N89-19855

INTELLIGENT TUTORING IN THE SPACECRAFT COMMAND/CONTROL ENVIRONMENT

Walter F. Truszkowski
Code 522.1
NASA/Goddard Space Flight Center
Greenbelt, MD 20771

ABSTRACT

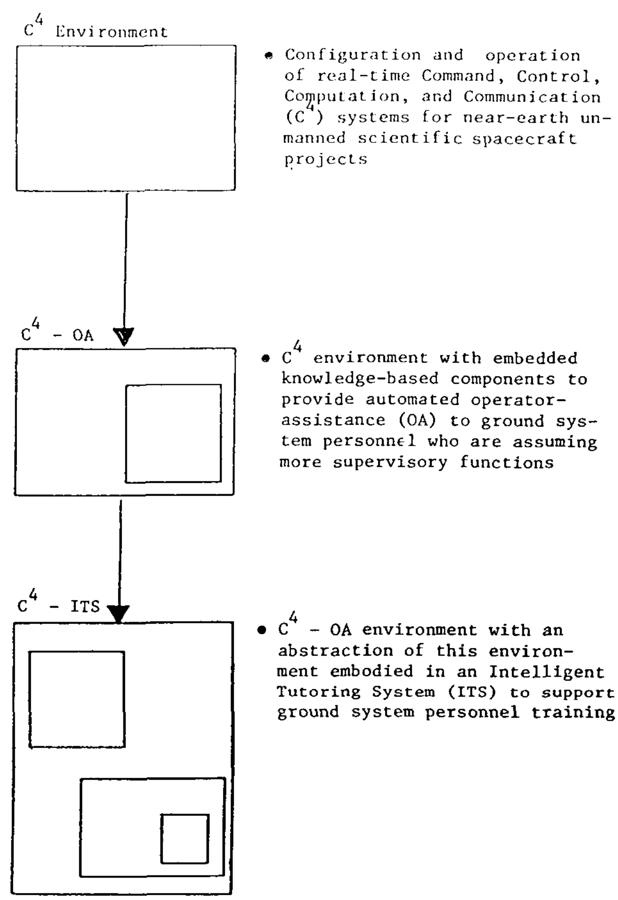
The spacecraft command/control environment is becoming increasingly complex. At this point in time as we are entering the era of Space Station and the era of more highly automated systems it is evident that the critical roles played by operations personnel in supervising the many required control center system components is becoming more cognitively demanding. In addition, the changing and emerging roles in the operations picture will have far-reaching effects on the achievement of mission objectives. Thus highly trained and competent operations personnel are mandatory for success.

Keeping pace with these developments has been computer-aided instruction utilizing various artificial intelligence technologies. The impacts of this growing capability on the stringent requirements for efficient and effective control center operations personnel is an area of much concentrated study.

This paper addresses the current research and development efforts in the area of automated tutoring systems for the spacecraft command/control environment being conducted by the Goddard Space Flight Center in conjunction with the Center for Man/Machine Studies at the Georgia Institute of Technology.

INTRODUCTION

Goddard's involvement with Intelligent Tutoring Systems (ITS) is coming about through an evolutionary system-upgrade process whose catalyst is the embedding of knowledge-based Operator Assistants (OA) in the ground systems for near-earth unmanned scientific satellites. Figure 1. depicts this evoluntionary process. The paper will concentrate on the three major phases of this system evolution and will detail the architectural aspects of the intelligent tutoring concept being formulated. Since data from some initial experiments is still being analyzed the performance of a prototype ITS in a ground system environment will be the subject matter of future papers.

