading readiness tests and a preliminary run on the CAI system so that estimates of the parameters A, B, and C have been made for each student. The problem then is to allocate time T among the

N students so as to optimize learning. In order to do this, it is first necessary to have a model of the learning process. Although the above equation does not offer a very detailed account of learning, it suffices as a model for purposes of this problem. This is an important point to keep in mind; the nature of the specific optimization problem determines the level of complexity that needs to be represented in the learning model. For some optimization problems, the model must provide a relatively detailed account of learning to specify a viable strategy, but for other problems a simple descriptive equation may suffice.

In addition to a model of the learning process, we must also specify an instructional objective. Only three possible objectives will be considered here:

I. Maximize the mean value of P over the class of students.

II. Minimize the variance of P over the class of students.

III. Maximize the mean value of P under the constraint that the resulting variance of P is less than or equal to the variance that would be obtained if no CAI were administered.

Objective I maximizes the gain for the class as a whole; Objective II reduces differences among students by making the class as homogeneous as possible; and Objective III attempts to maximize the class performance while insuring that differences among students are not amplified by CAI. If we select Objective I as the instructional objective, then the problem of deriving an optimal strategy reduces to maximizing the function

$$f[t(1),t(2),\ldots,t(N)] = \sum_{i} \{A(i)-B(i)\exp[-t(i)C(i)]\}$$

$$t(1) + t(2) + \ldots + t(N) = T$$

where $t(i)$ is the time allocated to Student i. This maximization can be done using the methods of dynamic programming. To illustrate the approach, computations were made for a first-grade class for which the parameters A, B, and C had been estimated for each student. Employing these estimates, computations were carried out to determine the time allocations that maximized the above equation. For the optimal policy, the predicted mean performance level of the class on the end-of-year tests was 14% higher than a policy that allocated time equally among students (i.e., an equal-time policy where $t(i) = T/N$ for all i). This gain represents a substantial improvement; the drawback is that the class variance is roughly 15% greater than the variance for the class using an equal-time policy. This means that if we are only interested in raising the class average, we will have to give the rapid learners substantially more time on the CAI system and let them progress far beyond the slow learners.

Although a time allocation that complies with Objective I does increase overall class performance, other objectives need to be considered. For comparison, time allocations also were computed for Objectives II and III. Table 4 presents the predicted gain in average class performance as a percentage of the mean value for the equal-time policy. Objective II yielded a negative gain in the mean; and so it should, since its goal was to minimize variability, which is accomplished by reducing the time allocations for rapid learners and giving more attention to the slower ones. The reduction in variability for Objective II is 12%. Objective III, which strikes a balance between Objective I and Objective II, yields an 8% gain in mean performance yet reduces variability by 6%.

Table 4

Predicted Percent Gain in the Mean of P and in the Variance of P When Compared with the Mean and Variance of the Equal-Time Policy

	Instructional objective		
	I	II	III
% gain in mean of P	14	-15	8
% gain in variance of P	15	-12	-6

In view of these results, Objective III would be preferred by most educators and laymen. It offers a substantial increase in average performance while maintaining a low level of variability. These computations make it clear that the selection of an instructional objective should not be done in isolation but should involve a comparative analysis of several objectives, taking into account more than one dimension of performance. Even if the principal goal is to maximize the class average, it is inappropriate in most educational situations to select Objective I over III if it is only slightly better for the class average, while permitting variability to mushroom.[2]

Effectiveness of the Reading Program

Several evaluation studies of the reading program have been conducted in the last few years. Rather than review these here, I would prefer to describe one in some detail (Fletcher & Atkinson, 1972). In this particular study, 50 pairs of kindergarten students were matched on a number of variables, including sex and readiness scores. At the start of the first grade, one member of each pair was assigned to the experimental group and the other to the control group. Students in the experimental group received CAI, but only during the first grade; students in the control group received no CAI. The CAI lasted approximately 15 minutes per day;[3] during this period the control group studied reading

---

[2]For a more detailed discussion of some of the issues involved in selecting objective functions see Jamison, Fletcher, Suppes, and Atkinson (1975).

