(2)

Validation and Application of COGNET Model of Human-Computer Interaction in Naval Air ASW

Wayne Zachary, Ph.D.
Joan M. Ryder, Ph.D.
Lorna Ross
Monica C. Zubritzky

CHI SYSTEMS
Gwynedd Plaza III
Bethlehem Pike and Sheble Lane
Spring House, PA 19477

CHI SYSTEMS TECHNICAL REPORT 900531-8704
31 May, 1990

CONTRACT N00014-87-C-0814

R&T Project Code: 4429007

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b RESTRICTIVE MARKINGS N.A.
2a. SECURITY CLASSIFICATION AUTHORITY N.A. 2b. DECLASSIFICATION / DOWNGRADING SCHEDULE N.A.	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited
4. PERFORMING ORGANIZATION REPORT NUMBER(S) Technical Report 900531.8704	5. MONITORING ORGANIZATION REPORT NUMBER(S) Same
6a. NAME OF PERFORMING ORGANIZATION CHI Systems Inc. 6b OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Office of Naval Research
6c. ADDRESS (City, State, and ZIP Code) Gwynedd Plaza III Bethlehem Pike at Sheble Lane Spring House, PA 19477	7b ADDRESS (City, State, and ZIP Code) 800 N. Quincy Street Arlington, VA 22217-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION Office of Naval Research 8b OFFICE SYMBOL (If applicable) Code 1142PS	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-87-C-0814
8c. ADDRESS (City, State, and ZIP Code) 800 N. Quincy Street Arlington, VA 22217-5000	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO: 61153N 42 PROJECT NO: RR04209 TASK NO: 0420901 WORK UNIT ACCESSION NO.: R&T4429007

11. TITLE (Include Security Classification)
Validation and Application of COGNET Model of Human-Computer Interaction in Naval Air ASW

12. PERSONAL AUTHOR(S)
Zachary, W.; Ryder, J.; Ross, L.R.; Zubritzky, M.

13a. TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Technical	FROM 87/09/30 TO 90/03/30	31 May 1990	84

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	human-computer interaction; decision-making; cognitive model; ASW; COGNET; attention; GOMS; adaptive interface

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

This is the final report on a two-year research effort to investigate the cognitive basis for human-computer interaction and decision-making in complex, real-world environments, particularly those which unfold in real-time and make multiple demands on the attention of the human decision-maker. The main emphasis in the project has been to explore the extent to which models of the computer-user's problem-solving strategies in these real-time multi-tasking (RTMT) environments can lead to the design of more effective human-computer interfaces. This research developed the COGNET (COGnitive Network of Tasks) RTMT modeling framework, as an integration of the GOMS and blackboard cognitive modeling techniques. COGNET was then applied to a vehicle tracking domain based on Naval Air AntiSubmarine Warfare (ASW). The COGNET Air ASW model was validated by comparing the attention shifts and task performance sequences generated by the model to experimental data from human ASW experts solving realistic problems. A first comparison used previously-recorded problem solving data that had been used to generate the Air

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Gerald Malecki	22b. TELEPHONE (Include Area Code) (202) 696-4044 22c OFFICE SYMBOL Code 1142

DD Form 1473, JUN 86 — Previous editions are obsolete — SECURITY CLASSIFICATION OF THIS PAGE

S/N 0102-LF-014-6603 — Unclassified

ASW model (for model verification). A second comparison used additional data from new human subjects and problems (for model validation). Verification data showed that COGNET model accounted for 93% of the human subject attention/task shifts, and actually predicting (in a temporal sense) 90% of them (with a mean prediction lead time of 4.1 minutes, over 27 attention/task shifts). Validation data showed a similar pattern. COGNET accounted for 94% of the observed attention shifts; and predicted human performance 90% of the time, with a mean prediction lead of 5.4 minutes (over 17 attention/task shifts).

The validated model was then used to develop an adaptive human-computer interface for the the vehicle tracking domain, using a novel human-computer interface architecture. The interface architecture included two subsystems -- an embedded user model and a set of performance aiding tools -- built on top of the baseline interface. The adaptive interface performs four specific adaptive functions:

1) alerts the user when the it believes the context is appropriate for initiation of or return to a specific task in the COGNET task network;

2) Indicates the expected order of precedence of the tasks, when multiple tasks are inferred as appropriate at the same time;

3) Provides decision structuring assistance, on user request, by identifying the internal organization of goals/subtasks within any given task identified as appropriate for initiation; and

4) automates performance of any or all of the subtasks in a task identified as appropriate for initiation, with the task instance adapted automatically to the interface's understanding of the current problem context.

These adaptive human-computer interaction functions represent an initial demonstration of how COGNET could be applied to extend user interface capability in important ways.

Table of Contents

List of Tables

List of Figures

Acknowledgements

We wish to acknowledge the important contributions to this research by the TACCOs stationed at the Naval Air Development Center and NAS Willow Grove who participated as experimental participants in this research. Without their enthusiasm and generous contribution of their own time, this research would not have been possible. They are not acknowledged by name only because their anonymity as experimental participants was promised. The authors also wish to thank Mr. Stanley Winsko (Naval Air Development Center) and LT Michael Holmes (USN Aerospace Medical Service Corps) in identifying and gaining access to experimental participants, and of Mr. Eric Haas for providing ASW expertise and advise. In addition, valuable comments and suggestions have been provided by Dr. John O'Hare of ONR, Mr. William Weiland of the University of Maryland, Dr. Lee Goodman of Magnavox Electronics (formerly of NADC), and Dr. Stanley Schwartz here at CHI Systems.

1. INTRODUCTION AND BACKGROUND

This is the final report on a two year research effort to investigate the cognitive basis for human-computer interaction and decision-making in complex, real-world environments, particularly those which unfold in real-time and make multiple demands on the attention of the human decision-maker. The main emphasis in the project has been to explore the extent to which models of the computer-user's problem-solving strategies in these real-time multi-tasking (RTMT) environments can lead to the design of a more effective human-computer interfaces. RTMT environments include many of the most challenging problem domains faced by humans. Examples of these environments include aircaft (and other vehicle) cockpits, nuclear power control rooms, automated manufacturing environments, air traffic control, hospital operating rooms, satellite and telecommunication network control, and weapons systems operation, to name but a few. These problem environments are undergoing rapid computerization, and are all critical to economic, personal, and national well-being. Therefore, they are inherently worthy of close study.

Three specific goals have been pursued in this project. The first was to develop new cognitive and human-computer interaction modeling methodologies and tools. At the outset of this reserach, the available tools for analyzing and representing cognitive processes in human-computer interaction were not directly applicable to real-time or multi-tasking domains. A major result of this research has been the development of such an RTMT modeling framework, which we have called COGNET (for COGnitive Network of Tasks).

The second goal of the research was to develop the modelling framework in the context of a specific, realistic RTMT domain. This is done to provide a basis for testing and refining the modeling languages and tools developed, and to demonstrate that the modeling framework was applicable to 'real-world' RTMT problems. The COGNET framework and the methodology used to apply it in developing specific human-computer interaction models was previously reported in Zachary, Ryder, and Zubritzky (1989), as was the initial applicaton of COGNET to the domain of Naval Air AntiSubmarine Warfare (ASW).

The third goal of the research was to explore the use of a COGNET model of Human Computer Interaction in a specific domain could be used as a basis for understanding human performance in the domain, and, if so, how it might be used as a basis for develoing more intelligent and adaptive human-computer interfaces in the domain. These subjects are the topic of this final report. The link between COGNET models of human-computer interaction and intelligent human-computer interfaces lies in the notion of embedded user models as discussed in Norman (1983) and elsewhere.

The theory of user models builds on the theory of mental models and their role in intelligent interaction (see various papers in Robertson, Zachary, and Black, Eds., 1990). This theory holds that one key way in which human-human interaction and traditional human-computer interaction differ is that when two human interact, each person has a mental model of the other, which is used to reason about how to conduct and adapt the interaction to the shared goal (e.g. Bruce and Newman, 1978) and/or the unfolding situation (e.g., Suchman, 1990). However, when a human and a computer interact, the computer does not have a model of the person. It cannot understand what the person is doing, nor reason about it, nor adapt the interaction to the person's problem solving approach or the evolving external situation. In other words, the computer lacks an embedded model of its user. This kind of embedded user model is particularly important in real-time domains where the cost of rigid control regiments on the part of the computer usually means lost time and lost opportunity. Eisenberg (1990) has very recently argued that having detailed models and expectations of the behavior of a 'partner' are critical to highly adaptive real-time interactions such as musical or sporting improvisation. Ideally, we would like the human-computer team to be as adaptive and improvisational as highly skilled human-human teams.

On the other hand, if the human-computer interface could be given a highly predictive model of the user, then the interface could use this to adaptively make itself more 'helpful' in several ways, including:

- identifying errors of commission or omission, particularly those performance errors that Norman (1983a) has called 'slips', and those competence errors that distinguish novice from expert performance;
- adapting the set of tools available at the interface to provide most ready access to those that are most relevant to the tasks that the user is now carrying out or is anticipated to be focusing on shortly; or
- providing reminders or alterts to the user when the user has failed to attend to some task or tasks that he was expected to perform.

These are, of course, not the only ways in which an embedded user model could be applied to enhance computer-human interaction. They are, however, the specific uses considered in the final phase of this research, and in this report.

This final phase of the reserach involved two major steps:

1) validating the ability of the COGNET model in the Air ASW domain to predict user performance; and

2) designing and implementing a prototype intelligent adpative interface that uses the COGNET Air ASW model as an embedded model of the Tactical Coordinator (TACCO), who is the key decision maker in the Air ASW mission.

As a preliminary to the validation and verification analysis, the COGNET framework and Air ASW COGNET model are reviewed in Section 2 below.