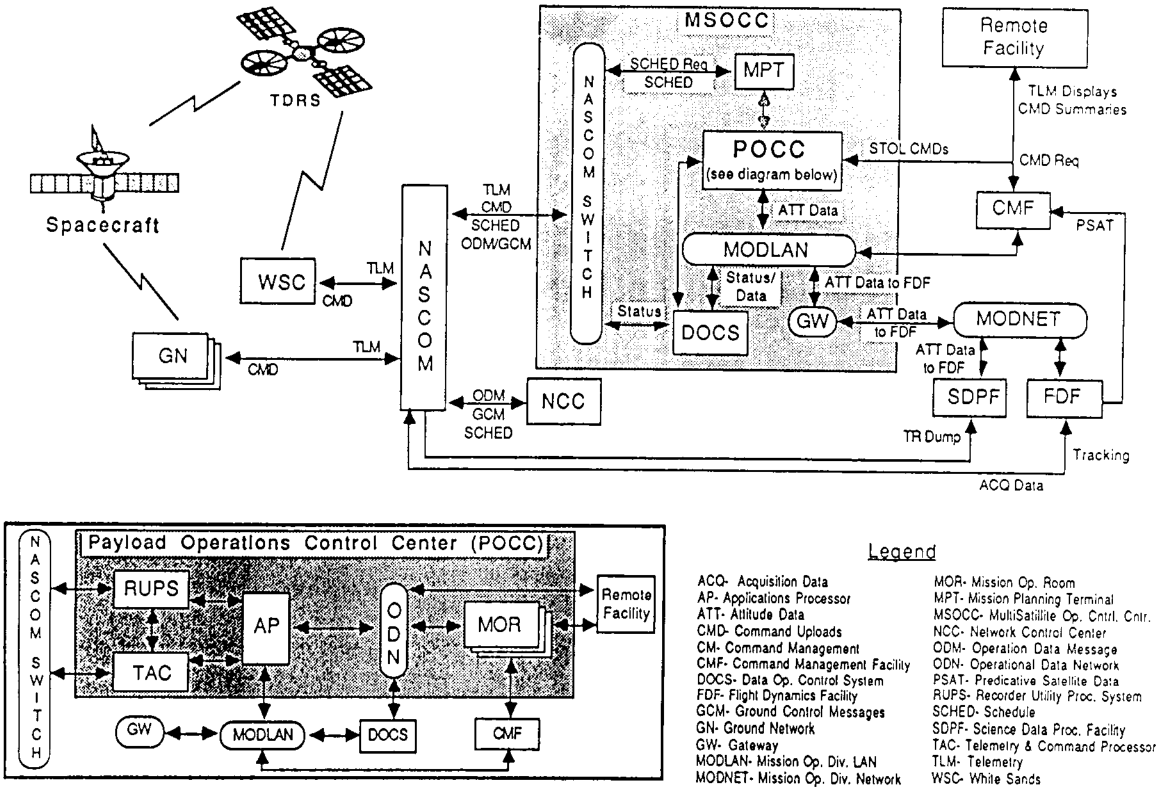


Evolution of C^4 Environment

Figure 1.

C^4 ENVIRONMENT

The environment in which we plan to embed intelligent tutoring systems is that which supports the command, control, computation and communication functions for near-earth unmanned scientific satellites. An abstracted view of this environment is presented in Figure 2.



The NASA End-to-End Data/Information System

Figure 2.

This NASA End-to End Date/Information System which embodies the C^4 functions is a highly complex and expensive real-time system which involves many personnel to orchestrate and participate in the efficient and effective operations needed to realize a successful spacecraft mission. The heart of the ground system is the Payload Operations Control Center (POCC). It is here where the health and safety of the spacecraft is monitored, where anamalous behaviors are detected and corrected, where normal commanding of the spacecraft is conducted, where initiation of the prescribed science agendas are triggered and where interfaces to remote facilities housing principal science investigators are maintained for the transmission of telemetered science data.

A major supporting function is provided to the POCC's by the Data Operations Control (DOC) system which supports the configuration and deconfiguration of communication and computer system support resources required for successful mission activities. To support the design and development of Operator's Associates (OA) which constitutes the second phase of the evolution, a real-time interactive simulation of the operator interface to a ground control system for unmanned earth-orbiting satellites was developed. This system, known as GT-MSOCC, provides a high fidelity environment in which to place prototype OAs for evaluation. The major functions supported by GT-MSOCC are

- control of current missions
- system configurations to meet support requests
- compensation for response schedule failures
- deconfiguration of systems

The reader is directed to [1] for a full discussion of the GT-MSOCC system.