[3]In this study no attempt was made to allocate time optimally among students in the experimental group; rather, an equal-time policy was employed.

in the classroom. Except for this 15-minute period, the school day for the CAI group was like that of the control group. Standardized tests were administered at the end of the first grade and again at the end of the second grade. All the tests showed roughly the same pattern of results; to summarize the findings, only data from the California Cooperative Primary Reading Test will be described. At the end of the first grade, the experimental group showed a 5.05-month gain over the control group. The groups, when tested a year later (with no intervening CAI treatment), showed a difference of 4.90 months. Thus, the initial difference observed following one year of CAI was maintained, although not amplified, during the second year when no CAI was administered to either group.

No definitive conclusions can be drawn from evaluation studies of this sort about the specific contributions of CAI versus other aspects of the situation. Obviously the curriculum materials used in the CAI program are important, as well as other factors. To do the type of study that would isolate the important variables is too large an undertaking to be worthwhile at this juncture in the development of the reading program. Thus, to some extent it is a matter of judgment in deciding which variables account for the differences observed in the above study. In my view, individualizing instruction is the key factor in successfully teaching reading. This does not mean that all phases of instruction should be individualized, but certain skills can be mastered only if instruction is sensitive to the student's particular difficulties. A reading teacher interacting on a one-to-one basis with a student may be more effective than our CAI program. However, when working with a group of children (even as few as four or five), it is unlikely that she can match the computer's effectiveness in making instructional decisions over an extended period of time.

SECOND-LANGUAGE VOCABULARY LEARNING

In this section, research on CAI programs for second-language vocabulary learning will be discussed. As noted elsewhere in this paper, the principal goal of our research on computerized instruction has been to develop adaptive teaching procedures--procedures that make moment-by-moment decisions about which instructional action should be taken next based on the student's unique response history. To help guide the theoretical aspect of this work, some years ago we initiated a series of experiments on the very restricted but well-defined problem of optimizing the teaching of a foreign-language vocabulary. This is an area where mathematical models provide an accurate description of learning, and these models can be used in conjunction with the methods of control theory to derive precise algorithms for sequencing instruction among vocabulary items. Although our original interest in this topic was primarily theoretical, the work has proved to have significant practical applications. These applications involve computerized vocabulary learning programs designed to supplement college-level courses in second-language instruction. A particularly interesting effort involves a supplementary Russian program in use at Stanford University. Students are exposed to approximately 1,000 words per academic quarter using the computer; in conjunction with normal classroom work this program enables them to develop a substantial vocabulary.[4] Many foreign language instructors believe that the major obstacle to successful instruction in a second language is not learning the grammar of the language, but rather in acquiring a sufficient vocabulary so that the student can engage in meaningful conversations and read materials other than the textbook.

In examining the work on vocabulary acquisition I will not describe the CAI programs, but will review some research on optimal sequencing schemes that provides the theoretical rationale for the programs. It will be useful to describe one experiment in some detail before considering more general issues.

An Experiment on Optimal Sequencing Schemes

In this study a large set of German-English items are to be learned during an instructional session that involves a series of trials. On each trial, one of the German words is presented and the student attempts to give the English translation; the correct translation is then presented for a brief study period. A predetermined number of trials is allocated for the instructional session, and after some intervening period a test is administered over the entire vocabulary. The problem is to specify a strategy for presenting items during the instructional session so that performance on the delayed test will be maximized.

---

[4] These CAI vocabulary programs make use of optimal sequencing schemes of the sort to be discussed in this section, as well as certain mnemonic aids. For a discussion of these mnemonic aids see Raugh and Atkinson (1975) and Atkinson and Raugh (1975).

Four strategies for sequencing the instructional material will be considered. One strategy, designed RO for random order, is to cycle through the set of items randomly; this strategy is not expected to be particularly effective, but it provides a benchmark against which to evaluate other procedures. A second strategy, designated SS for self selection, is to let the student determine for himself how best to sequence the material. In this mode, the student decides on each trial which item is to be presented; the learner rather than an external controller determines the sequence of instruction.