The actual validation effort is then described in Section 3. The methodology for analysis of the Air ASW model made maximum use of the Experimental Enviornment develolped earlier in this program (see Zachary and Zubritzky, 1988). Subjects were asked to solve Air ASW problems using the simulation testbed portion of the Experimental Enviornment, and their performance was logged by the computer. Then, the COGNET model was programmed and provided with the keystroke level log of a subject's actions as well as the state of the problem that was visible to the subject on the computer screen. From this keystroke level log and problem context data, the COGNET model was exercised to infer the sequence of the attention shifts (between well-defined tasks) that the user would be expected to exhibit , and the times in which these shifts would be first expected and the times at which they would not longer be appropriate. As detailed in Section 3., the model correctly predicted more than 90% of these shifts. Even more impressive, 90% of these inferences predicted the time window of the tasks, and 70% predicted the attention shifts within two minutes prior to their initiation by the human operator. Significant differences were not found between previously recorded logs of subjects used to develop the model and separate validation subjects. Taken together, the results of this analysis were taken to indicate that the model was indeed able to provide an Air ASW interface with an ability to model and anticipate the actions of its users.

The design and implementation of a demonstration intelligent interface is presented in Sections 4 and 5. Section 4 discusses the theoretical model underlying the use of embedded users models to achieve intelligent human-computer interaction. It also presents the computational architecture used to embed the COGNET model into the interface of the Experimental Environment's Air ASW mission simulator. Section 5 details the layout of the specific demonstration intelligent Air ASW interface built with the architecture introduced in Section 4, as well as the rationale for this specific design. This section also describes an example session with the intelligent interface, and how it differs from what is currently used in the fleet. Section 6 presents final conclusions and a summary of accomplishments of the program.

2. THE COGNET FRAMEWORK AND THE COGNET MODEL OF AIR ASW MISSION MANAGEMENT

This section reviews the structure of the COGNET modeling framework. It also describes the specific model that was built using this framework, and the vehicle tracking domain which the model addresses. It is the COGNET model of the vehicle tracking problem that is subjected to a detailed validation analysis in Section 3, and used as an embedded user model in an adaptive human-computer interface for the vehicle tracking domain in Section 5.

2.1 COGNET Framework and COGNET Modeling Notations

The conceptual basis for COGNET lies in the early work of Selfridge (1959). He proposed a "pandemonium" metaphor of cognitive processes composed of "shrieking demons." Each demon was able to perform some aspect of cognition and shrieked for attention as an opportunity arose for that process to occur. As the situation became closer to the ideal conditions for the demon, it shrieked louder and louder. Attention, in Selfridge's model, was simply the process of placating the shrieking demons by allowing the loudest one to act. Real-time multi-tasking arises in this conceptual framework when the context and temporal dynamics of the problem allow (or require) many of these shrieking demons to compete for attention in such a way that no one of them maintains control for very long.

COGNET represents RTMT decision making in a similar manner. The person is conceptualized as a network of cognitive tasks, each of which represents a partial or local strategy for performing some task or for solving some aspect of the overall problem. The flow of attention from one task to another is triggered by momentary changes in the problem environment. These changes may be the result of actions taken by the person or the result of actions of other agents and/or the environment as perceived by the decision-maker. Figure 2-1 shows an abstraction of how attention flows through this network of tasks.

In Figure 2-1, the user is performing a task ("Task 1"), when he/she receives notice of some condition at the workstation, such as an error or warning alert. This explicit condition triggers the user to defer further work on Task 1 and instead to initiate work on Task 4 (perhaps a trouble shooting task). While performing Task 4, however, a piece of information is observed on the screen. This datum, in the context of the trouble-shooting task and the user's prior experience with this specific information item, causes her/him to suspend Task 4 and instead begin Task 3 (perhaps an analytical or information gathering task). The analysis performed in doing this task then uncovers yet another piece of data ("Item D"), which in the context of the original condition (Condition A in Figure 2-1) leads the user to cease the analytical task (Task 3) and initiate yet another task (e.g., Task 5). While performing this task, however, the user enccunters condition A again (as he/she did while previously performing Task 1). Because the context is different now, though, the response is also different. Whereas previously Condition A triggered an initiation of Task 4, in this case it triggers an initiation of Task 2. Figure 2-1 thus indicates various factors that can influence shifts in attention among tasks -- explicit cues, prior knowledge, local problem-solving context, and associations or inferences (i.e., knowledge).

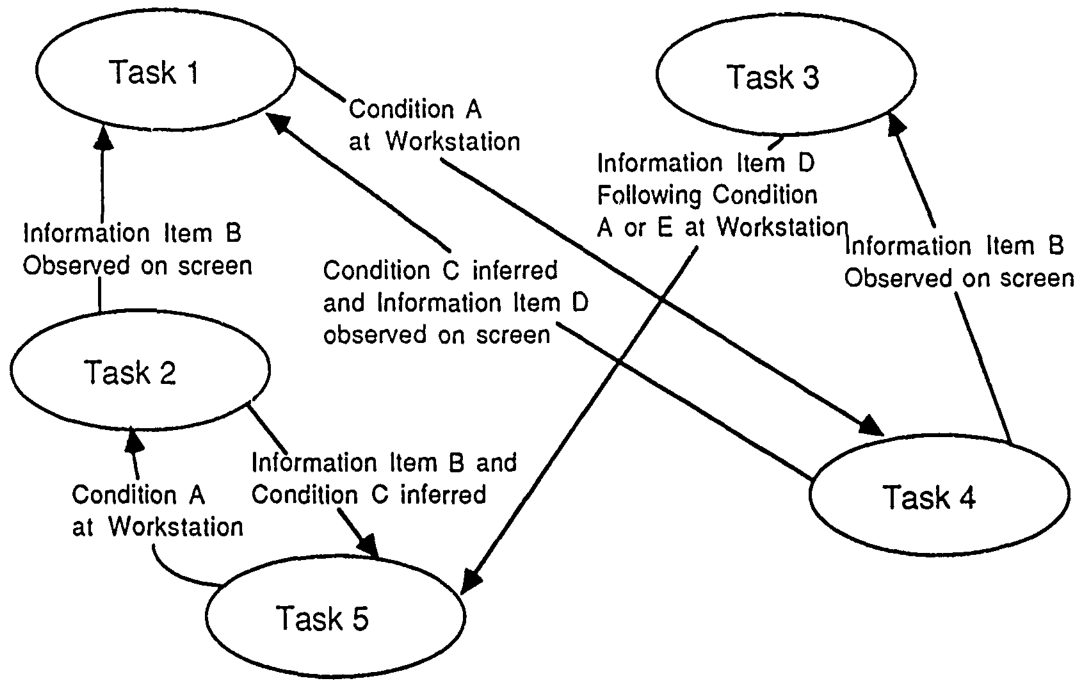


Figure 2-1 COGNET Concept

One element of COGNET not anticipated in Selfridge's metaphor is the basis for the coordination among the various tasks or 'demons'. Coordination, as defined in Malone (1990), refers to the means by which cooperating but independent agents organize their individual problem-solving activity to achieve a solution to a more global problem. In COGNET, each task acts as an independent problem-solving agent (like one of Selfridge's demons). Each task may be partly completed, interrupted by some other task, and perhaps later resumed at the place where it was interrupted. Thus, the tasks are all in various states of completion at any one time, giving them the appearance of concurrence, albeit not actual simultaneous activity. COGNET considers this to be a form of weak concurrence (as opposed to the strong concurrence of completely independent, active-in-parallel agents). Moreover, these weakly concurrent tasks (unlike Selfridge's demons), are also all interrelated, in that they each contribute to some higher-level problem-solving goal. This common interrelationship among the tasks -- their linkages to a common goal -- implies some mechanism for coordination among the tasks. In COGNET, coordination among the individual tasks is established through their use of a common global problem representation. That is, all tasks use the same (overall) representation of the problem being solved. Each task contributes to some specific portion of the problem (as bounded by the representation), acting to move that portion of the representation toward a solution (or at least away from some losing or unacceptable state).

The COGNET framework is actually a meta-model, or architecture, for building models of specific RTMT domains. The flow of cognitive processing in a COGNET model resides at any given moment in a specific task in the network. The focus of attention remains there for some time until it is captured by another task/decision node (by a change in the problem representation that enables the second task node to capture control from the first). The flow may also be opportunistic, such as when a goal-based shift is supported by a specific state of knowledge in the problem representation. Overall, the flow of attention between and among the tasks both reflects the changing context of the problem evolution (via the changes in the common problem representation). This flow also contributes to the problem evolution and change in context by directing the specific sequence of operations that are performed by the sequence of tasks that gain sufficient priority to gain control.

To support the development of domain-specific COGNET models, three formal constructs have been developed. The first is a global problem representation notation. The common problem representation shared by the individual cognitive tasks is a generalized, multi-panel blackboard structure of the kind described by Hayes-Roth (1983) and Nii (1986). The flow of information processing within this COGNET architecture is shown in Figure 2-2. A given Task (e.g., Task B in Figure 2-2) may be active at a certain point in the problem evolution. Task B 'reads' or uses certain information from the current blackboard contents in its task processes, and may then subrogate to Task A to provide more localized or complementary analysis. Task A both 'reads' information from the blackboard and posts new information to the blackboard. This new information, upon its posting, may then complete a blackboard pattern that satisfies a triggering condition for Task C, which then captures control. Task C similarly posts new information on the blackboard, but this does not satisfy the pattern associated with any trigger with sufficient priority to displace it. Ultimately, however, the activity undertaken in Task C leads to an action which involves a substantial time delay before its effects are known. Task C then suspends itself until these effects are known (i.e., posted on the blackboard), and this allows Task D to capture control. Task D previously had its triggering condition satisfied but had insufficient priority to capture attention from Task C, which is now suspended.