C^4 - OA ENVIRONMENT

The following, based on [2,3,4,5], introduces the Operator Function Model (OFM) and its role in the concept of an Operator's Associate.

For the design of an Operator's Associate a model of the human operator is needed and to maximize its usefulness it should be e normative. In this way it will be able to evaluate actual human performance and if necessary offer advice and reminders. Furthermore, the model should have normative procedural knowledge so that is will be able to control any part of the system on command. In order to implement a model, it must be computational (i.e., well-specified enough to code into software). The model should be an operator model. Since the Operator's Associate will in general have access only to the operator's actions, there must be a

ORIGINAL PAGE IS
OF POOR QUALITY

way of representing how actions reflect cognitive processes. The model's structure should be a blend of a task performance and metacognitive model. We need to consider both the domain-specific characteristics of performance and domain-independent problem-solving behavior.

The basis for our operator's associate is the Operator Function Model (OFM). The OFM is a mathematical representation of how an operator might decompose the task of controlling a complex system into its simpler parts. The OFM is structured as a heterarchic-hierarchic network of finite-state automata. It is also a dynamic model, and this dynamic quality is represented as next-state transition functions that describe movement between states (nodes in the network). The OFM models operator-system interaction rather than the workings of the systems itself. The nodes in the network are operator actions, tasks, subfunctions, and functions. In particular, at the topmost heterarchic level are the major operator functions. Each function decomposes hierarchically into subfunctions, tasks and actions (either cognitive or manual). The next state transition functions can be modeled as system triggering events that cause the operator to switch to a different function, subfunction, or task. Figure 3 illustrates the OFM heterarchic-hierarchic framework.

The OFM specifies normatively how an operator should control the system. Given that the operator-system interaction is well-defined, an OFM can be constructed to model at least one reasonable method of control. Thus, the OFM is a prescriptive model in that it specifies non-deterministically a set of plausible manual and cognitive control actions, as well as goals and subgoals, given the current system state. Operator behavior is prescribed in the context of the current state of the system. As a well-defined mathematical entity, the OFM is a computational model of human performance. The OFM is an operator model that is concerned with the operator-system interaction and the operator's functions within that interaction. Finally, the OFM is both a task and metacognitive model. It is a task performance model in that actual operator actions are mapped onto hypothesized tasks, and errors of omission or commission are clearly recognizable. It is also in part a metacognitive model that characterizes generally the decomposition of operator functions from goals to subgoals, function, and manual and cognitive actions. It combines the richness of a multi-leveled representation with the mathematical rigor necessary for implementation. Thus, the OFM, both comceptually and methodologically, is a suitable model for the basis of an intelligent aiding system.

After the successful application of the OFM in analyzing and developing a normative model of the subset of operator activities in the Multi-Satellite Operations Control Center (MSOCC) it became apparent that this modelling technique could be used to develop a knowledge base for an expert system whose function was to provide assistance to an operator. This expert system, named OFMspert, is an example of a type of intelligent system we call an "Operator Associate". To date an experimental version of OFMspert has been developed and demonstrated at the Center for Man-Machine Studies at Georgia Tech.

