APPLICATION OF A

KNOWLEDGE COMPILATION MODELOF INSTRUCTION TO EMBEDDED TRAINING

Thomas F. Carolan
University of Central Florida • Institute for Simulation and Training • Orlando, Florida

Kent E. Williams
Virginia Polytechnic Institute and State University • Management Systems Laboratories • Blacksburg, Virginia

Nina Chatham
ORAU Postgraduate Research • Participation Program • Naval Training Systems Center • Orlando, Florida

ABSTRACT

Considerable research has been directed at understanding the processes involved in acquiring and using knowledge and skills. One focus of this cognitive research is the application of formal models of learning and knowledge representation to the design of computer based instruction. Advances in adaptive instruction and intelligent tutoring have been driven by implementing explicit models of the knowledge to be learned, and the strategies used to communicate that knowledge. Two recent experiments, employing Navy personnel, have demonstrated the effectiveness of using a formal approach to instruction in an embedded training environment.

The design of the instructional material began with a fine grain analysis to determine the knowledge to be learned and to develop the basic structures upon which the adaptive processes operate. There is evidence that curriculum content which is generated from the results of an explicit cognitive analysis process enhances learning. In the first experiment the effectiveness of using a cognitive analysis process to structure the information and an adaptive process to sequence the information was evaluated for domain naive students. In the second experiment the effectiveness of the knowledge compilation approach was evaluated using Navy instructors. The results of this research are discussed in the context of the application of current cognitive learning research to embedded training.

INTRODUCTION

During the past decade, the cognitive psychology research effort has advanced the development of instructional theory through the application of cognitive models of competent performance to the development of more precise instructional methods and strategies (Glaser & Bassok, 1989). Over this same time period, the study of intelligent tutoring systems (ITS) has emerged as an interdisciplinary research field (Johnson, 1991) in which artificial intelligence techniques are applied to implement these cognitive-based methods and strategies as instructional system programs and architectures. An emphasis on the implementation of more explicit instructional methods and strategies should also benefit more traditional computer based training systems.

The military services have moved to provide effective embedded training (ET) through building training capabilities into or adding them onto operational systems. Williams and Reynolds (1989), in a review of existing ET systems, proposed that the implementation of cognitive learning principles and intelligent tutoring strategies would improve the instructional technology component of ET and lead to considerable training gain over conventional computer-based instruction.

What types of instructional strategies will be most effective for embedded training environments? On-board embedded training (ET) provides console operators with a high fidelity environment in which to practice, refresh, and refine the highly perishable cognitive and motor skills needed to skillfully interact with complex systems. Also, ET provides the opportunity to gain the new knowledge and skills required for qualification on advanced watch stations. The objectives of an ET session is to build on existing knowledge, to diagnose and correct deficiencies as efficiently as possible, and to allow for consolidation of skills through practice. The training effectiveness of ET sessions will depend upon instructional technologies and strategies that promote efficient acquisition and retention of skills and knowledge.

Candidate instructional technologies were selected to be implemented, and experimental evaluations of the impact of these technologies on the acquisition and retention of skills and knowledge were initiated. Two of these experimental evaluations were conducted on the Navy's Lesson Translator (L-TRAN) embedded training system for the general purpose tactical consoles supporting Navy Tactical Data System (NTDS) sensor and weapons operations. The first experiment (Williams et al., 1989) evaluated, for entry level console operators, the impact on training gain of structuring the lesson content based on a cognitive analysis process, and of adapting the sequence of training to individual performance. Both strategies were found to have an effect on learning with the domain naive students. A second experiment extended that effectiveness evaluation to more experienced users, and identified the distribution of practice throughout the lesson as a contributing factor of the increased effectiveness. This paper summarizes the research, discusses the application of a cognitive analysis process to the design and sequencing of instructional exercises, and interprets the research results within a knowledge compilation framework (Anderson, 1983).

PROCEDURAL LEARNING AND INSTRUCTION

Tactical console operations can be characterized as involving the skillful application of specific perceptual, cognitive, and motor operations to the goal-directed processing of complex information patterns. Developing competence at organizing appropriate sequences of operations in response to particular conditional contexts can be understood as a process of acquiring and proceduralizing knowledge. Given this perspective, it was determined that an appropriate instructional strategy for embedded training of console operations could be based on applying a cognitive skill acquisition approach to the problem. One assumption regarding the implementation of this approach is that the essential information and operational procedures required for performing a task, such as those involved in operating a tactical console, constitute the elements of an ideal model of competent performance (Anderson, Boyle, Corbett & Lewis 1990). For the operator, learning to perform a task can also be interpreted as an active knowledge construction process through which a representation, or cognitive model, of the essential information and required procedures is developed. This model then serves as a basis for the operator's future interaction with the device.

Generating Rules

There is considerable research evidence to support a characterization of human processing as inclined towards encoding patterns of information in a manner consistent with the generation of rules. People are very good at abstracting rules from examples, when those examples involve concrete events to which they can relate (e.g., Cheng, et. al., 1986). One difference between experts' and novices' knowledge is that experts tend to functionally organize information in cause and effect relations that are linked to the conditions under which it can be used (Glaser & Bassok, 1989). The utility of coding knowledge in terms of rules is further illustrated by examples from instructional research. For example, Chi, Bassok, Lewis, Reimann and Glaser (1989) demonstrated that good students are able to explain the conditions and consequences associated with a specific action, whereas poor students are not. Good students also generate more complete condition-action rules during the learning process. They construct such rules from the instructional material presented to them (Bovair & Kieras, 1990).