The third and fourth schemes are based on a decision-theoretic analysis of the task. A mathematical model that provides an accurate account of vocabulary acquisition is assumed to hold in the present situation. The model is used to compute, on a trial-by-trial basis, an individual student's current state of learning. Based on these computations, items are selected for test and study so as to optimize the level of learning achieved at the termination of the instructional session. Two optimization strategies derived from this type of analysis will be examined. In one case, the computations for determining an optimal strategy are carried out assuming that all vocabulary items are of equal difficulty; this strategy is designated OE (i.e., optimal under the assumption of equal item difficulty). In the other case, the computations take into account variations in difficulty level among items; this strategy is called OU (i.e., optimal under the assumption of unequal item difficulty). The details of these two strategies will be described later.

The experiment was carried out under computer control; the details of the experimental procedure are given in Atkinson (1972b). The students participated in two sessions: an "instructional session" of approximately two hours and a briefer "delayed-test session" administered one week later. The delayed test was the same for all students and involved a test over the entire vocabulary. The instructional session was more complicated. The vocabulary items were divided into seven lists, each containing 12 German words; the seven lists were arranged in a round-robin order. On each trial of the instructional session a list was displayed on a projection screen, and the student inspected it for a brief period of time; the list involved only the 12 German words and not their English translations. Then one of the items on the list was selected for test and study. In the RO, OE, and OU conditions the item was selected by the computer; in the SS condition the item was chosen by the student. After an item was selected for test, the student attempted to provide a translation by typing it on his computer console; then feedback regarding the correct translation was given. The next trial began with the computer displaying the next list in the round robin, and the same procedure was repeated. The instructional session continued in this fashion for 336 trials.

The results of the experiment are summarized in Figure 5. Data are presented on the left side of the figure for performance on successive blocks of trials during the instructional session; on the right are results from the test session administered one week after the instructional session. The data from the instructional session are presented in successive blocks of 84 trials; for the RO condition this means that