This line of research recognizes that today and well into the future the human operator is a

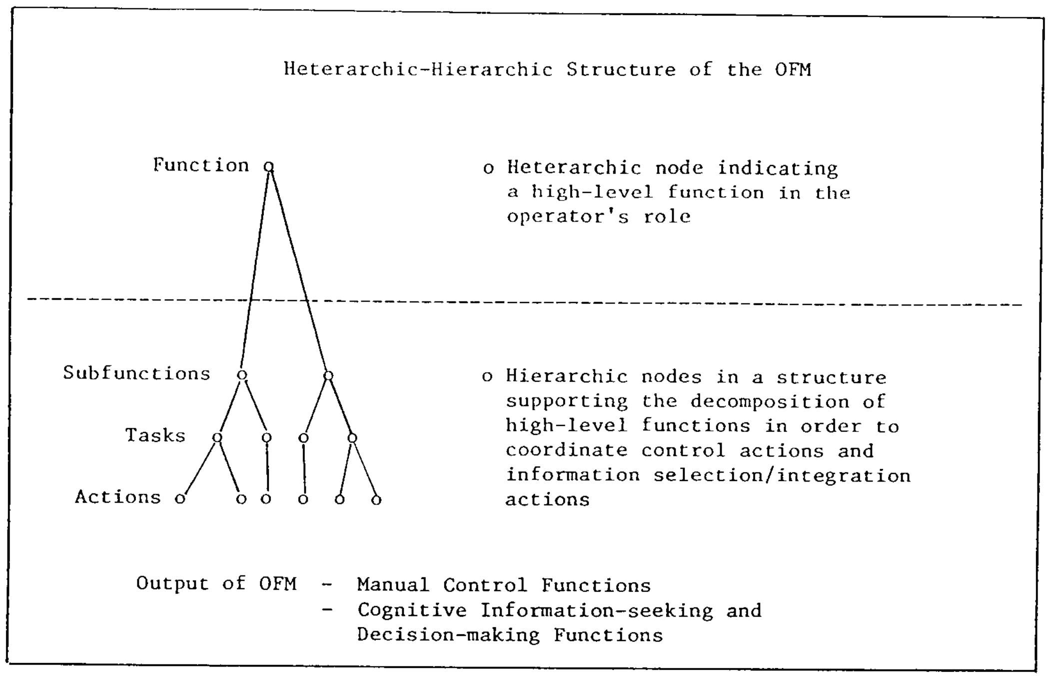


Figure 3.

critical component in control systems. The goal of an Operator Associate is to enhance and amplify the skills of the human operator and to exploit the strengths of all system components human or otherwise. The Operator Associate is designed to provide a dynamic symbiosis between the human operator and the rest of the control system.

The basic requirement for a viable Operator Associate is a normative model of the human operator behavior in various system contexts. For our system this model is provided by the OFM. As a basis the OFM provides information relative to the current operator state, predicted operator state and an assessment/predictor of the operator's goals, functions, intuition and performance.

In characterizing our version of Operator Associate two broad classes of operational capabilities or properties emerge, namely control and understanding. The control properties allow for the assumption,by the Associate, of varying levels of dynamic control of some part of the operational system. The level of control turned over to the Associate is determined by the human operator. The understanding properties provide the Associate with the capability of inferring current system goals, and offering context dependent assistance, advice and/or reminders to the human operator.

At the current time the focus of our development activity is on the understanding of the Associate. The level of understanding which can be supported by the Associate is, of course, a function of the application of the underlying operator model in explaining system operations.

Part of any understanding system are functions which we collectively call "intent inferencing". Intent inferencing tries to provide plausible explanations for observed operator actions given the current system state and past operator actions. Intent inferencing attempts to understand operator actions by interpreting them within the context of some normative model. In our case this normative model is provided by the OFM.

In providing information to support intent infercing the heterarchic-hierarchic structure of the OFM comes into play. Briefly, the heterarchic nodes correspond to the high-level functions in an operator's role. Each such high-level function has associated with it a three-level hierarchy which supports a decomposition of the function into subfunctions, tasks, and actions. In the current implementation of the Associate this model information is manipulated by means of a blackboard system typical of those in use in current artificial intelligence systems.