The production system framework (Newell & Simon, 1972; Anderson, 1983) for modeling cognitive processes provides an explicit formalization of cognitive skill development. Models implemented as production systems form the basis of many current intelligent tutoring systems. Briefly, a production system consists of a set or network of production rules or condition-action pairs of the IF-THEN form, a context or environment, and a control strategy. The condition side of the production rule specifies task goals and subgoals, along with specific states of the environment. The conditions are the inputs to the rule. If all of the conditions of a rule are matched, then the rule is triggered and the action side is executed. Essentially, production system models attempt to establish links between cognition and the production of actions.

Knowledge of procedures can be represented by sets of production rules. These sets of productions are directed towards achieving goals and incorporate goals in their structure, starting with a high level goal and decomposing it into subgoals. The goal structure, in turn, assists in specifying an appropriate set of production rules. The production rules can be reorganized depending on the goal structure of the task. Anderson (1986) has noted that by establishing a goal tree, one can predict what rules are likely to be composed by students.

From this perspective, the console operator's knowledge and skill is structured as a production system model of the instructional information. The goal of training is to provide an environment for students to acquire or refresh these production rules (Anderson, Conrad, & Corbett, 1989) which represent the tasks to be performed and the information necessary for their performance.

Knowledge Compilation

The learning process through which procedural knowledge is acquired has been referred to as knowledge compilation (Anderson, 1983, 1986). An overview of this approach with its implications for instruction provides a basis for understanding the acquisition of console operation skills. Knowledge is thought to be encoded as declarative structures through language comprehension and perceptual processes. However, in order to convert this knowledge of facts (declarative knowledge) into knowledge of how to correctly use those facts to produce actions in a particular situation (procedural knowledge), the declarative knowledge must be applied and interpreted in the context in which it is to be used.

Declarative knowledge is typically applied in a trial and error fashion resulting in the generation of more efficient and task specific production rules. Performance becomes more goal directed. As practice continues, the knowledge becomes more procedural, the need to refer to declarative knowledge is reduced, and the application of the rules becomes more automatic. This transition from using declarative knowledge to using procedural knowledge has been called proceduralization (Anderson, 1983). Through knowledge compilation, high level rules that link key information and action components of the situation in a more efficient manner are produced. Small steps are "composed" into single units which can be accessed independently and applied when appropriate. In addition to composing rules into larger 'chunks' and proceduralizing successful rules, new information about specific conditions and actions may be added to declarative memory. With continued practice, more productions can be formed based on these additional declarative structures. Anderson (1986) and others, have found that experts have difficulty recalling procedures which they used early in their training. As experts acquire knowledge, old procedures are composed to form higher level condition-action segments. At some point, it may be difficult for experts to retrieve the lower-level procedures from which the higher-level procedures were formed.

In addition to the compilation process, empirical evidence exists to support a strengthening process (Anderson, 1982, Anderson, et al., 1990). As declarative and procedural knowledge are used, they are applied more efficiently, allowing resources to be used to process new information. Anderson reports evidence which demonstrates that encoding time is reduced from first to second application of a production. This is interpreted as the compilation of knowledge into procedural form followed by a gradual reduction in encoding time as the production rules become strengthened. From a knowledge compilation perspective, learning involves a process of acquiring new declarative knowledge using existing productions, applying this declarative knowledge to new situations, compiling task-specific productions, and strengthening both declarative and procedural knowledge.

A number of investigators (Anderson, et al. 1990; Carroll, 1990) have identified a problem with technical instructional material. Typically, some of the information the trainee needs for task performance is omitted. The student must then infer the missing information or discover it through trial and error. Also, information that is not useful for either understanding or performance in the task domain is presented. The student then uses time and cognitive resources attempting to interpret this information in the context of the task environment (Kieras, 1987a). When instruction is not precise and to the point, the cost in terms of time and cognitive resources increases, as does the potential to negatively affect performance through incorrect interpretation (Anderson & Jeffries, 1985). A number of researchers (Kieras, 1987a; Reder, Charney & Morgan, 1986) have demonstrated the effectiveness of concise and well focused textual instruction.

The cognitive modeling perspective provides a framework for determining the content of instruction. A cognitive model, or in Anderson's terminology, an ideal student model, provides a cognitive level representation of training requirements for achieving competent performance. One instructional strategy is to embed this ideal model into the instructional system and use it to diagnose student errors and generate instruction. Evaluations of these intelligent tutoring systems indicate that they can achieve better results than standard classroom instruction (Anderson et al. 1990). Anderson also reports that instructional materials designed to communicate the information in the ideal model can be more effective than standard texts even without an adaptive tutoring component.