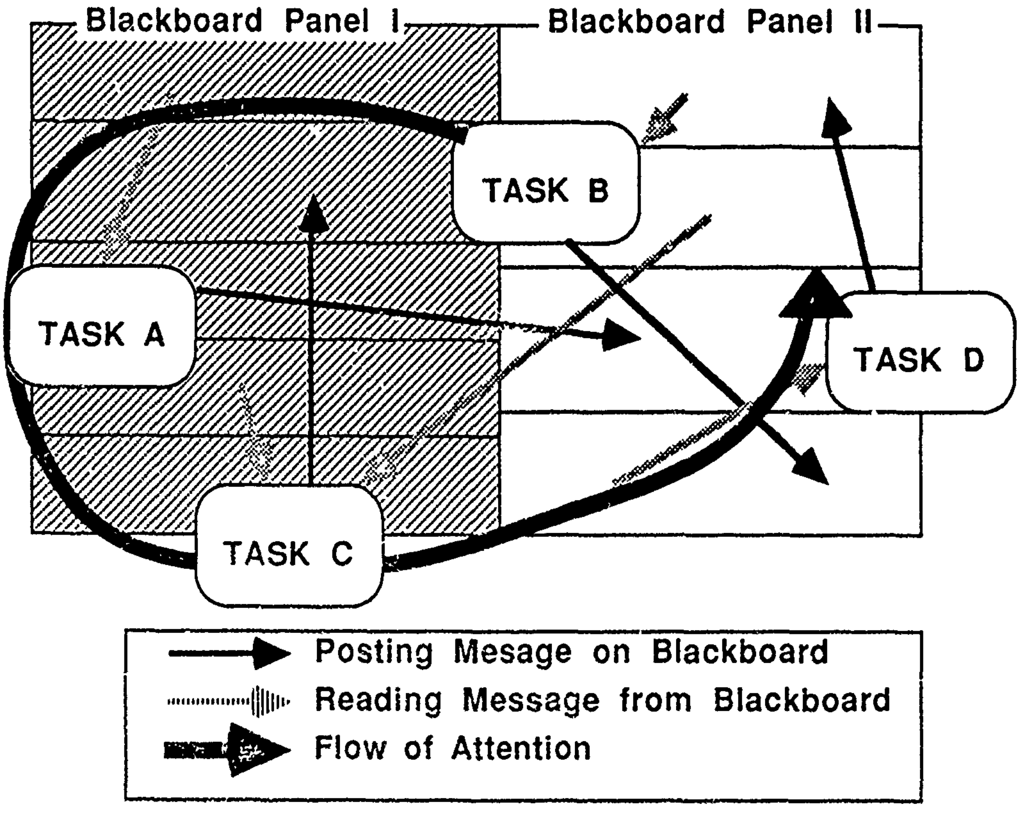


Figure 2-2 Attention Flow Between Tasks

The second formalism for building COGNET models is a notation for describing the information processing and associated person-machine behavioral interactions associated with each task in the COGNET network. The COGNET task level notation is summarized in Figure 2-3. This notation is related to the GOMS notation of Card, Moran and Newell (1984), but it includes additional features to allow for the accessing and creation of information on the blackboard structure, and to allow for the interruption, suspension, and subrogation of the current task. (For a detailed discussion of GOMS/COGNET differences, see Zachary, Ryder and Zubritzky (1989).

GOAL: Goal name...*TRIGGER*
GOAL: SUBGOAL NAME...<...*conditions*>
OPERATORs <...*conditions*>
Perform FUNCTION. <(accompanying data/ parameters)>
where FUNCTION=any invokable function
Point element/location on screen
Enter alphanumeric data in response to cues
Select item from screen
Post object on blackboard
Unpost object on blackboard
Transform object on blackboard
Suspend until condition
Subrogate to "new Goal Name"
Determine .....(generic mental operator -- find from display, calculate, decide,etc.)
TRIGGERS
Message pattern templates based on blackboard contents
CONDITIONS include
context free CONTROL information:
if ..., repeat until ..., repeat n times, optional, etc.
domain-dependent EVALUATIVE information:
boolean statements based on blackboard message patterns
SELECTION RULES
Use Method...based on selection factors
Selection Rules:
if condition then Method 1...
if condition then Method 2 <with probability .x>
METHODS
Method 1 Name:
list of operators (and subgoals)

Figure 2-3 COGNET Decision Task Description Language

Finally, the third element of the modeling notation is a mechanism to deal with perceptual events. A problem with the task description language as defined above is that it links the global problem representation -- the blackboard -- strictly to cognitive operations performed within the individual tasks in the COGNET network. However, in a human-computer interaction situation much elementary information on the blackboard arises from essentially perceptual events (e.g; observing a symbol appear or change on the display screen). COGNET also provides a notation similar to production rules, which describes the processes by which information, once perceived, is introduced into the person's understanding of the problem (i.e., the global problem representation). The perceptual demons therefore have a key role in changing the momentary problem representation (i.e., blackboard contents) independent from the operations of the individual tasks in the network. This role is pictured in an enhanced version of Figure 2-2 shown as Figure 2-4.

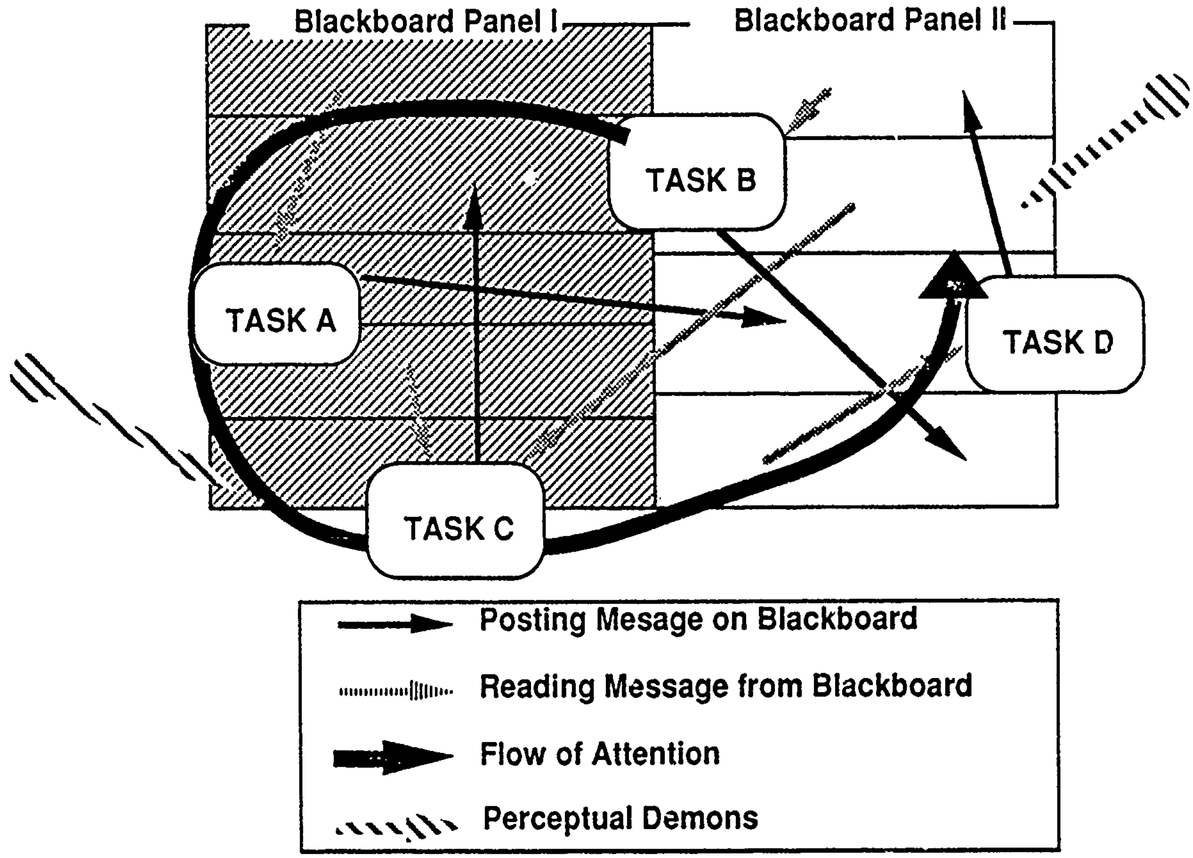


Figure 2-4 COGNET Attention Flow With Perceptual Demons

The COGNET architecture represents real-time multi-tasking in such a way that performance is sensitive to both situation effects and the experience and knowledge of the human operator. Experience and knowledge affect both the methods encoded in the individual tasks in the network as well as the triggers that allow for attention flow between tasks in the network. Situational effects are introduced by the attention mechanism, as well as by the use of a common, problem-specific representation by all the tasks in the network.

2.2 A Vehicle Tracking Problem Domain

It is almost impossible to study human cognitive processes (or tools to model them) without doing so in some specific problem domain. In addition, the domain had to require real-time computer-based problem solving on the part of the person, and to involve some competing demands for the person's attention. The domain selected to do this was a vehicle tracking task, in which the tracking is done via remote sensing devices. This task is based in the real-world domain of Naval Air Anti-Submarine Warfare (ASW), in which the vehicles being tracked are submarines. The person doing the tracking is located in an aircraft and using remote sensors to gather data about the vehicle, and is processing those data to detect, locate, and track the vehicle. The motivation for using this problem was discussed in Z[illegible] *et al.* (1989). A summary of the Air ASW domain is provided below for readers unfamiliar with it.

Air Anti-Submarine Warfare (ASW) is concerned with the detection and identification of a target submarine from an aircraft. The Air ASW mission begins with searching an area of ocean where a submarine is thought to be located. The mission progresses through a series of stages in which it systematically increases knowledge of the target's location, until it is possibl to track it (in peacetime), or attack and destroy it (in wartime). This problem is currently solved jointly by several cooperating crewmembers aboard a P-3 or S-3 aircraft. The crew typically consists of the aircraft pilot and navigator, one or more Sensor Operators (SENSOs), and other miscellaneous operators (not concerned with tactics). The Tactical Coordinator (TACCO) guides all of these personnel during the mission, coordinating their efforts and the available resources.