Figure 4 depicts the blackboard model used in OFMspert. This model uses a three level hierarchy of knowledge sources (KS). The Strategy KS determines what type of event to focus on, the Activator KS selects a specialist KS appropriate for the selected event, and the Specialist KS provides the knowledge used to make modifications

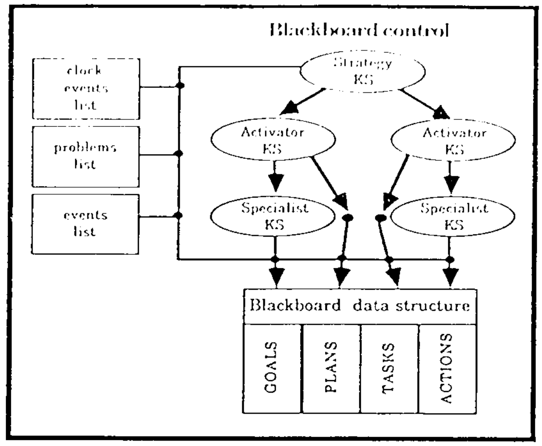


Figure 4.

to the blackboard. The current overall architecture for the OFMspert is shown in Figure 5. The Enhanced Normative Model with its Goals, Plan and Task (GPT) is based on the OFM.

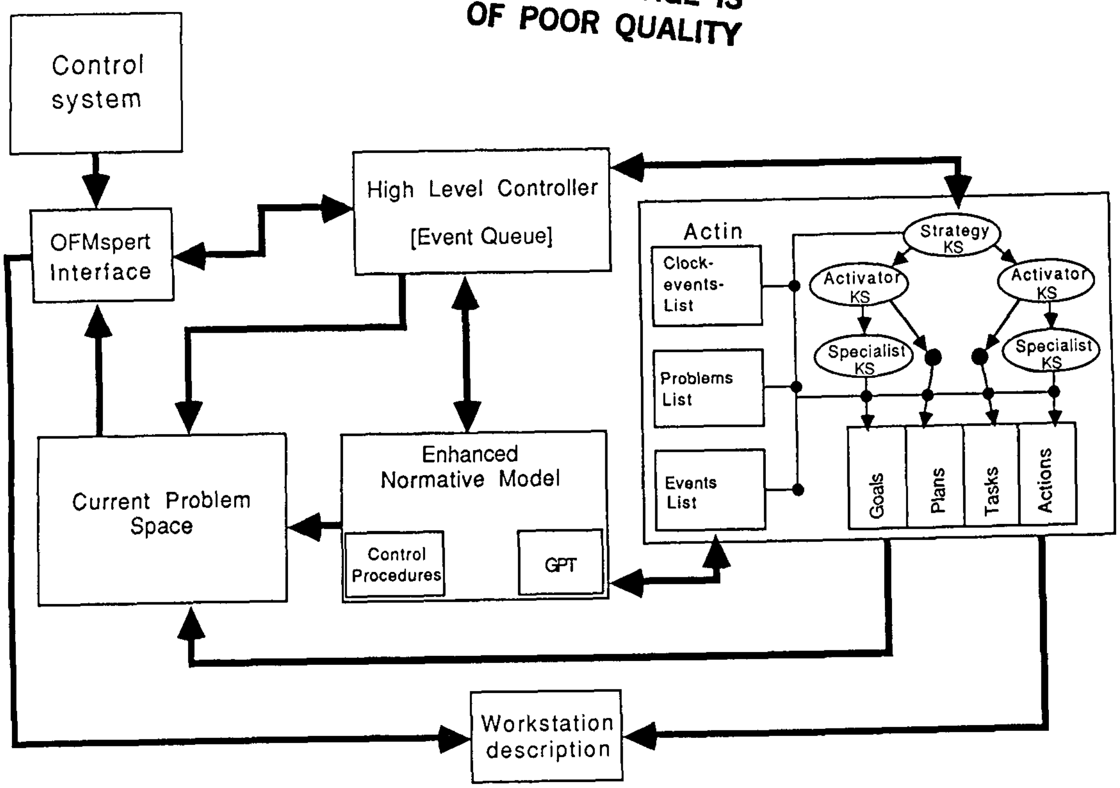
OFMspert is proposed as a conceptual foundation for the implementation of artificial intelligence in POCC ground control systems. The OFMspert does not automate a portion of the control system, but rather extends the capabilities of the human operator responsible for system operation. OFMspert is an architecture that provides the operator with an automated associate to which the operator can delegate routine tasks. The transfer of control from human to computer and back is accomplished smoothly and the computer-based associate is capable of inferring current system state and likely operator functions. Finally, OFMspert is capable of explaining what it is doing; explanations are given in the context of the OFM that defines both the operator and computer-based associate's role. These capabilities rest on the integrity of the OFM and the knowledge structures that comprise it. As such, OFMspert is an extension of the OFM models developed as part of this program with extensions in areas that are quite promising for C^4 applications.