The explicit cognitive model framework has generated a number of instructional principles that provide guidance with reference to designing and presenting material to the trainee in a way that will make learning more efficient. For example, learning will progress more efficiently when the individual can explicitly observe the rules to be learned. Therefore, by keeping the language concise and focused on an explicit encoding of the facts, relations, and rules, the process of interpreting declarative knowledge can be made more effective. In addition, graphical representations that make logical and spatial relationships explicit will reduce the inference requirements and facilitate interpretation (Larkin & Simon, 1987). When the focus of declarative instruction is skill development, learning can be enhanced by presenting material in a format appropriate for mapping onto the goal and subgoal structure of the task environment. The knowledge representation should specify not only the rules but the functions of these rules, any preconditions that apply, and the anticipated consequences.

The knowledge compilation model makes certain assumptions about how the student's knowledge changes as the student progresses through a learning program. These assumptions can be used to guide a strategy for presenting the components of instruction. For example, the components of a rule should be explicitly identified and coupled to allow for more efficient composition of rules. This will minimize the working memory load required during the learning process, and ultimately during performance.

Proceduralization can be facilitated by providing interactive practice opportunities and feedback to assure that trainees have correctly interpreted the declarative instructions. The knowledge and skill objectives of the instruction, the structure of the domain to be learned, and the existing skill level of the trainee set the bounds on the appropriate level of composition to be achieved. Within these bounds, rules can be composed into higher level productions or decomposed into substeps. By providing opportunities for practice at both the composed and the substep levels, the trainee, or the system, can adapt instruction and apply practice based on the strength of existing declarative and procedural knowledge.

THE INSTRUCTIONAL METHODOLOGY

Console operators typically monitor or manually perform activities which include: searching for, detecting, and tracking targets with radar/sonar, identifying the target track(s) as friend or foe, evaluating and ranking by threat posed (e.g., platform, proximity, speed, heading), assigning and engaging weapons to counter targets, and assessing the results of engagement. These tasks are complex and require a significant amount of knowledge as well as cognitive and psychomotor skills. Operators must possess the required knowledge so that they do not have to look up information and procedures; they must also possess sufficient speed to operate their console while handling multiple incoming threats.

The development of a production system model of the instructional material is accomplished by conducting a cognitive task analysis (Kieras, 1987a, 1987b, 1988; Williams, Reynolds, & Carolan, 1990). The cognitive analysis methodology shares many of the principles and processes of the GOMS model of Card, Moran and Newell (1983). The purpose of the cognitive task analysis is to determine what specific knowledge has to be learned through the training and to develop representations of that knowledge which are consistent with those produced by mechanisms of human learning, such as production memory and its associated cognitive processes. The technique of generating production units for instructional systems has been applied to mathematics, (Anderson, 1981; Reif, 1989), programming, (Anderson, Conrad & Corbett, 1989), and tactical console operations (Williams, et al., 1989).

Cognitive Task Analysis

Consistent with the approach discussed by Kieras (1987a, 1987b), the initial analysis focused on the development of a hierarchy of task goals from the information required to effectively operate a tactical console. This analysis involved

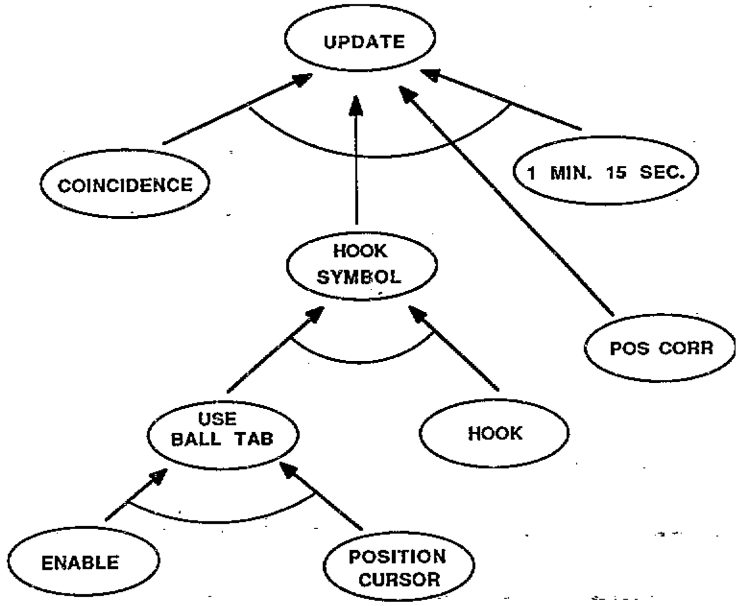

Figure 1. A partial graph of a method to accomplish the UPDATE goal.

the specification of high level goals, and the identification of the tasks that could be performed on the device which directly related to the attainment of those goals. For console operations, this analysis provided information at different levels: (1) a description of the physical layout of the console with locations of controls and outputs; (2) a description of the various functions of the console; (3) the external console operating procedures; (4) the functional relationships between console parameters and task parameters; and (5) how to use the console parameters to perform the relevant tasks.

A device description hierarchy (see Williams, Reynolds, Carolan, 1989; 1990 for detail) consisted of the information that would be used to explain or describe the relationships among the input and output components of the console. This hierarchy included things like action buttons and display components. A goal or function hierarchy consisted of the information that explains what tasks the console is to be used for. It is used for accomplishing such tasks as acquiring a new track on a potential target, updating information on a firm track, and identifying characteristics of the target. Each element of the function hierarchy is a goal that the user or trainee must accomplish. Achieving each goal requires using elements of the device description hierarchy to perform specific tasks. By linking the elements of the device description hierarchy to the appropriate elements of the function or goal hierarchy, a task-goal hierarchy is developed.