Virtually all information about the target is gained from a suite of remote sensors that includes passive acoustic sensors, or sonobuoys, active sonobuoys, RADAR, and Magnetic Anomaly Detection (MAD) sensors. Passive sonobuoys are the principal means for detection and localization of the submarine, and are preferred for tracking as well. They are often used in combination with active acoustic (and nonacoustic) sensors to speed the localization process or to deal with a target that has been alerted to its pursuit by the ASW aircrart. Typical phases of an Air ASW mission are:

1. Search -- Initial detection of a target is sought using passive sonobuoys dropped in the water to form a geometric pattern.
2. Direct-Path Contact -- Additional passive sonobuoys are dropped to obtain a sensor contact in which the target is detected within close range of a sonobuoy.
3. Target Fix -- One or more specific locational hypotheses about the target's precise location (i.e., 'fixes') are developed using analysis of sensor data over time.
4. Target Track -- Fixes are viewed together to develop a motion hypothesis that can predict target location over time.
5. Attack -- Doctrinal criteria for using a weapon are met, and a weapon, usually a torpedo, is deployed against the target track.
6. Alerted Target -- The target detects the presence of the #ASW aircraft, and takes evasive action, making localization, fix development, and tracking much more difficult.

During the use of passive sonobuoys, a physical phenomenon called the convergence zone (CZ) often makes the interpretation of the received data difficult. An acoustic sonobuoy can 'hear' sound directly propagated from an emitting source over a small distance; this is called its direct path (DP) detection range (e.g., 2 -5 nautical miles). Because of ducting of sound underwater, there may be a small annular region quite distant from the DP zone in which detection may also occur. This is called a CZ. There may be none, one, or possibly two CZs in any given acoustical environment. The presence of CZs creates a complex pattern of potential detection regions in a field of sonobuoys. Directional passive sonobuoys also provide a bearing to the target, but this bearing contains error, and does not help disambiguate whether the sound originates from the DP zone or from a first or second CZ.

Thus, the heart of the Air ASW problem involves using this ambiguous, errorful data from fields of sonobuoys in conjunction with data from active sonobuoys and nonacoustic sensors to detect the presence of a submarine and iteratively refine its location, course, and speed. The problem requires the TACCO to continuously revise target hypotheses based on the current situation and plan new tactics to gather further data, which in turn will cause an update of the hypotheses. Embedded within this process is the fact that the TACCO must direct the aircraft to deploy new sensors as a way of disambiguating data from existing sensors. This makes the movement dynamics and attendant time-lags another part of the decision process. Often, for example, the TACCO may know where to place an additional sensor, but cannot get the aircraft to the desired location in time for the data to be meaningful. Thus, the vehicle tracking problem based on Air ASW provides a rich and difficult domain in which human problem solvers must make real-time decisions and share their attention among a variety of tasks.

2.3 COGNET Model of Air ASW Mission Management

Earlier phases of this research applied the COGNET modeling formalisms to the vehicle tracking problems based on Naval Air ASW. The emphasis in this model was on responsibilities of the TACCO that were termed mission management. This term encapsulates the TACCO's tactical role in the later (i.e., post search) portions of the Air ASW problem. These are the portions which involve a protracted period of real-time multi-tasking on the part of the TACCO.

The TACCO actually begins to build him mental model (or in COGNET terms, to populate a blackboard representation) of the mission as early as the preflight briefing. The mission management model, however, concerns problem evolution only after an initial contact with a target is obtained through the search strategies. Once the first sensor contact is gained, then the TACCO must begin a protracted prosecution of the contact that is in part goal-driven (based on training, experience, and standard doctrine) and in part data-driven (based on the specific sensor data received). The overall goal of this prosecution is to form a highly accurate hypothesis as to the location, depth, course and speed of the target submarine. "Highly accurate" is operationally defined as sufficiently accurate to launch against the target with a standard (torpedo) weapon. In peacetime, of course, the TACCO does not launch an attack but merely attempts to maintain this level of knowledge and track the submarine. Mission management, covers a period of persistent RTMT activity, that in practial terms ranges from 30 minutes to two hours.

2.3.1 Task Decomposition

The mission management COGNET model was developed using an experimental paradigm and data analysis methodology described in Zachary,et. al. (1989). A total of fourteen information processing activities were identified as individual cognitive tasks in the basic COGNET network. These tasks can be grouped conveniently into related areas as follows:

- maintaining a complete picture of the tactical situation,
- managing control of the aircraft,

- hypothesizing/inferring the activities of the target, and
- managing the patterns of sonobuoys deployed.

Table 2-1 lists the fourteen tasks according to these groups.

Control Aircraft	**Control Sensor Suite**
Position in Area of Interest	Manage Sonobuoy Resources
Preposition for Expected Events	Broaden Initial Contact
Maneuver for MAD	Investigate Convergence Zones
	Expand Pattern for Contact Continuity
	Deploy Sensor/Pattern
Maintain Situational Awareness	**Hypothesize Target Activity**
Plot Acoustic Environmental Features	Identify Area of Interest
Review Overall Situation	Develop Target Fix
	Gain Attack Criteria
	Determine Target Track

Table 2-1 Individual Tasks in COGNET ASW Mission Management Model

2.3.2 Blackboard Organization

As described in Subsection 2.1 above, the integrative problem representation is formalized in COGNET as a blackboard structure. In general, a blackboard structure may contain separate partitions, or panels, to deal with information about different aspects of the problem. Each panel is decomposed hierarchically into levels containing one or more hypotheses or partial solutions constructed by the individual tasks. The ASW mission management blackboard structure contains two panels--the target panel and the situation panel -- each representing separate yet highly interrelated aspects of the problem solution space. The target panel contains information about the TACCO's evolving hypotheses about target behavior; the situation panel contains an understanding of the evolving prosecution. These two solution aspects are constructed separately, but draw on each other as sources of data.

The target panel is divided into six levels of abstraction which are used by the TACCO to process sensor data about a possible target. Each level represents increasingly more refined hypotheses about target behavior.

The lowest level is the contact level. It contains hypotheses and/or perceptual events denoting sensor contact. When a contact 'appears' as a display symbol on the TACCOs tactical display, a perceptual demon is immediately triggered which posts the information on the contact level.

The next level of abstraction involves the application of knowledge about the sensor which produced the contact, properties of the specific acoustic environment, and the overall situation to define areas of interest that may arise from sensor contacts and/or other, more abstract, information on the blackboard.

The third hierarchical level on the blackboard represents a special kind of area of interest, a direct path region around an acoustic sensor, MAD, or radar contact. Direct path information represents a maximally crude locational hypotheses about a target.

The fourth level refers to more precise locational hypotheses that are based on various combinations of other hypotheses, knowledge, and/or sensor data from other levels on the blackboard. Usually, it is necessary to fuse information from multiple contacts to obtain a location hypothesis.

The fifth level of the target blackboard refers to directional hypotheses about the target. These may be mixed levels of abstraction, from general directional information gleaned from prior knowledge to specific directional hypotheses inferred from sensor data on the blackboard.

Finally, the sixth and highest level on the target blackboard refers to fused directional and locational hypotheses, which are referred to as tracks. These hypotheses often correspond to moving track symbols generated by the TACCO via the workstation software. It is not uncommon for a TACCO to have five or more of these track hypotheses for a single target.

Figure 2-5 indicates the blackboard target panel organization, showing its contents as reconstructed from a specific experimental trial. The arrows show the general flow by which information is posted and transformed, indicating the mixed directions in which information is processed on the blackboard. Initially, there is only a weak directional hypothesis of an expected southwesterly motion of the target. This hypothesis would have been developed prior to take off, as the result of intelligence information in the pre-flight briefing. Because the pre-contact mission phases are not included in the present model, such hypotheses are dealt with as assumptions. That is, the model assumes that relevant information about the mission that would have been developed prior to the beginning of the mission management phase is already posted on the blackboard at the start of this phase. (This assumption is generally more important to the situation blackboard than to the target blackboard).