C^4 - ITS

ITSSO (intelligent tutoring system for satellite operators) [6] is a design for a model-based, on-line tutuoring system for operators of complex, predominantly automated dynamic systems. The design is illustrated in the context of GT-MSOCC. The purpose is to provide embedded training for novice operators learnign how to supervise a complex, multifunction ground control system. This application is one with a great deal of pratical appeal for current and future ground control applications.

Computer-based training is a natural application of the OFM for a dynamic system. The model dynamically specifies current operator functions related subfunctions, and both manual and cogni-

ORIGINAL PAGE IS
OF POOR QUALITY



OFMspert Architecture

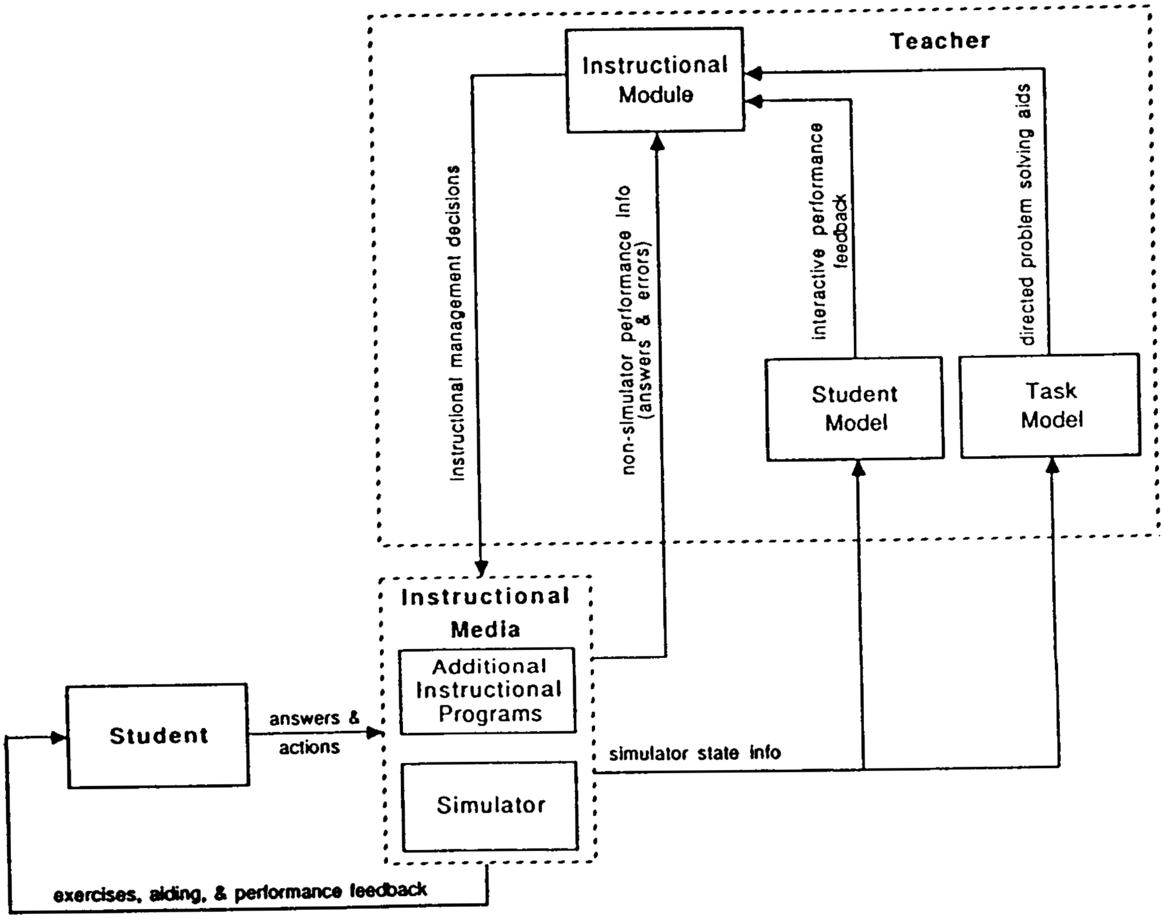
Figure 5.