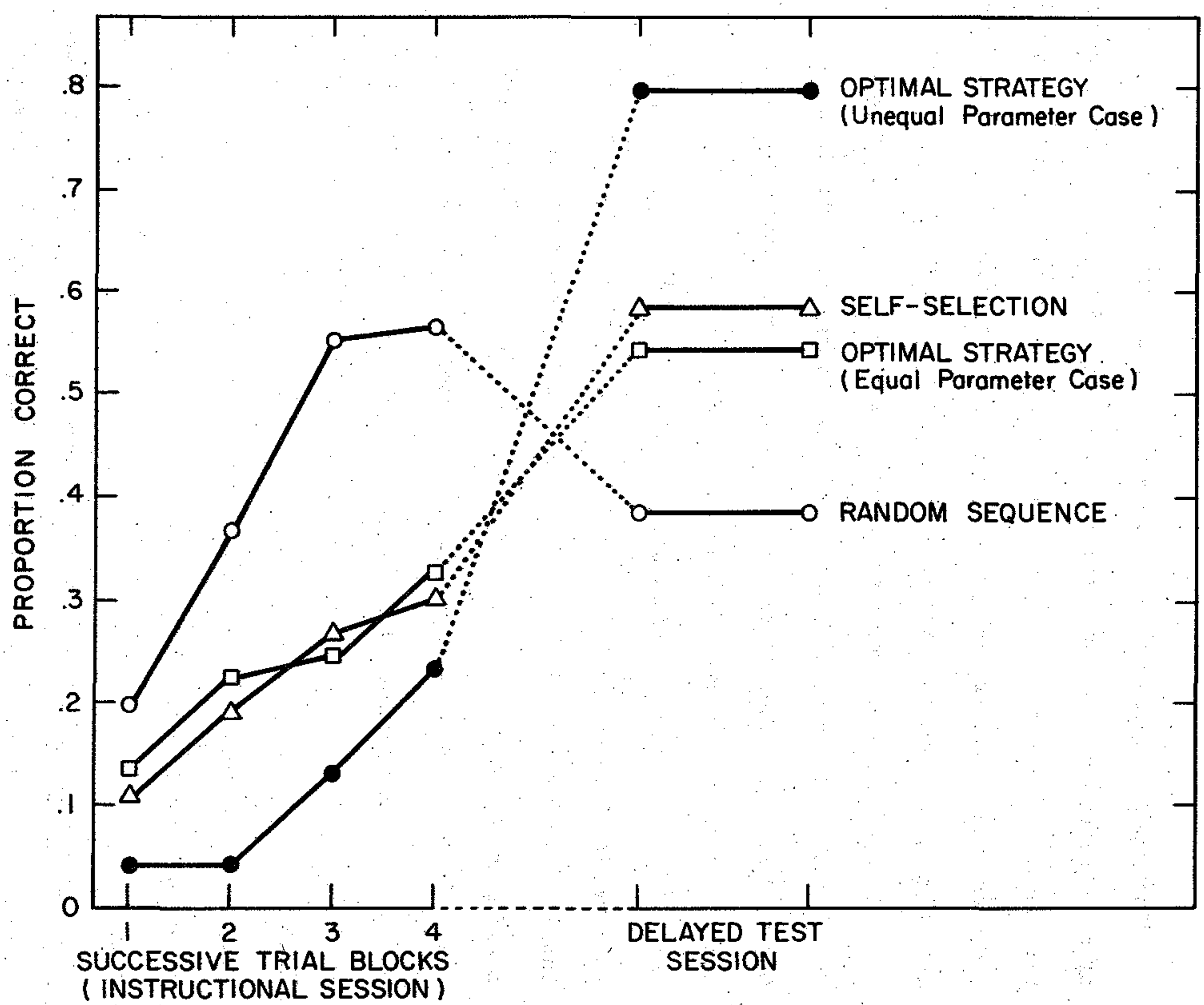

Figure 5. Proportion of correct responses in successive trial blocks during the instructional session and on the delayed test administered one week later.

on the average each item was presented once in each of these blocks. Note that performance during the instructional session is best for the RO condition, next best for the OE condition which is slightly better than the SS condition, and poorest for the OU condition. The order of the groups is reversed on the delayed test. (Two points are displayed in the figure for the delayed test to indicate that the test involved two random cycles through the entire vocabulary; however, the values given are the average over the two test cycles.) The OU condition is best with a correct response probability of .79; the SS condition is next with .58; the OE condition follows closely at .54 and the RO condition is poorest at .38. The observed pattern of results is what one would expect. In the SS condition, the students are trying to test themselves on items they do not know; consequently, during the instructional session, they should have a lower proportion of correct responses than students run on the RO procedure where items are tested at random. Similarly, the OE and OU conditions involve a procedure that attempts to identify and test those items that have not yet been mastered and should produce high error rates during the instructional session. The ordering of groups on the delayed test is reversed since all words are tested in a non-selective fashion; under these conditions the proportion of correct responses provides a measure of a student's true mastery of the total set of vocabulary items.

The magnitude of the effects observed on the delayed test are of practical significance. The SS condition (when compared to the RO condition) leads to a relative gain of 53%, whereas the OU condition yields a relative gain of 108%. It is interesting that students were somewhat effective in determining an optimal study sequence, but not so effective as the best of the two adaptive teaching systems.

Rationale for Sequencing Schemes

Both the OU and OE schemes assume that vocabulary learning can be described by a fairly simple model. We postulate that a given item is in one of three states (P, T, and U) at any moment in time. If the item is in State P, then its translation is known and this knowledge is "relatively" permanent in the sense that the learning of other items will not interfere with it. If the item is in State T, then it is also known but on a "temporary" basis; in State T the learning of other items can give rise to interference effects that cause the item to be forgotten. In State U the item is not known, and the student is unable to give a translation.