Consistent with the GOMS approach, once the task goals are identified, the methods which make up the task procedures are detailed. Each method specifies a goal or subgoal to be accomplished, as well as the sequence of steps to be executed, and conditions to be met, in order to accomplish that task goal. All responses that the individual requires in performing the to-be-learned tasks are specified and associated with each task goal and subgoal in the hierarchy. All conditions associated with each individual response are identified and linked to that response. All constraints required by console operating procedures are identified and linked to the appropriate responses. Each step of each method consists of an operator or action which is executed. Operators can be perceptual, such as observing the location of a symbol; actual, such as pressing a button; or cognitive, such as making a decision, or storing or retrieving memory information. This detailed cognitive analysis of the domain knowledge then organizes the goals, subgoals, methods, and operators into a graph hierarchy. A portion of the task-goal hierarchy representing tactical console operations in AND/OR graph form is illustrated in Figure 1.

Goals and subgoals can be nested within methods and submethods when specifying the operation of a device, and within a method, other methods may be called by a particular step. If more than one method may be used, the GOMS model requires a selection rule which discriminates between alternative methods, and determines which specific method should be used. For example, an alternative method for hooking a target is to press the sequence button and then the hook button. This method will sequence control to the next target. A selection rule would specify the conditions under which each method is used.

The result of this analysis is the detailed specification of all of the knowledge needed to complete tasks associated with console operations. The specification of methods within the GOMS model essentially identifies a set of production units and rules for relating these productions through the goals and subgoals. The goal and subgoal analysis divides the problem into small enough parts so that the methods or rules can be generated. The specification of these rules is the development of the production system that simulates system functions and operations. Exercises developed as a result of this cognitive analysis process explicitly specify the cognitive model which the trainee must acquire to effectively produce correct responses consistent with the domain knowledge. The goal of the training program is achieved when the trainee works through the exercises and develops an accurate model of the knowledge domain.

Developing Instructional Content

Instructional frames were developed from this hierarchical production system model. When working with existing lessons, the lesson content is restructured based on the production system model developed through the cognitive task analysis. Each individual rule or method in the lesson, as well as each rule that combines relevant facts and/or lower level rules, is composed into an individual exercise. An exercise can consist of information presented on a single screen or frame, or over a sequence of frames. A lesson is made up of a number of exercises each presented over one or more a frames. This breakdown of the lesson into a hierarchy of individual exercises is not inconsistent with the modular structure of the L-TRAN system. Each subject matter module in a lesson is considered to be an independent unit in content and in presentation order. These subject matter modules represent the task goal level of the GOMS hierarchy. Our explicit use of the GOMS methodology separates the material within a subject matter module into instructional units at the methods and at the operator levels. It is these units which we refer to as exercises. Each exercise explicitly describes the subgoal or goal state which triggers the production represented by the frames making up the exercise. In working through the frames the student learns all the declarative facts and how to compose or compile these facts so that when specific cues or subgoals are set, specific actions or sequences of actions are triggered. Each exercise frame created with this methodology is explicitly linked to the conditions, actions, or other rules that make up a production. This methodology is consistent with the principles for lesson design discussed in the L-TRAN style guide, but provides for a more specific design process.

Each exercise consisted of three parts: (1) an exposition of the information specifying a rule; that is, the knowledge to be learned; (2) a problem or example which required a response to a question or the performance of a procedure, and which tests the student's ability to use the knowledge and provides practice in implementing the procedure; and (3) a set of diagnostics that determine the source of any error(s) in terms of the specific conditions or actions of a production for which the student's knowledge is weak.

Exercise Sequence

Advantages of coupling instructional content to a rule-based cognitive model include efficient delivery of exercise content and effective diagnosis of errors relating to specific rules, declarative facts, or combinations of rules. Since each exercise created employing this methodology can be explicitly linked to pieces of a production in the form of conditions, actions, or other rules, the training program can query the trainee to determine what conditions, action, or other rules have been or not been learned. This assessment of student strengths and weaknesses can then help to determine what lesson content should be presented next.

The hierarchical structure of exercises, the specification of goals and subgoals, and the derivation of exercises from underlying productions are all consistent with empirical evidence from cognitive learning experiments. The structuring of the information content of exercises is consistent with the way procedural knowledge is organized. The problem segment of each exercise provides the student with the opportunity to practice (Rosenbloom & Newell, 1986) what is learned from the exposition segment, and thereby use declarative knowledge to develop procedural structures. By using diagnostics to localize strengths and weaknesses, more specific feedback can be provided (Hayes-Roth, Klahr, & Mostow, 1981) to the student and the system can be guided in its selection of the next best exercise for that student.