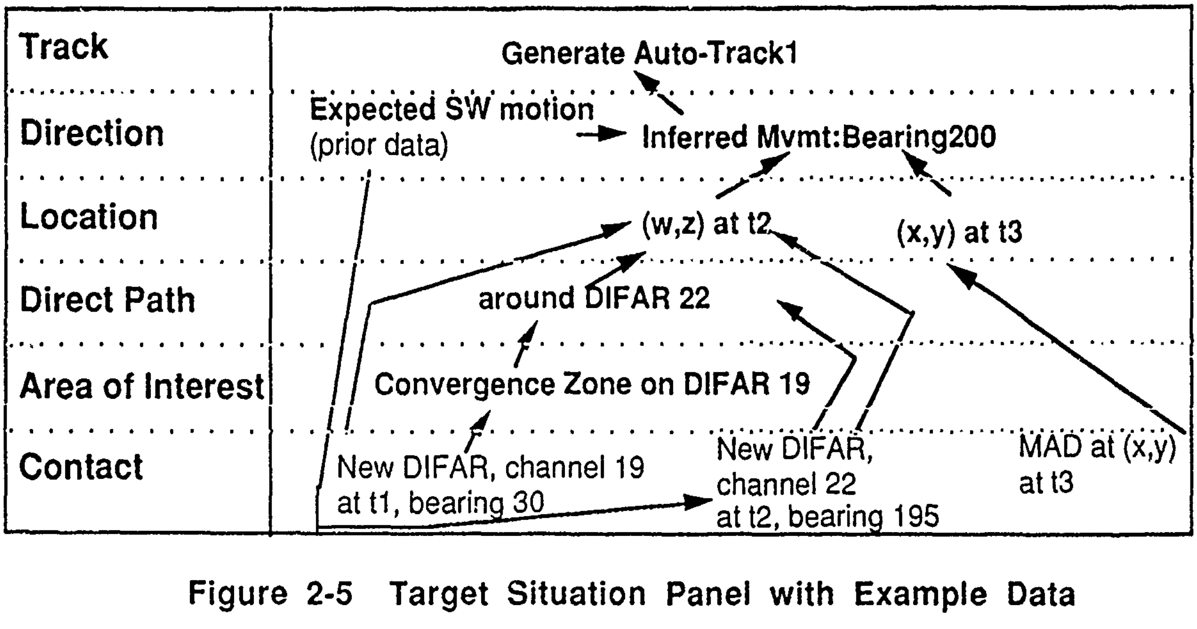


Figure 2-5 Target Situation Panel with Example Data

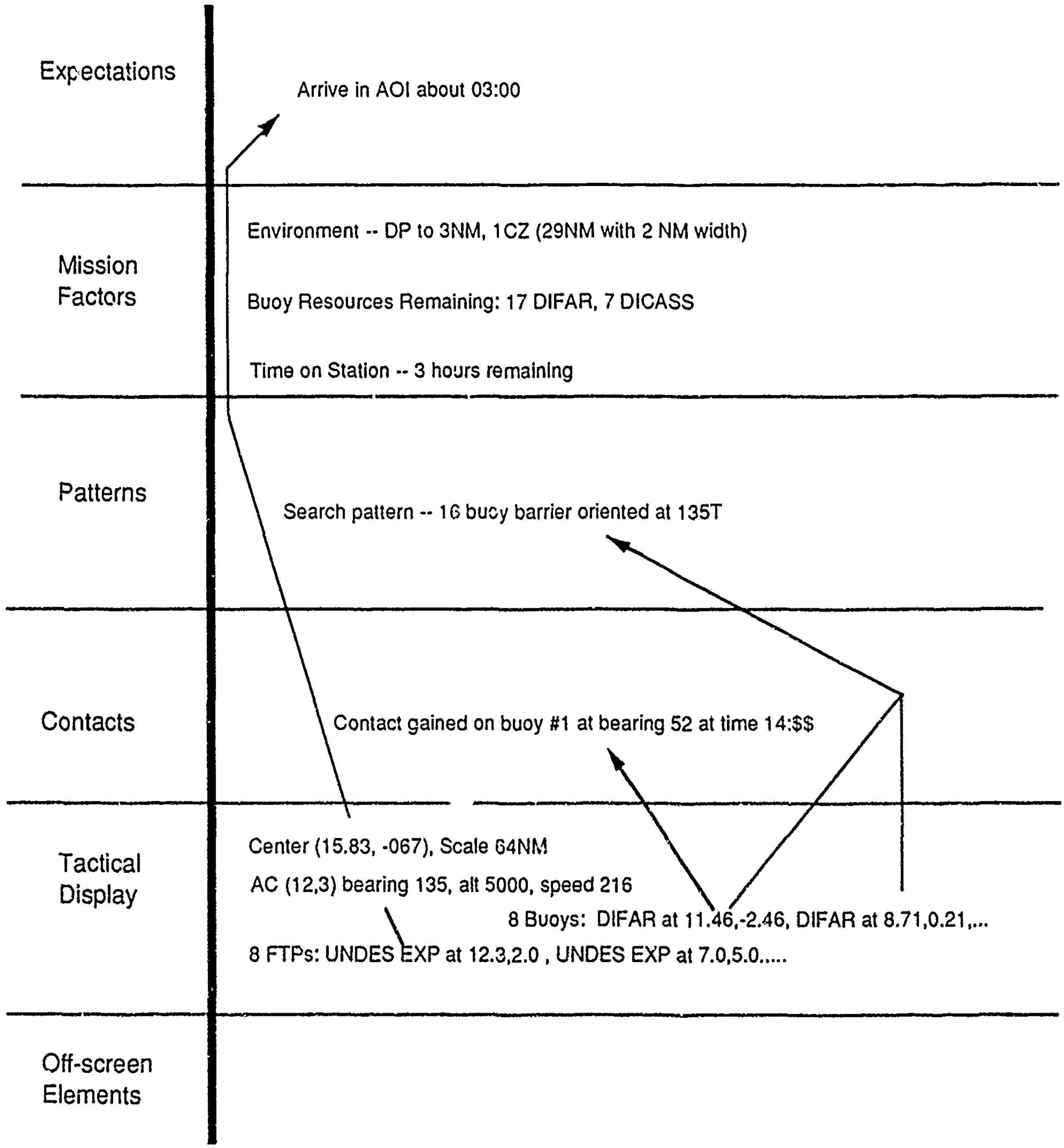
The example in Figure 2-5 begins with deployment of the initial search pattern. One sensor on Channel 19 from that pattern gains a directional contact. That contact is posted on the contact level of this blackboard panel as "New DIFAR on 19 at t1, bearing 30" ,indicating a new directional contact was first obtained on a DIFAR sensor using channel 19 at time t1, with the directional bearing at 30° to the sensor having the contact. The TACCO incorporates this information into a representation through a perceptual event, i.e., by perceiving the contact and associated bearing line as they appear on the tactical screen. Thus, this initial posting in the model is done via a perceptual demon.

The posting of a new contact on the blackboard triggers the Identify Areas of Interest (AOI) task which determines the possible areas of interest associated with the new contact and posts them as AOIs on the AOI level of the target panel. Only one of these is shown in Figure 2-5, the Convergence Zone Area of Interest that is associated with the sensor on Channel 19. The overall pattern of information on the blackboard at this time triggers two additional tasks. These are 1) the Broaden Initial Contact task, which deploys a sonobuoy pattern around the sensor on channel 19; and (2) the Investigate Convergence Zone task, which deploys a pattern of sensors in the Channel 19 CZ AOI. The layout of this pattern is influenced by the existing motion hypothesis already on the direction layer of the target panel. Because of this hypothesis, the Investigate Convergence Zone task maps out the pattern slightly to the southwest of where it would have been laid out (in a no-information case). One of these sonobuoys in the CZ pattern is deployed using Channel 22, and soon gains a contact at time t2. This new contact is also posted on the blackboard by a perceptual demon. The influence of the existing directional hypothesis on this contact is indicated by the arrow from the directional hypothesis to the contact message.

The new contact on Channel 22 again triggers the Identify AOI task, which this time implies that only a direct path contact is reasonable for the sensor using Channel 22. This information, in combination with the continuing CZ AOI on sensor 19, further leads to a direct path hypothesis being posted on the Direct Path layer of the target panel. After a short time, the TACCO makes an initial locational hypothesis about the target at the location of the intersecting bearing lines for sensors on channels 19 and 22. This locational hypothesis is denoted as (w,z) and time t2 on the blackboard.

The new pattern of information on the blackboard at this time triggers the Maneuver for MAD (Magnetic Anomaly Detector) task, through which the TACCO attempts to develop a more precise locational hypothesis via a combined DIFAR and MAD contact. Through this task, the aircraft does obtain a MAD contact, at location (x,y) and time t2, as indicated on the contact layer. This contact is then combined with the directional information and direct path hypothesis from the sensor on Channel 22, to yield a revised hypothesis that the target was near x,y at time t3. Moreover, this locational hypothesis is further combined with the previous directional hypothesis and the previous locational hypothesis (i.e., w,z at t2) to generate a refined directional hypothesis, i.e., that of movement along bearing 200°. This is also followed by creation of a moving track symbol on the screen, anchored at location x,y at time t2, and moving on course 200°.

The Situation panel of the blackboard contains information about the individual elements (e.g., the aircraft and sonobuoys) and features (e.g., environmental properties) of the tactical situation. It also is divided into six levels as shown in Figure



2-6. In this figure, as in Figure 2-6, the contents show data from a specific experimental trial.

Figure 2-6. Situation Blackboard Panel with Example Data

The first two levels of the situation panel, off-screen elements and tactical display, contain situational information that is represented on the TACCO workstation. Most of these data are plotted on the tactical screen, such as location of the aircraft all tactical display symbols. These display symbols include buoy locations, fly-to-points, contact symbology (e.g., bearing lines, MAD contact symbols), 'chalkboard' information such as reference circles, reference lines, and reference marks; and other symbology.

These two levels provide a means for representing the spatial relationships among elements over the complete tactical area. The tactical display level contains those elements which are visible on the TACCO's tactical display screen, and hence, are of immediate interest. The off-screen elements level contains all other elements over the whole mission area. Because the tactical display has a 'zoom/pan' organization, there is often symbology that is not currently on the screen. These data, represented as they were last viewed by the TACCO, are contained on the off-screen elements layer. In addition to the actual tactical display contents, the tactical display level indicates other items of information that are perceived from the workstation, such as the center point of the display and its scale, the aircraft bearing and speed, and other status information.

The third and fourth levels are contacts and patterns. The contact level contains a time-stamped history of the sensors that have gained and/or lost contact. If at some point the TACCO gets information conflicting with a current target hypothesis, that individual may refer to the contact history to re-evaluate what is known about the target.

The pattern level contains a record of all buoy patterns planned or in use. Different patterns are used for different mission phases and are selected based on the TACCO's current target hypothesis and various mission factors. An important aspect of the pattern level is that patterns are usually defined relative to other aspects of the situation. For example, a convergence zone investigative pattern reflects a standard geometry that is adjusted to reflect features of the acoustical environment and laid out relative to another sensor that has the contact being investigated. Thus, an entry on the pattern level will indicate pattern type (e.g., investigative, contact-broadening) and also status (e.g., planned, partially-in-water, fully-deployed, partially dead, dead). It will also contain links to other sensors or features on the tactical display layer (e.g., sensor locations) and the mission factors layer (e.g., environmental features).

The fifth level contains data/hypotheses about mission factors. These include environmental information about how sound will be propagated in the mission area and resources remaining. The TACCO's current hypothesis about the environment influences how that person interprets contacts. Normally, these hypotheses are developed prior to the mission management period at either the pre-flight briefing or the initial on-station phase where environmental sensors are deployed and analyzed. However, patterns of sensor contacts may conflict with posted environmental information, which may in some cases cause the TACCO to revise environmental hypotheses. Resources remaining influence what strategies will be used for target prosecution. For example, the number of buoys remaining influences the selection of buoy patterns to use.

The sixth level contains expectations about future events and when they are likely to occur. For example, when the TACCO enters a Fly-to-Point (FTP), a steering command to the pilot , he creates an expectation about when that FTP will be captured. This influences his decisions about what he can accomplish until that time. Such expectations can lead to suspensions of currently active tasks, as well as triggers for suspended tasks to recapture control.