When Item i is presented on a trial during the instructional session, the following transition matrix describes the possible chance in its state:

$$
L(i) = \begin{array}{c c} & \begin{array}{ccc} P & T & U \end{array} \\ \begin{array}{c} P \\ T \\ U \end{array} & \left[\begin{array}{ccc} 1 & 0 & 0 \\ x(i) & 1-x(i) & 0 \\ y(i) & z(i) & 1-y(i)-z(i) \end{array} \right] \end{array} .
$$

Rows of the matrix represent the state of the item at the start of the trial, and columns the state at the end of the trial. On a trial when some item other than Item i is presented for test and study, transitions in the state of Item i also may take place. Such transitions can occur only if the student makes an error to the other item; in that case the transition matrix applied to Item i is as follows:

$$
\begin{array}{cc}
 & \begin{array}{ccc} P & T & U \end{array} \\
F(i) = \begin{array}{c} P \\ T \\ U \end{array} & \begin{bmatrix} 1 & 0 & 0 \\ & 1-f(i) & f(i) \\ 0 & 0 & 1 \end{bmatrix} .
\end{array}
$$

Basically, the idea is that when some other item is presented that the student does not know, forgetting may occur for Item i if it is in State T.

To summarize, when Item i is presented for test and study, transition Matrix L(i) is applied; when some other item is presented that elicits an error, Matrix F(i) is applied. It is also assumed that at the start of the instructional session Item i is either in State P, with probability g(i), or in State U, with probability 1-g(i); the student either knows the translation without having studied the item or does not. The above assumptions provide a complete description of the learning process. The parameter vector [x(i), y(i), z(i), f(i), g(i)] characterizes the learning of Item i in the vocabulary set. The first three parameters govern the acquisition process; the next parameter, forgetting; and the last, the student's knowledge prior to entering the experiment.

We now turn to a discussion of how the OE and OU procedures were derived from the model. Prior to conducting the experiment reported here, a pilot study was run using the same word lists and the RO procedure described above. Data from the pilot study were employed to estimate the parameters of the model; the estimates were obtained using the minimum chi-square procedures described in Atkinson (1972b). Two separate estimates of parameters were made. In one case it was assumed that the items were all equally difficult, and data from all 84 items were lumped together to obtain a single estimate of the parameter vector; this estimation procedure will be called the equal-parameter case (E case). In the second case the data were separated by items, and an estimate of the parameter vector was made for each of the 84 items; this procedure will be called the unequal-parameter case (U case). The two sets of parameter estimates were then used to generate the optimization schemes previously referred to as the OE and OU procedures.

In order to formulate an instructional strategy, it is necessary to be precise about the quantity to be maximized. For the present experiment the goal is to maximize the total number of items the student correctly translates on the delayed test.[5] To do this, we need to specify the relationship between the state of learning at the end of the instructional session and performance on the delayed test. The assumption made here is that only those items in State P at the end of the instructional session will be translated correctly on the delayed test; an item in State T is presumed to be forgotten during the intervening week. Thus, the problem of maximizing delayed-test performance involves maximizing the number of items in State P at the end of the instructional session.

---

[5] Other measures can be used to assess the benefits of an instructional strategy; e.g., in this case weights could be assigned to items measuring their relative importance. Also costs may be associated with the various actions taken during an instructional session. Thus, for the general case, the optimization problem involves assessing costs and benefits and finding a strategy that maximizes an appropriate function defined on them. For a discussion of these points see Dear, Silberman, Estavan, and Atkinson (1967), and Smallwood (1962, 1971).