tive actions. The OFM is a normative model and runs on-line. It interprets current system state and infers likely operator functions. As a result for novice operators, it could be quite useful in providing suggestions about what to do and how to do it given current system state.

A more interesting feature of the work is the intelligent and adaptive nature of the training system. In addition to simply providing suggestions, the OFM can be used to model the novice's current knowledge of the operator tasks. By examining the differences between the normative OFM and the student OFM, an on-line system tailors problems or scenarios within the context of GT-MSOCC to help the novice learn a given function or a procedure needed to carry out the function.

Essentially, the ITSSO research assumes that intelligent, on-line training or tutoring involves several components: a domain or task to be learned, e.g., GT-MSOCC operator operations, a model of the teacher (normative OFM), a model of the novice or student (descriptive OFM), and a set of teaching strategies. Teaching strategies allow the teacher model to modify the domain in order to teach the student new knowledge, increase experience or understanding of previeouly learned knowledge or procedures, and remediate error or misconceptions in the student's understanding of system or operator functions.

The current ITSSO concept is based on a tutoring system architecture established in [1]. Figure 6 displays this architecture.



Architecture for an Instructional System.

Figure 6.

One of the major basic accomplishments that has been achieved principally in the work conducted by our colleagues at Georgia Tech has been a characterization of the three major models, the task, student and instructional models, which form a basis of our ITS architectural concepts. The preliminary and high-level view of these models is being refined and detailed as we gain experience with the models.

The following tables, developed in [2], present a summary of the ITS architecture as it is currently envisioned.

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS OF POOR QUALITY

TABLE I SUMMARY OF ITS ARCHITECTURE

Assumptions:

The instructional program has access to a simulator of the large-scale, dynamic system on which the task to be learned is performed.

General Knowledge Representation Requirements:

1. Represent knowledge from multiple viewpoints
2. Represent knowledge at mulitple levels
3. Represent knowledge at the correct grain size

Task Model

Function:

The function of the task model is to provide directed problem-solving aids.

Description:

The task model is a prescriptive model of the task, which specifies dynamically the relationship between goals, system states, and actions.

Requirements:

1. The model prescribes actions based on the current system state and current operator goals.
   a. It must account for a dynamically changing system.
   b. Suggested actions must be readily understandable by the student.
   c. Suggested actions must be readily applicable by the student.
2. Students must be able to learn the the prescribed strategy.
3. Knowledge is organized as concepts and metaconcepts.

Knowledge Representation:

1. Concepts are represented as frames.
2. Metaconcepts are represented as production rules.

Student Model

Function:

The student model identifies correct and erroneous student actions and provides performance feedback.

Description:

The student model is a descriptive model of the student, which organizes student actions in the context of expert behavior.

Requirements:

1. The student model provides a method of organizing correct student actions.
2. It provides a method of orginizing student errors.

Knowledge Representation:

1. Concepts are represented as frames with slots for correct and incorrect actions.
2. Metaconcepts are represented as production rules.
3. Error rules specify improtant or common errors.