2.3.3 Perceptual Demons

Effective mission management relies on the TACCO's ability to monitor the large amount of information on the tactical screen and perceive importnat events and/or information as they are displayed there, e.g.,observing a contact bearing come up on the screen. The perdeptual demon construct in COGNET accounts for this type of information access. In the ASW Mission Management model, the perceptual demons are linked with the kinds of visual events that the TACCO encounters. These are the display events that were programmed into the ASW Mission Management experimental environment (see, Zachary & Zubritzky, 1988). In an operational application, the perceptual demons would be based on the display and/or auditory information that could be presented at the operational crewstation. The list of the perceptual demons developed for the Mission Management model is shown in Table 2-2.

2.3.4 Cognitive Task Models

Detailed models were built of all fourteen tasks indicated in Table 2-1 above, using the COGNET task description language (Figure 2-3). Different tasks contain different mixtures of cognitive and motor/psychomotor (i.e., human-computer interaction) operators. Figure 2-7 shows an extreme case, the initial portion of the model for the task "Identify an Area of Interest". This model is triggered essentially any time there is a new sensor contact posted on the target panel of the blackboard, and reflects a cognitive process in which contact data are transformed into areas of interest for further examination. This task is triggered by the perception of a new sensor contact (posted on the target blackboard by a perceptual demon), and produces no externally observable actions. Instead, this task model captures an inferential process by which other blackboard information is applied to the contact datum to generate specific areas of interest. It is essential to the development of the blackboard representation and to any computer-human interface using the Mission Management model as an embedded user model.

In contrast to Figure 2-7, Figure 2-8 shows a portion of a task that involves a substantial amount of directly observable human-computer interaction, the "Broaden Initial Contact" task. This table shows the portions of the task that focus the display on the area in which the pattern is to be plotted and that draws the actual pattern on the screen as a series of fly-to-points, based on doctrine and operator training. It represents a segment of human-computer interaction that can is readily observable and recognizable from this type of model of the task. Zachary, et..al. (1989: Appendix) provide a complete listing of all individual task descriptions for the Mission Management model.

Buoy gains contact ==>
if DIFAR contact
POST : "New DIFAR Contact at time [mission-time] on Buoy [Channel No] with Bearing [bearing] and Strength [high/low] on Target/Contacts and Situation/Contacts"
if LOFAR contact
POST: "New LOFAR Contact at time [mission-time] on Buoy [Channel No] and Strength [high/low] on Target/Contacts and Situation/Contacts"
if DICASS contact
POST: "New DICASS Contact at time [mission time] on Buoy [Channel No] indicating range of [range in yards] and Bearing [bearing] on Target/Contacts and Situation/Contacts"
if CASS contact
POST: "New CASS Contact at time [mission time] on Buoy [Channel No] indicating range of [range in yards] on Target/Contacts and Situation/Contacts"
if MAD contact
POST "MAD Contact at time [mission time] at [x,y] on Target/Contacts and Situation/Contacts"
if RADAR contact
POST: " RADAR contact at time [mission time] at [x,y] on Target/Contacts and Situation/Contacts"
Buoy loses contact ==>
if DIFAR contact
POST: "DIFAR Contact on Buoy [Channel No] lost contact at time [mission-time] on Target/Contacts and Situation/Contacts"
if LOFAR contact
POST: "LOFAR Contact on Buoy [Channel No] lost contact at time [mission-time] on Target/Contacts and Situation/Contacts"
Bearing shift of DIFAR contact ==>
POST: "DIFAR Contact on Buoy [Channel No] bearing shift at time [mission time] to bearing [bearing] on Target/Contacts and Situation/Contacts"
Change in signal strength ==>
if DIFAR contact
POST: "DIFAR Contact on Buoy [Channel No] changed strength to [high/low] at time [mission time] on Target/Contacts and Situation/Contacts"
if LOFAR contact
POST: "LOFAR Contact on Buoy [Channel No] changed strength to [high/low] at time [mission time] on Target/Contacts and Situation/Contacts"
Capture of FTP ==>
if no expenditure
UNPOST: "[type] at [x,y]

Table 2-2. Perceptual Demons

if expenditure
TRANSFORM: "[type] at [x,y]" TO "[symbol type] at location [x,y] at time [time] on Situation/Tactical Display"

Change in aircraft location ==>
POST: "AC (x,y) bearing [bearing] alt [feet] speed [TAS] on Situation/Tactical Display

Symbol moved off-screen by recenter or downscale ==>
TRANSFORM: [type] at [x,y] on situation/tactical display" to "[type] at [x,y] at time [time] on Situation/Off-screen Elements"

Expenditure of buoy resources ==>
if DIFAR
TRANSFORM: "Buoy resources remaining: [number] DIFAR" to "Buoy resources remaining: [number -1] DIFAR"
if DICASS
TRANSFORM: "Buoy resources remaining: [number] DICASS" to "Buoy resources remaining: [number -1] DICASS"

Table 2-2. Perceptual Demons (contd)

GOAL: IDENTIFY AREA OF INTEREST...*ANY NEW [SENSOR TYPE] CONTACT POSTED ON CONTACT LEVEL OF TARGET BB*

GOAL: Identify Acoustic AOI...*if new DIFAR, LOFAR, CASS. or DICASS contact*

GOAL: Identify passive AOI...*if new DIFAR or LOFAR contact*

Determine [contact buoy number] from contact level of target BB
Post "DP AOI on 'contact buoy number' on AOI level of target BB
Post "BTmBn AOI on 'contact buoy number' on AOI level of target BB...*if Bottom Bounce on mission factors level of situation BB"*
Post "CZ1 AOI on 'contact buoy number' on target BB...*if 2CZ or 1CZ on mission factors level of situation BB*
Post "CZ2 AOI on 'contact buoy number' on target BB...*if 2CZ on mission factors level of situation BB*
Transform "New [buoy type] contact at time [mission-time] on Buoy [Channel No] with Bearing [bearing]" into "[buoy type] contact at time [mission-time] on Buoy [Channel No] with Bearing [bearing]"on contact level of target BB

GOAL: Identify active AOI...*if CASS or DICASS contact*

Determine [contact buoy number] from contact level of target BB
Post "CASS AOI with MDR [distance] on buoy [Channel No]" on AOI level of target BB...*if CASS contact*
Post "DICASS AOI with bearing [bearing] and with MDR [distance] on buoy [Channel No]" on AOI level of target BB...*if DICASS contact*
Transform "New [buoy type] contact at time [mission-time] on Buoy [Channel No] with Bearing [bearing]" into "[buoy type] contact at time [mission-time] on Buoy [Channel No] with Bearing [bearing]"on contact level of target BB

Figure 2-7. Portion of "Identify Area of Interest" Task Model

GOAL: Focus tactical display on AOI

<u>Perform</u> UPSCAL...*until AOI visible on display*
<u>Perform</u> RECENTER (contact buoy)...*if contact buoy not near center of screen*
<u>Perform</u> DOWNSCAL...*until AOI fills display*

GOAL: Build prudential pattern

<u>Subrogate</u> to "Plot Acoustic Environmental Features"...*if DP not already plotted*
GOAL: Enter FTPs
<u>Hook 1st point on DP circle</u>
<u>Perform UNDES FTP (expendable, hooked location)</u>
<u>Hook 2nd point on DP circle</u>
<u>Perform UNDES FTP (expendable, hooked location)</u>
<u>Hook 3rd point on DP circle</u>
<u>Perform UNDES FTP (expendable, hooked location)</u>
<u>Hook 4th point on DP circle</u>
<u>Perform UNDES FTP (expendable, hooked location)</u>

Figure 2-8. Portion of "Broaden Initial Contact" Task Model

3. MODEL ANALYSIS AND VALIDATION

The value of the COGNET framework as a methodology for modeling RTMT domains lies ultimately in its utility for developing more adaptive and intelligent human-computer interfaces. A necessary, though not sufficient, condition for utility is the validity of the models the COGNET methodology generates. Validation of the COGNET model of air ASW mission management involved evaluating the model's predictive power. In other words, for a given operator and mission scenario does the model allow prediction of what tasks should be undertaken next and when attention shifts should occur. A related, but somewhat different question is whether the model implemented as computer code allows this prediction. This is the focus that was taken here.

This section describes the analysis and validation methodology and the results of the validiation analysis applied first to the data from the subjects who were used to construct the model (post-hoc verification), and subsequently to the data from additional subjects (validation analysis).

3.1 Analysis and Validation Methodology

The model described in Subsection 2.3 above was developed directly from experimental data on human performance in air ASW (see Zachary et al., 1989, for a complete description of the modeling methodology). Using an experimental environment, data were collected from experienced TACCOs solving a variety of problems using different mission scenarios. Five problems which differed on the basis of mission objectives, target behavior, environmental conditions of the ocean and sensor capabilities were used. These problems were chosen in order to view a range of possible TACCO strategies. Five experienced TACCOs participated in the study solving one practice problem and from one to four experimental problems each. Data were collected on a total of sixteen trials; however, the first problem each TACCO solved was considered practice and was not included, leaving eleven problems which were used in actual model development.

The data collected for each problem were used both for model construction and subsequently for model verification. The experimental data recorded during each trial included a time-stamped record of all TACCO actions (key presses and mouse inputs on the graphical screen), and a symbol data file which recorded the state of each tactical symbol displayed on the screen at the occurrence of every TACCO action or any change in the tactical situation. On average, each trial generated an average of 400 actions and 40,000 display items. The action data file was transformed into a narrative timeline for ease of use during model construction and analysis. In addition, protocol data for each problem was recorded by replaying the completed problem (using the saved action data file as input to the problem environment simulation) with the TACCO, and asking for a description of his/her thoughts and intentions in performing each particular action or group of actions. Thus, for each trial three distinct but interrelated types of data were collected: 1) TACCO actions, 2) Situational context of TACCO actions, and 3) TACCO intentions while performing sets of actions. All three of the data sources described above were used for model construction (see Section 3.2 of Zachary et al., 1989, for a complete description of the model construction process).