Having numerical values for parameters and knowing a student's response history, it is possible to estimate his current state of learning.[6] Stated more precisely, the learning model can be used to derive equations and, in turn, compute the probabilities of being in States P, T, and U for each item at the start of any trial, conditionalized on the student's response history up to that trial. Given numerical estimates of these probabilities, a strategy for optimizing performance is to select that item for presentation that has the greatest probability of moving into State P. This strategy has been termed the one-stage optimization procedure because it looks ahead one trial in making decisions. The true optimal policy (i.e., an N-stage procedure) would consider all possible item-response sequences for the remaining trials and select the next item so as to maximize the number of items in State P at the termination of the instructional session. Unfortunately, for the present case the N-stage policy cannot be applied because the computations are too time consuming even for a large computer. Monte Carlo studies indicate that the one-stage policy is a good approximation to the optimal strategy; it was for this reason, as well as the relative ease

---

[6] The student's "response history" is a record for each trial of the vocabulary item presented and the response that occurred. It can be shown that there exists a "sufficient history" that contains only the information necessary to estimate the student's current state of learning; the sufficient history is a function of the complete history and the assumed learning model (Groen & Atkinson, 1966). For the model considered in this paper the sufficient history is fairly simple. It is specified in terms of individual vocabulary items for each student; we need to know the ordered sequence of correct and incorrect responses to a given item plus the number of errors (to other items) that intervene between each presentation of the item.

of computing, that the one-stage procedure was employed. For a discussion of one-stage and N-stage policies and Monte Carlo studies comparing them see Groen and Atkinson (1966), Calfee (1970), and Laubsch (1970).

The optimization procedure described above was implemented on the computer and permitted decisions to be made for each student on a trial-by-trial basis. For students in the OE group, the computations were carried out using the five parameter values estimated under the assumption of homogeneous items (E case); for students in the OU group the computations were based on the 420 parameter values estimated under the assumption of heterogeneous items (U case).

The OU procedure is sensitive to interitem differences and consequently generates a more effective optimization strategy than the OE procedure. The OE procedure, however, is almost as effective as having the student make his own instructional decisions and far superior to a random presentation scheme.

The study reported here is one in a series of experiments dealing with optimal sequencing schemes. It was selected because it is easily described and permits direct comparison between a learner-controlled procedure versus procedures based on a decision-theoretic analysis. For a review of other studies similar to the one reported above see Chiang (1974), Delaney (1974), Laubsch (1970), Kimball (1973), Paulson (1973), and Atkinson and Paulson (1972). Some of these studies examine procedures that are more powerful than the ones described here, but they are complicated and difficult to describe without going into mathematical detail. The major improvements involve two factors: (1) methods for estimating the model's parameters during the course of instruction, and

(2) more sophisticated ways of interpreting the model's parameters to take account of both differences among students and differences among items. For example, let P(i,j) be a generic symbol for a parameter vector characterizing student i learning vocabulary item j. In these studies P(i,j) is specified as a function of a vector A(i) measuring the ability of student i and a vector D(j) measuring the difficulty of item j. The problem then is to estimate the ability level of each student and the difficulty of each item while the student is running on the program. In a study reported in Atkinson and Paulson (1972), rather dramatic results were obtained using such a procedure. A special feature of the study was that students were run in successive groups, each starting after the prior group had completed the experiment. As would be expected, the overall gains increased from one group to the next. The reason is that for the first group of students the estimates of item difficulty, D(j), were crude but improved with the accumulation of data from each successive wave of students. Near the end of the study estimates of D(j) were quite precise and were essentially constants in the system. The only task that remained when a new student came on the system was to estimate A(i); that is, the parameters characterizing his particular ability level. This study provides an example of an adaptive instructional system that meets both of the requirements stated earlier in this paper. The sequencing of instruction varies as a function of each student's history record, and over time the system improved in efficiency by using data from previous students to sharpen its estimates of the difficulty of instructional materials.

CONCLUDING REMARKS

The projects described in this paper have one theme in common, namely, developing computer-controlled procedures for optimizing the instructional process. For several of the instructional tasks considered here, mathematical models of the learning process were formulated which made it possible to use formal methods in deriving optimal policies. In other cases the "optimal schemes" were not optimal in a well-defined sense, but were based on our intuitions about learning and some relevant experiments. In a sense, the diversity represented in these examples corresponds to the state of the art in the field of instructional design. For some tasks we can use psychological theory to help define optimal procedures; for others our intuitions, modified by experiments, must guide the effort. Hopefully, our understanding of these matters will increase as more projects are undertaken to develop sophisticated instructional procedures.