Instructional Model

Function:

The function of the instructional module is to make insturctional management decisions.

Description:

The instructional module contains a set of production rules for choosing the instructional medium, curriculum, pace of instruction, amount of feedback, and degree of control a student may exercise.

Requirements:

1. Instructional media
   - Alternate problem solving with quizzes and exercises
2. Curriculum
   - The order of presentation of material proceeds from easy to difficult
3. Pace
   a. Present new concepts if none are to be reviewed.
   b. Alternate new and review concepts.
   c. Review concepts in which the number of errors exceeds the threshold value.
4. Feedback
   a. Provide feedback during only part of the lesson.
   b. When allowing feedback, present feedback immediately after every error.
   c. Summarize feedback for a problem when a student requests it and at the end of a problem.

Knowledge Representation

1. Problems and evercises are contained within slots of the concept frames.
2. Rules make instructional management decisions.

ORIGINAL PAGE IS OF POOR QUALITY

CONCLUSIONS

A prototype of ITSSO was implemented in the context of GT-MSOCC and empirically evaluated. Tem subjects, tutored by ITSSO, controlled the GT-MSOCC system. After training, the system performance of ITSSO-trained subjects was compared to that of subjects trained by human instructors. Data analysis is almost complete. Preliminary analysis indicates that subjects trained with ITSSO control GT-MSOCC as effectively as those trained one-on-one by a human instructor.

As the ITSSO concepts mature it is planned that ITSs will be introduced into the operational environments here at Goddard to provide additional support and training for the operators of the future.

The purpose of this paper has been to introduce the reader to emerging ITS-related research and development being sponsored by the Goddard Space Flight Center. Spurred on by the success we have experienced in introducing expert system technology into our C^4 environments, the expansion of our knowledge-based technologies to include ITSs is proving to be a productive and focused application of experiences with artificial intelligence.

A major focus of our ITS-related research will be teaching strategies which can reflect changes in both the Student and Domain Models.

The reader who would like a more detailed view of our current work is urged to consult the references from which this article was developed.

ACKNOWLEDGEMENTS

The author wishes to sincerely thank Dr. Christine Mitchell and her associates at the Center for Man-Machine Systems Research at Georgia Institute of Technology for providing a rich research base to support continued work on ITSs.

Thanks also go to Robert Dominy, Troy Ames and Robert Dutilly at the Goddard Space Flight Center for helping to clarify the C^4 Environment and to Dorothy Perkins and John Dalton for providing management support and encouragement.

REFERENCES

1. Mitchell, C. M., "GT-MSOCC: A Research Domain for Modeling Human-Computer Interaction and Aiding Decision Making in Supervisory Control Systems," IEEE Trans., Systems, Man, and Cybernetics, SMC-17, No. 4, July/August, pgs. 553-572, 1987.

2. Fath, J. L., "An Architecture for Adaptive Computer-Assisted Instruction Programs for Complex Dynamic Systems," Center for Man-Machine Systems Research, Georgia Institute of Technology, Report No. 87-3, June, 1987.

3. Rubin, K. S., Jones, Patricia M., and Mitchell, C. M., "OFMspert: Inference of Operator Intentions in Supervisory Control Using A Blackboard Architecture," Center for Man-Machine Systems Research, Georgia Insitute of Technology, Report No. 87-6, December, 1987.

4. Jones, P. M., "Constructing and Validating a Model-Based Operator's Associate," Center for Man-Machine Systems Research, Georgia Institute of Technology, Report No. 88-1, March, 1988.

5. Truszkowski, W., "The Operators Associate: An Operator Function Model Expert System," Research and Technology 1987,Goddard Space Flight Center, May, 1988, pgs. 207-209.

6. Resnick, D. E., "ITSSO: Intelligent Tutoring System for Satellite Operators," Center for Man-Machine Systems Research, Georgia Institute of Technology, Report No. 88-2, June, 1988 (in progress).