The narrative timeline provided the basis for model validation. Using the timeline, the task sequence was reconstructed, identifying the point at which each task was initiated by the TACCO. Analysis and validation of the model involved a comparison of individual performance on each problem with that predicted by the COGNET model. COGNET model predictions were generated by a programmed version of the model.

The COGNET model of air ASW mission management was implemented as computer code in the project's experimental environment. The model code executed "on top of" the problem environment simulation and emulated TACCO workstation (see Sections 4 and 5 below). The implemented model included the full blackboard structure, all perceptual demons and five of the fourteen individual tasks. The blackboard contents were continually updated by perceptual demons viewing the interface and posting information (assuming the TACCO perceived every change in the display), and by POST and UNPOST operators in the task models. Attention triggering conditions for each task monitored the blackboard contents and "triggered" whenever the pattern on the blackboard matched. When the pattern no longer matched the triggering condition, the task was "untriggered." The time the task trigger was active was taken as the time when the model predicted that task.

The validation analysis was conducted for four tasks, which had been implemented in the programmed model:

1) Broaden Initial Contact (BIC)

2) Investigate Convergence Zones (CZI)

3) Manuever for MAD (MAD)

4) Expand Contact for Continuity (EXP)

The fifth task which was included in the implemented model, Identify Area of Interest, was not included in the validation analysis because it is a purely cognitive task, with no observable confirmation of its occurrence. The analysis was conducted for one hour of data for each trial beginning at the time of first contact. For each problem, the validation methodology was as follows:

1) Each instance that the TACCO initiated one of the four tasks listed above was identified.

2) The problem replay file was rerun through the programmed COGNET model to identify when each of the four tasks was triggered and untriggered (i.e., to determine when the model predicted the operator should shift attention to that task).

3) Each TACCO-initiated task was evaluated to determine whether it was started while the task trigger was active.

4) The time the TACCO initiated each task was compared to the time when the the modeled task trigger first became active.

Thus, two types of data were derived:

1) Task Occurrence Predictions: For each task instance performed by the subject, the model did or did not predict the task. If the model produced a corresponding task trigger, it indicated the model predicted the attention shift to that task. When the model did not produce a corresponding task trigger, it was taken to indicate that the attention shift was not predicted.

2) Task Prediction Lead: For those tasks that were predicted, a simple calculation yielded the amount of time by which the model prediction lead the actual task initiation.

3.2 Post-hoc Model Verification

The initial stage in model validation involved comparing the derived COGNET model of air ASW mission management with the performance of the individual TACCOs on each problem used to derive the model. The baseline data set included eleven trials (five subjects solving between one and four problems each). The post-hoc model verification also involved these eleven problems. A total of 30 instances of the four tasks were performed by the subjects. Of these 27, or 90%, were predicted by the COGNET model. That is, they were performed while the task trigger was active. Assuming that the probability of correctly predicting a single task initiation is .5 (i.e., 50/50), the probability of correctly predicting 27 of 30 tasks by chance is < .001. For the 27 predicted tasks the task trigger lead the actual task initiation by an average of 4:06 min (3:12 min when only the first occurrence of each task is considered). In subsequent paragraphs and accompanying figures and tables, the details of the results are presented. One additional task was predicted by the model just after the subject's initiation of the task (within 2 min). If this task is considered to be predicted by the model, the prediction accuracy of the model increases to 93% (28 of 30 tasks). However, the data presented in the following discussion takes the more conservative approach -- including only those tasks initiated during the exact time the task trigger was active.

Figures 3-1 through 3-11 show actual and predicted task performance on each of the eleven trials. Each figure shows the actual task initiations (subject actions) at the top and the model predictions (active triggers) at the bottom.