Some have argued that any attempt to devise optimal strategies is doomed to failure, and that the learner himself is the best judge of appropriate instructional actions. I am not sympathetic to a learner-controlled approach to instruction, because I believe its advocates are trying to avoid the difficult but challenging task of developing a viable theory of instruction. There obviously is a place for the learner's judgments in making instructional decisions; for example, such judgments play an important role in several parts of our BIP course. However, using the learner's judgment as one of several items of information in making instructional decisions is different from proposing that the learner should have complete control. Results presented in this paper and those cited in Beard, Lorton, Searle, and Atkinson (1973) indicate that the learner is not a particularly effective decision maker in guiding the learning process.

Elsewhere I have defined the criteria that must be satisfied before an optimal instructional procedure can be derived using formal methods (Atkinson, 1972a). Roughly stated, they require that the following elements of an instructional situation be clearly specified:

(1) The set of admissible instructional actions

(2) The instructional objectives

(3) A measurement scale that permits costs to be assigned to each of the instructional actions and payoffs to the achievement of instructional objectives

(4) A model of the learning process

If these four elements can be given a precise interpretation, then it is usually possible to derive an optimal instructional policy. The solution for an optimal policy is not guaranteed, but in recent years powerful tools have been developed for discovering optimal, or near optimal, procedures if they exist. I will not discuss these four elements here except to note that the first three can usually be specified with a fair degree of consensus. Issues of short-term versus long-term assessments of costs and payoffs raise important questions regarding educational policy, but at least for the types of instructional situations examined in this paper reasonable specifications can be offered for the first three elements. However, the fourth element--the specification of a model of the learning process--represents a major obstacle. Our theoretical understanding of learning is so limited that only in very special cases can a model be specified in enough detail to enable the derivation of optimal procedures. Until we have a much deeper understanding of the learning process, the identification of truly effective strategies will not be possible. However, an all-inclusive theory of learning is not a prerequisite for the development of optimal procedures. What is needed is a model that captures the essential features of that part of the learning process being tapped by a given instructional task. Even models that have been rejected on the basis of laboratory investigations may be useful in deriving instructional strategies. Several of the learning models considered in this paper have proven unsatisfactory when tested in the laboratory and evaluated using standard goodness-of-fit criteria; nevertheless, the optimal strategies they generate are often quite effective. My own preference is to formulate as complete a learning model as intuition and data will permit and then use that model to investigate optimal procedures. When possible the learning model should be represented in the form of mathematical equations, but otherwise as a set of statements in a computer-simulation program. The main point is that the development of a theory of instruction cannot progress if one holds the view that a comprehensive theory of learning is a prerequisite. Rather, advances in learning theory will affect the development of a theory of instruction, and conversely the development of a theory of instruction will influence the direction of research on learning.

REFERENCES

Anderson, J. R., & Bower, G. H. Human associative memory. Washington, D.C.: Winston, 1973.

Atkinson, R. C. Computerized instruction and the learning process. American Psychologist, 1968, 23, 225-239.

Atkinson, R. C. Ingredients for a theory of instruction. American Psychologist, 1972a, 27, 921-931.

Atkinson, R. C. Optimizing the learning of a second-language vocabulary. Journal of Experimental Psychology, 1972b, 96, 124-129.

Atkinson, R. C., Fletcher, J. D., Lindsay, E. J., Campbell, J. O., & Barr, A. Computer-assisted instruction in initial reading. Educational Technology, 1973, 13, 27-37.

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

Atkinson, R. C., & Raugh, M. R. An application of the mnemonic keyword method to the acquisition of a Russian vocabulary. Journal of Experimental Psychology, 1975, in press.

Barr, A., Beard, M., & Atkinson, R. C. A rationale and description of the BASIC instructional program. Technical Report No. 228. Stanford, Calif.: Institute for Mathematical Studies in the Social Sciences, Stanford University, 1974.