The capability to select an exercise which overlaps most with what the student knows can be built into an instructional system through an adaptive selection heuristic. The adaptive heuristic facilitates learning by sequencing exercises so that the contents of any two consecutive exercises share as much knowledge as possible. The sequence is based upon what each individual has learned, and therefore the sequence can be different for any two individuals. Upon failure or success on a particular exercise, the system searches for and selects an exercise for presentation which overlaps most with what the student knows and least with what the student does not know. This method exploits strong existing knowledge to build new knowledge. To keep track of the student's progress, the system keeps a record called the student model. The student model records the status of each criterion frame and diagnostic as the lesson progresses. The strength of each piece of knowledge or production rule is determined by the strength of the individual elements which make up that knowledge. The student completes the criterion frame, and if unsuccessful, the system presents the appropriate diagnostics. Then the strength of all rules is updated to reflect the current state of the student's knowledge of the lesson. The heuristic search routine selects the next frame to be presented, based on the information about the current strength of the individual knowledge and rule units which make up the lesson (see Williams, Reynolds, & Carolan, 1990 for detail).

In situations where adaptive exercise sequencing is not possible, the knowledge compilation and strengthening processes suggest guidelines for fixed exercise sequences and branching. When lower levels in the hierarchy of related facts and rules are strengthened initially, with ample opportunity to interact and develop procedural representations, the effect spreads to other related rules. Composite rules, further up in the hierarchy, are also strengthened by virtue of their microunits, thereby facilitating their assimilation.

EXPERIMENT ONE

In the first experiment (Williams et al., 1989), a study was conducted to evaluate the effectiveness of lessons which were restructured based on the cognitive analysis process and sequenced using the adaptive methodology. The domain was tactical console operation procedures. An intelligent computer aided instruction system composed of cognitively developed instructional content and the adaptive instructional strategy was integrated into the Navy's L-TRAN system. A detailed discussion of this experiment was presented at this conference last year (Williams, Reynolds, Carolan, 1990) and will be summarized below.

Forty-eight enlisted Navy students, none of whom had any prior experience with tactical training consoles, were randomly presented one of three versions of a basic lesson on a standard software emulation of an Navy Tactical Decision System (NTDS) console. The content of the L-TRAN lesson was restructured based on the cognitive task analysis process. An adaptive exercise sequencing heuristic was implemented in software, and diagnostics were developed. The result was three versions of the L-TRAN lesson: (1) the original version currently in use, (2) a cognitive version in which the content was structured according to production rules developed from a cognitive task analysis and presented in a fixed sequence, and (3) an adaptive version in which cognitively structured exercises were sequenced, based on the individual's performance.

There was a dramatic impact on training gain when the instructional content of the test lesson was restructured using the cognitive engineering process. Trainees using the cognitively restructured lesson made 65% fewer errors on a final performance test than the control group (See Figure 2). This final performance test required the trainee to interact with the device to recognize the appropriate task goal to be achieved for a given situation and to perform the explicit procedures required for the achievement of that task goal. These results confirmed, for the domain of console operations, that structuring the information to be learned according to guidelines developed from a production system model of knowledge representation can have a profound impact on learning effectiveness.

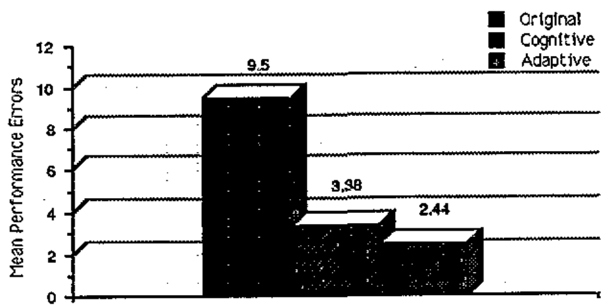

Figure 2. Mean number of errors in each group on the final performance test trial.

In this experiment, performance was further improved when the sequence in which exercises were presented was controlled by an adaptive selection heuristic. After an initial learning trial, the use of this adaptive selection heuristic resulted in increased efficiency of learning; that is, more improvement in learning per unit of learning time. These results indicate that by taking advantage of each individual's prior knowledge and history of performance on the lesson, adaptive frame sequencing is a means for improving the efficiency of ET sessions. To the extent that students with varying degrees of competency are using the ET, the effectiveness and efficiency of adaptive exercise sequencing should be maximized.

The results of this experiment provided convincing evidence that, for naive trainees with little existing domain specific declarative knowledge and virtually no domain specific procedural knowledge, the more formal cognitive approach to instructional design was more effective than a traditional, less formal approach. The target population for embedded training may consist of fleet personnel with entry level to advanced training and experience backgrounds. The trainees using embedded training lessons enter a particular lesson or series of lessons with individual differences in both the declarative knowledge they bring to the lesson and in the device specific procedural knowledge. Will exercises which have been structured based on the cognitive modeling framework make any difference when the trainees are already competent in both the general knowledge domain, and in console operation procedures?

EXPERIMENT TWO

A second experiment (Carolan, Williams, Moskal, 1991) was designed to: 1) transition the previous research on an NTDS console emulator to research on NTDS operational consoles, 2) evaluate the effectiveness of this knowledge engineering methodology with more experienced subjects, and 3) evaluate the impact on learning gain of varying the opportunities for performance feedback.

In the L-TRAN embedded training research described above, fully adaptive sequencing was possible only when NTDS console emulator systems were used. The emulator system allowed for modifications to the control software to implement the student model and sequencing algorithms. In this second experiment, NTDS instructional consoles were used and modification of the NTDS software was not feasible. For this reason, it was not possible to include further evaluation of adaptive exercise sequencing in the second experiment.

An L-TRAN lesson designed to teach the material and procedures required for an advanced watch station was subjected to the cognitive analysis process described in the previous sections and detailed in Williams, Reynolds, and Carolan (1989). The content of the lesson was restructured based on this analysis. Each exercise in the lesson was tightly constrained to match a single rule, or composed rule. Not every rule however, was represented as an exercise in the lesson. Since this was an advanced lesson, the production rules included in the lesson represented middle to upper levels of the skill. Subgoals which involved lower level productions were not always explicitly presented. These include recognition of basic symbology, location of fixed and variable action buttons, composition of low level methods such as hook and ball tab. These productions are prerequisite to the current lesson, and should already be well-learned.

Each exercise generally consisted of two parts: (1) expository information specifying the knowledge to be learned; and (2) a problem or example which required a response or the performance of a procedure. The second part of the exercise required the student to practice using the knowledge and provided an opportunity to implement a proceduralization process and to receive feedback. Three versions of the lesson were compared. These versions differed in the structure of the expository and the practice parts of the lesson. For one version, the expository information was structured to be explicitly related to the sequence of rules developed from the cognitive analysis process described above. The lesson proceeded, for the most part, in a fixed sequence in which expository text explicitly defining a rule is presented and is followed by practice in order to test one's interpretation of the text and to incorporate the rule as a procedure. This sequence of exposition and practice continued at each step in the lesson. For example, referring back to

Figure 1, each node in the graph of UPDATE represents a rule, and each rule is represented in the lesson by an exercise that includes both an expository and practice component. This version of the lesson is consistent with the fixed sequence cognitive version of the previous study (Williams et al. 1989), and is therefore referred to as the cognitive version. A second version of the lesson was developed using the same restructured information as in the cognitive version. In this version, however, the student did not have the opportunity to evaluate one's interpretation of the declarative text information, and receive appropriate feedback, at each of the lower level nodes. The only opportunities for practice occurred at the level of the composite task goals (i.e., the UPDATE node in Figure 1). Hence, we refer to this version of the lesson as the composite practice version. The remaining version is the original L-TRAN lesson in which the structure is not based on an explicit cognitive analysis, and, as in the composite practice version, the opportunity to practice is usually provided only after all the required steps have been presented.

In summary, the result was three versions of the lesson: (1) the original L-TRAN version currently in use (original); (2) the cognitive version in which both the declarative and procedural components were cognitively restructured (cognitive); and (3) a composite practice version in which only the presentation of declarative content was restructured (comp. practice). The experiment was designed to test the effectiveness of the cognitive and composite practice versions as compared to the original version.

Thirty-six enlisted Navy instructors participated as trainees in this experiment. The rank and rate of the instructors varied. They were selected from two categories based on their NTDS console experience. The less experienced participants were not NTDS console instructors and had little or no NTDS console experience. The more experienced participants were NTDS console instructors, but were not experienced in the general lesson content. There were 18 subjects in each category. Instructors were randomly assigned to one of three lesson conditions (i.e., the original version, the composite practice version, or the cognitive version). A pretest-posttest design was used.

As in the previous study, a performance test was designed that included most of the major lesson goals and procedures. The performance test was essentially a graphic simulation of a series of tactical situations composed of the scenarios presented during the lesson. It required the trainee to take the appropriate actions, based on knowledge of symbology and procedures. The performance test was made up of 18 procedures. Correct performance of each procedure was scored manually by an experimenter standing behind the subject. In addition, there were paper and pencil tests for recognition and recall of the lesson material.

Results and Discussion

The goal of the analysis was to determine the effect of lesson structure on procedural learning. The criterion for having learned a procedure is error-free performance of that procedure on the postlesson simulation test. The criterion for having learned all the procedures presented in the lesson is error-free performance on the entire simulation. Since time was limited to one pass through the lesson, it was not feasible to train each subject to asymptotic performance and then to measure the amount of training time required to reach criterion. In the ET environment, the trainee has a limited time available for training and should, therefore, get as much as possible out of each session. An alternative to repeating the lesson until some criterion is reached is to look at how close subjects can come to that criterion after one pass through the lesson. Of the 36 subjects, 27 made two or less errors (of 18 possible) on the post lesson performance test, the remaining nine made five or more errors. One measure of the relative effectiveness of the lesson structures is the percentage of subjects in each group who reached the given level of performance. Of the 27 subjects who made two or less errors, 11 were in the cognitive group (91%), nine in the composite practice group (75%), and seven in the original group (58%)

The three groups were compared for effectiveness of training as measured by performance error difference scores, that is the number of errors made on the pretest minus the number of errors made on the posttest. For all three groups, the decision to end the lesson was controlled by the subject. They always had the option of going back over any portion of the lesson before ending the lesson and attempting the performance test. An analysis of variance was performed on the difference scores for each subject in each group; the means are shown in Figure 3. Main effects were found for experience ($p<.01$) and for lesson structure ($p=.05$). There was no significant interaction.