Beard, M. H., Lorton, P., Jr., Searle, B. W., & Atkinson, R. C. Comparison of student performance and attitude under three lesson selection strategies in computer-assisted instruction. Technical Report No. 222. Stanford, Calif.: Institute for Mathematical Studies in the Social Sciences, Stanford University, 1973.

Calfee, R. C. The role of mathematical models in optimizing instruction. Scientia: Revue Internationale de Synthese Scientifique, 1970, 105, 1-25.

Carbonell, J. R. AI in CAI: An artificial intelligence approach to computer-assisted instruction. IEEE Transactions on Man-Machine Systems, 1970, MMS-11, 190-202.

Chant, V. G., & Atkinson, R. C. Optimal allocation of instructional effort to interrelated learning strands. Journal of Mathematical Psychology, 1973, 10, 1-25.

Chiang, A. Instructional algorithms derived from mathematical models: An application in computer assisted instruction of Chinese vocabulary. Unpublished Ph.D. thesis, City University of New York, 1974.

Collins, A. M., Carbonell, J. R., & Warnock, E. H. Analysis and synthesis of tutorial dialogues. Technical Report No. 2631. Cambridge, Mass.: Bolt, Beranek and Newman, 1973.

Dear, R. E., Silberman, H. F., Estavan, D. P., & Atkinson, R. C. An optimal strategy for the presentation of paired-associate items. Behavioral Science, 1967, 12, 1-13.

Delaney, H. D. Individualizing instruction: An extension of the RTI model and its optimal presentation strategy. Unpublished M.A. thesis, University of North Carolina, 1973.

Feldman, J. A., Low, J. R., Swinehart, D. C., & Taylor, R. H. Recent developments in SAIL. AFIPS Fall Joint Computer Conference, 1972, 1193-1202.

Fletcher, J. D., & Atkinson, R. C. Evolution of the Stanford CAI program in initial reading. Journal of Educational Psychology, 1972, 63, 597-602.

Groen, G. J., & Atkinson, R. C. Models for optimizing the learning process. Psychological Bulletin, 1966, 66, 309-320.

Jamison, D., Fletcher, J. D., Suppes, P., & Atkinson, R. C. Cost and performance of computer-assisted instruction for compensatory education. In R. Radner and J. Froomkin (eds.), Education as an industry. Columbia University Press, 1975.

Kimball, R. B. Self-optimizing computer-assisted tutoring: Theory and practice. Technical Report No. 206. Stanford, Calif.: Institute for Mathematical Studies in the Social Sciences, Stanford University, 1973.

Laubsch, J. H. Optimal item allocation in computer-assisted instruction. IAG Journal, 1970, 3, 295-311.

Newell, A., & Simon, H. A. Human problem solving. Englewood Cliffs, N.J.: Prentice-Hall, 1972.

Paulson, J. A. An evaluation of instructional strategies in a simple learning situation. Unpublished Ph.D. thesis, Stanford University, 1973.

Raugh, M. R., & Atkinson, R. C. A mnemonic method for the learning of a second-language vocabulary. Journal of Educational Psychology, 1975, in press.

Rumelhart, D. E., Lindsay, P. H., & Norman, D. A. A process model for long-term memory. In E. Tulving and W. Donaldson (eds.), Organization and memory. New York: Academic Press, 1972.

Schank, R. C. Conceptual dependency: A theory of natural-language understanding. Cognitive Psychology, 1972, 3, 552-631.

Smallwood, R. D. A decision structure for teaching machines. Cambridge, Mass.: M.I.T. Press, 1962.

Smallwood, R. D. The analysis of economic teaching strategies for a simple learning model. Journal of Mathematical Psychology, 1971, 8, 285-301.

Swinehart, D. C., & Sproull, R. F. SAIL. Stanford, Calif.: Stanford Artificial Intelligence Laboratory Operating Note 57.2, Stanford University, 1971.

VanLehn, K. SAIL user manual. Stanford, Calif.: Stanford Artificial Intelligence Laboratory, Stanford University, 1973.