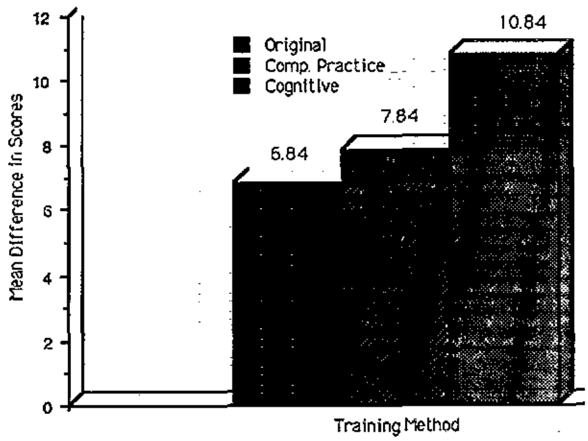

Figure 3. Mean reduction in performance errors from pretest to post test

The significant overall F allows us to reject the null hypothesis that there are no differences in performance that can be attributed to lesson structure. An analysis of differences in time to complete the lesson did not yield significant results and therefore do not account for differences in performance. In order to further assess the validity of our hypotheses, that both the declarative (composite practice) and the cognitive structuring of the exercises would improve performance over the original version, and to further identify the source of the variance, a pair of contrasts comparing the composite practice and the cognitive groups to the original group were computed. The differences between the group means were evaluated using Duncan's multiple comparison procedure. The difference between the means of the composite practice group and the original group did not fall within the significant range. The difference between the cognitive group means and original group means was found to be significant at $p<.05$. Therefore, our hypothesis was confirmed regarding the improvement which cognitive restructuring produced relative to a standard instructional systems design approach to curriculum content. However, perhaps more important than how the declarative information is structured is the manner in which the opportunities to practice using that information and receive feedback are distributed throughout the lesson.

Having further isolated the source of the main effect to the variability between the cognitive and original versions of the lesson, the next logical question is to ask whether this difference is equally distributed over both levels of experience? To evaluate the hypothesis that the approach will be more pronounced for less experienced subjects, comparisons were computed between the cognitive and original cells for both the less experienced and the more experienced groups. The difference between cell means was found to be significant for the less experienced group ($p<.025$), but not for the more experienced group. Figure 4 provides a comparative illustration of performance error difference means for each of the experience level by lesson structure cells.

While the lesson structure had an impact on the ability to perform procedures correctly, there was little effect on declarative recognition or recall. An analysis of the variance between the three groups in recognition test difference scores, and in recall test difference scores yielded no significant effects. That is, the difference in lesson structure did not influence the trainees' ability or motivation to memorize information as much as it did their ability to perform procedures. This can probably be attributed to a general 'learning by doing' principle and to the opportunity to receive immediate feedback. When we present opportunities for interaction with a system during training, we are emphasizing the importance of the information presented in the interactive format, and de-emphasizing all other information. The student attends to just that information with which he is encouraged to practice, and for which he is provided feedback.

CONCLUSIONS

The results of these experiments support the argument that procedural knowledge and skills can be acquired more effectively when instruction is presented in accordance with cognitive learning principles interpreted within a knowledge compilation framework. Those instructors with NTDS experience came to the training session with device specific procedural knowledge as well as general domain declarative and procedural knowledge. They are more highly tuned to the training environment and can therefore easily apply existing knowledge and skills to organize and interpret the new instructional material within the training environment. Differences in organization of the instructional text do not significantly enhance their ability to encode and apply the relevant information. In addition, these instructors have existing console specific procedural knowledge which probably makes practice at the microstep level unnecessary.

Those instructors without NTDS experience came to the training session with general domain knowledge but without device specific procedural knowledge. In fact as illustrated by their comments (e.g., "I'm not used to doing it this way."), their methods for achieving specific task goals often were in conflict with the target methods of the training session. The opportunity to interpret and apply declarative information at each substep before going to the composed level enhanced the effectiveness of the training lesson for these individuals. In general, while the effect was much more dramatic with entry level trainees, even these well trained and motivated individuals learn the procedural methods more effectively when the opportunity is provided to perform and practice those methods, and receive feedback, at each individual step in the procedure.

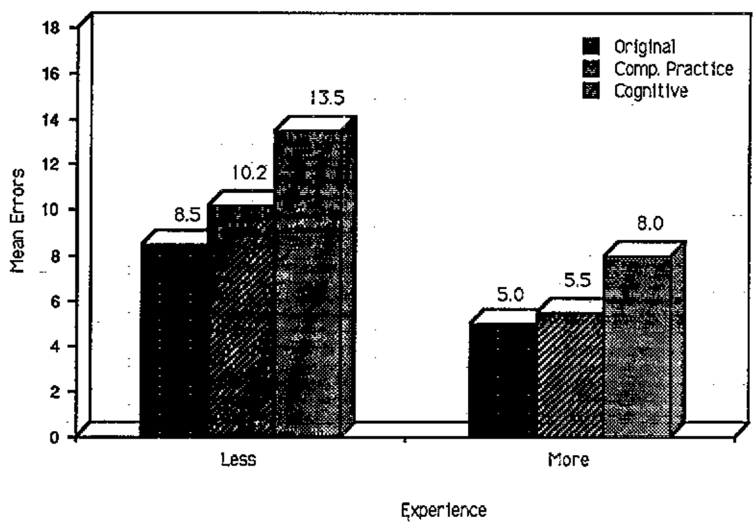

Figure 4. Mean pretest-post test performance difference for each of the experience levels by lesson structure cells

This approach can be applied to other embedded training systems and is currently being transitioned to the Aegis Computer Assisted Submode Training (CAST) system.

REFERENCES

Anderson, J. R. (1981). *Cognitive skills and their acquisition.* Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Anderson, J. R. (1983). *The architecture of cognition.* Cambridge, MA: Harvard University Press.

Anderson, J. R. (1986). Knowledge compilation: The general learning mechanism. In R. S. Michalski, J. G. Carbonell, & T. M., Mitchell, (Eds.). *Machine learning: An artificial intelligence approach* (Vol. 2, pp. 289-310). Los Altos, CA: Morgan Kaufmann.

Anderson, J. R., & Jeffries, R. (1985). Novice LISP errors: Undetected losses of information from working memory. *Human-Computer Interaction,* 22 403-423.

Anderson, J. R., Conrad, F. G., & Corbett, A. T. (1989). Skill acquisition and the LISP tutor. *Cognitive Science,* 13, 467-505.

Anderson, J. R., Boyle, C. F., Corbett, A. T., & Lewis, M. W. (1990). Cognitive modeling and intelligent tutoring. *Artificial Intelligence,* 42, 7-49.

Bovair, S., & Kieras, D. (1990). Toward a model of acquiring procedures from text. In Bass, Kamil, Rosenthal, and Pearson (Eds.), *Handbook of reading research* (Vol. II). New York: Longman.

Carolan, T.F., Williams K.E., and Moskal P.D. (1991). Structuring Curriculum for Embedded Training: An Evaluation in an Operational Setting. Final technical report submitted to NTSC.

Carroll, J. M. (1990). *The Nurnberg Funnel: Designing minimalist instruction for practical computer skill.* Cambridge, MA: MIT Press.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction.* Hillsdale, New Jersey: Lawrence Erlbaum Assoc.

Chi, M. J. H., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self explanations: How students study and use examples in learning to solve problems. *Cognitive Science,* 13 (2), 145-182.

Glaser, R., & Bassok, M. (1989) Learning theory and the study of instruction. *Annual Review of Psychology,* 40, 631-666

Hayes-Roth, F., Klahr, P., & Mostow, D. J. (1981). Advice taking and knowledge refinement: An iterative view of skill acquisition. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 231-253). Hillsdale, NJ: Lawrence Erlbaum Associates.

Johnson, W. L. (1991) Review of J. Psotka, L. D. Massey and S. A. Mutter *Intelligent Tutoring Systems: Lessons Learned.* Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Kieras, D. E. (1987a). *The role of cognitive simulation models in the development of advanced training and testing systems.* Office of Naval Research Technical Report No. 23 (TR-87/ONR-23).

Kieras, D. E. (1987b). *What model should be taught: Choosing instructional content for complex engineered systems.* Office of Naval Research Technical Report No. 24 (TR-87/ONR-24).

Kieras, D. E. (1988). Towards a practical GOMS model methodology for user interface design. In M. Helander (Ed.), *Handbook of human-computer interaction* (pp. 135-157). North Holland: Elsevier Science Publishers.

Larkin, J. H., & Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. In M. D. Ringle, J.G. Greeno, and W. G. Lehnert (Eds.), *Cognitive Science* (Vol.11, pp. 65-100). Norwood, NJ: Ablex Publishing Corp.

Newell, A., & Simon, H. A. (1972). *Human problem solving.* Englewood Cliffs, NJ: Prentice Hall.

Reder, L. M., Charney, D. H. & Morgan, K. I. (1986). The role of elaborations in learning skill from an instructional text. *Memory and Cognition,* 14, 64-78.

Reif, F. (1989). Interpretation of scientific or mathematical concepts: Cognitive issues and instructional implications. *Cognitive Science,* 11, 395-416.

Rosenbloom, P. S., & Newell, A. (1986). The chunking of goal hierarchies: A generalized model of practice. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (Vol. 2, pp. 247-288). Los Altos, CA: Morgan Kaufmann.

Williams, K. E., & Reynolds, R. E. (1989) *Embedded training instructional technology enhancement* (NTSC Tech. Rep. 89-034). Orlando, FL: Naval Training Systems.

Williams, K. E., Reynolds, R. E., & Carolan, T. F. (1989). *Knowledge engineering domain knowledge for an intelligent computer aided instruction system.* In the proceedings of the Second Florida Artificial Intelligence Research Symposium, April 1989.

Williams, K. E., Reynolds, R. E., & Carolan, T. F. (1990). *Automated Adaptive Instruction for Embedded Training.* In the proceedings from the Interservice/Industry Training Systems Conference (I/ITSC) 12th annual meeting, November 1990.

Williams, K. E., Reynolds, R. E., Carolan, T. F., Anglin, P. D., & Shrestha, L. B. (1989) *An evaluation of a methodology for cognitively structuring and adaptively sequencing exercise content for embedded training.* (NTSC Tech. Rep. 89-035), Orlando, Florida: Naval Training Systems Center.