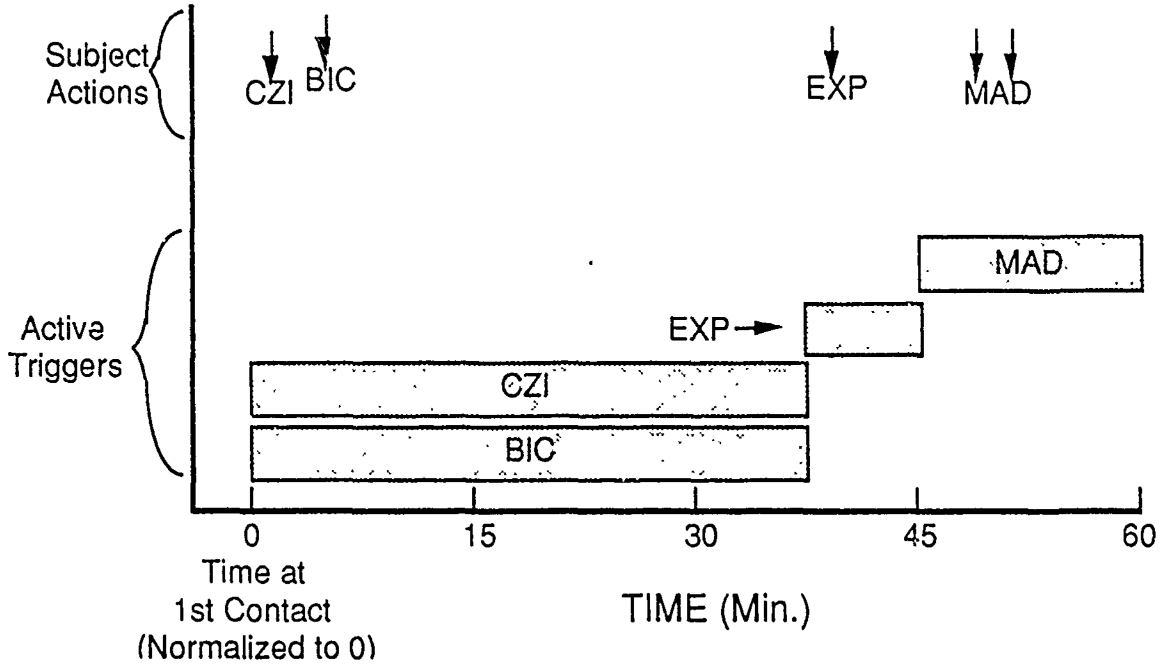


Figure 3-1 Subject 1 Problem 3

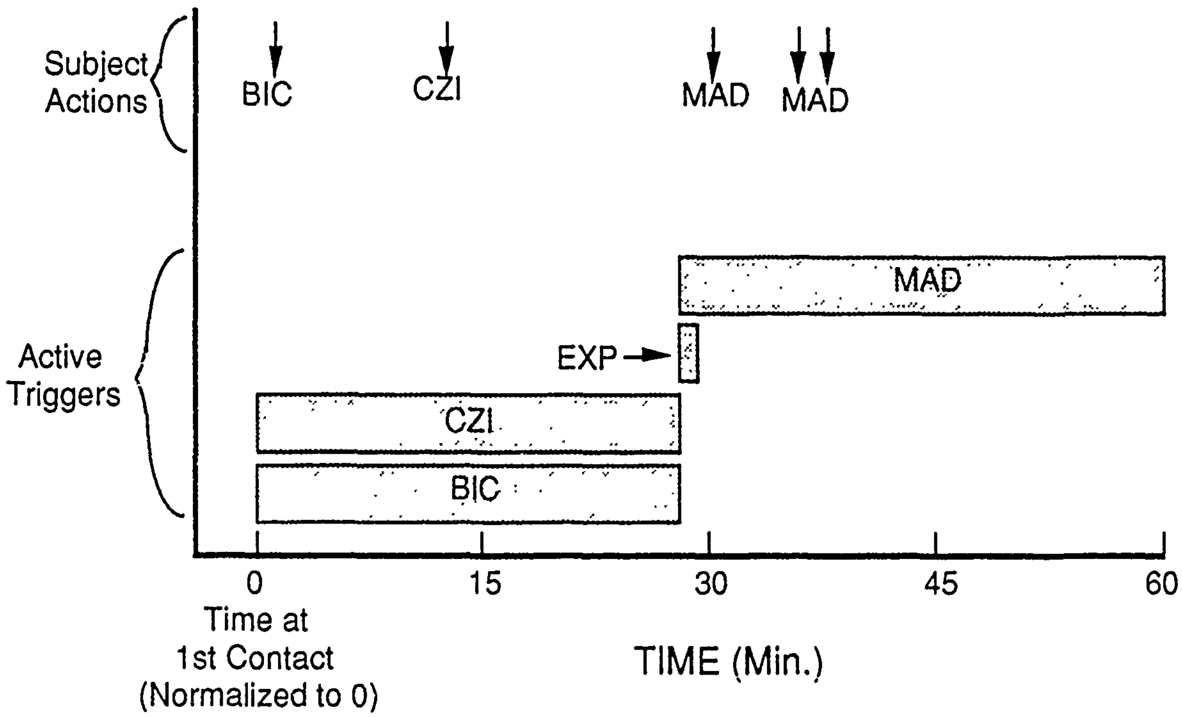


Figure 3-2 Subject 1 Problem 4

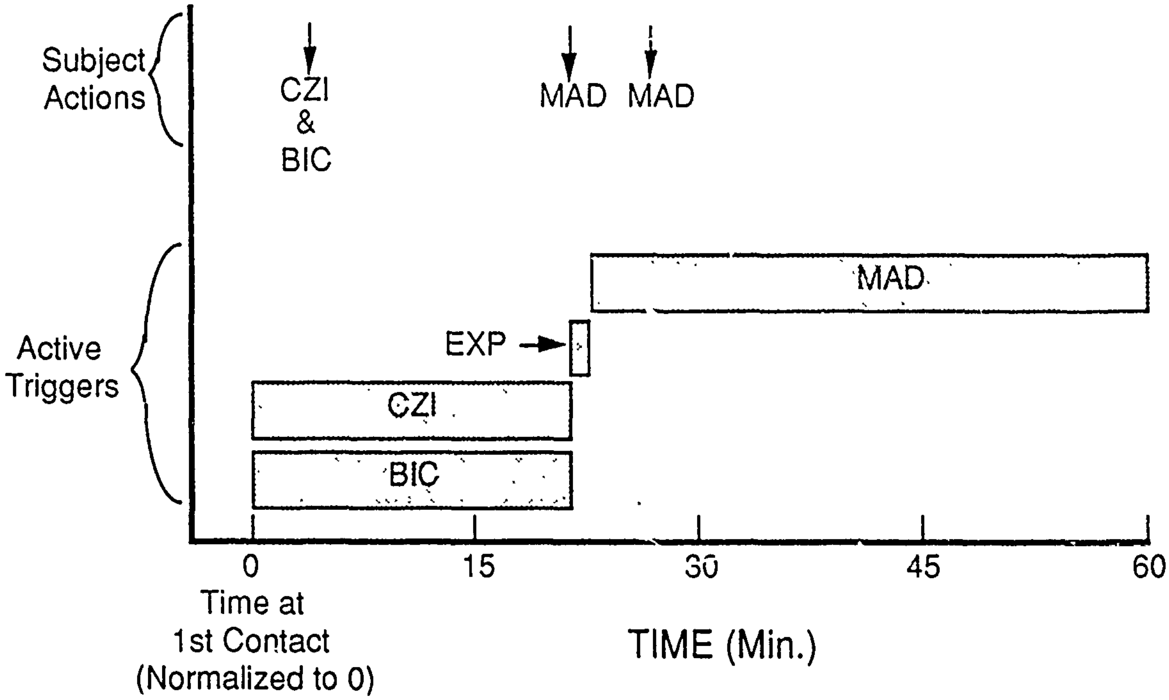


Figure 3-3 Subject 2 Problem 1

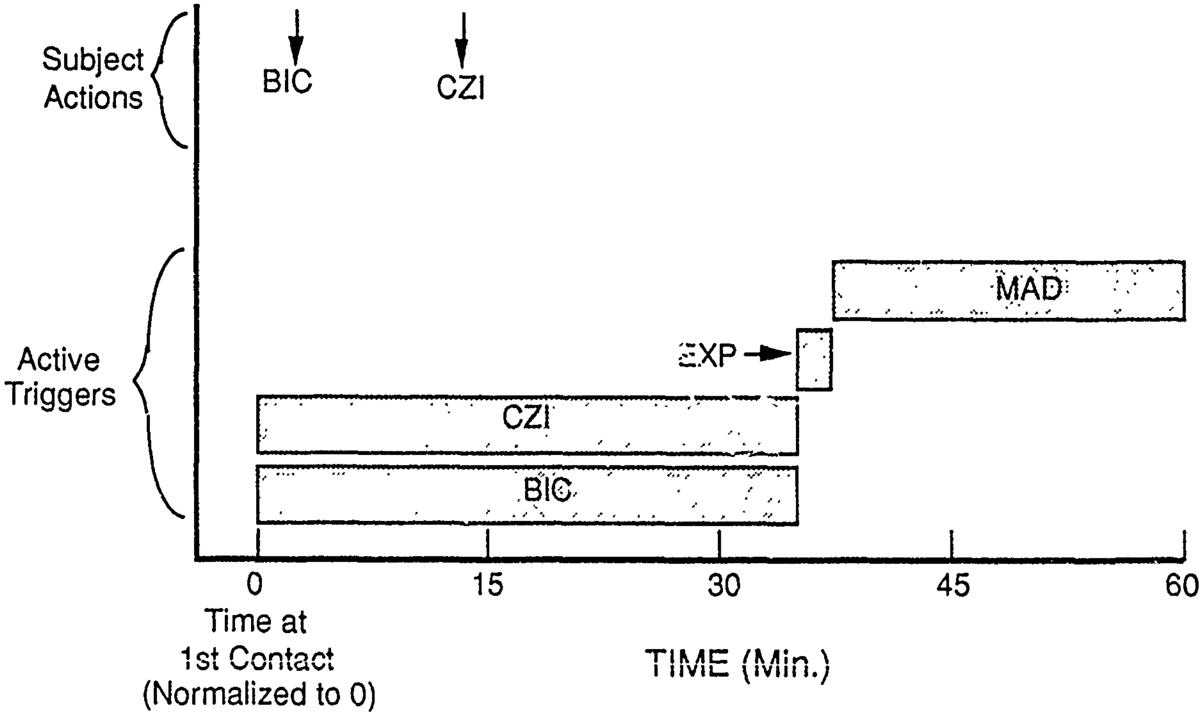


Figure 3-4 Subject 2 Problem 2

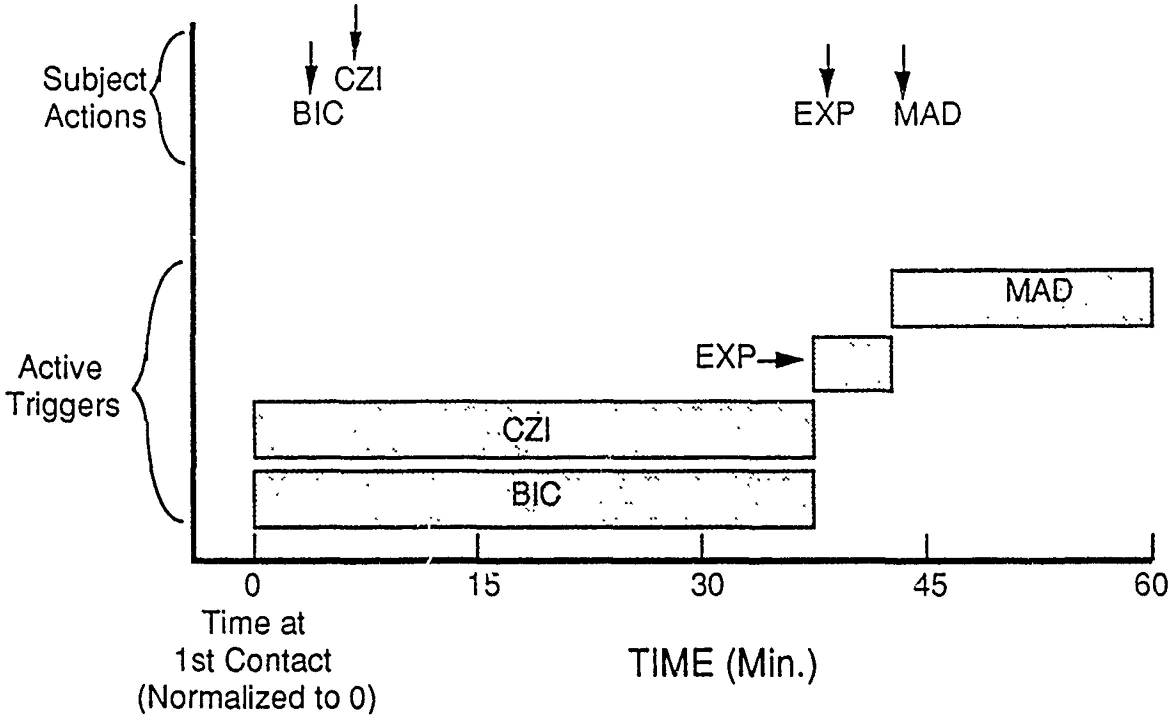


Figure 3-5 Subject 2 Problem 3

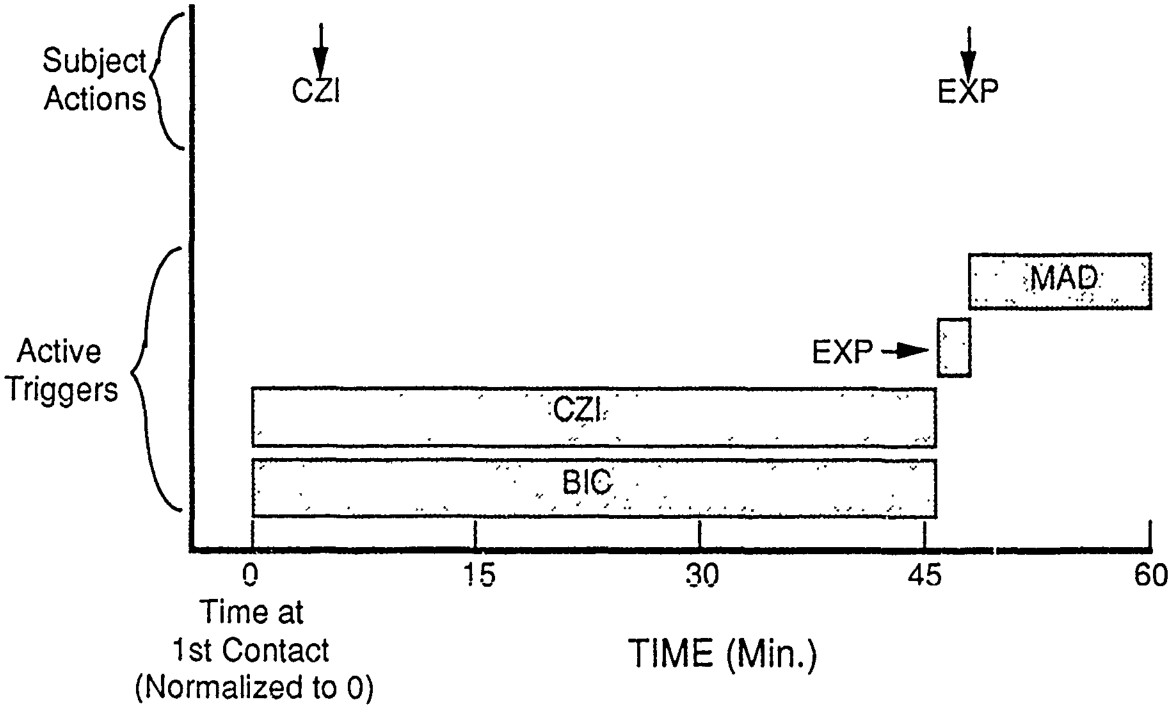


Figure 3-6 Subject 3 Problem 3

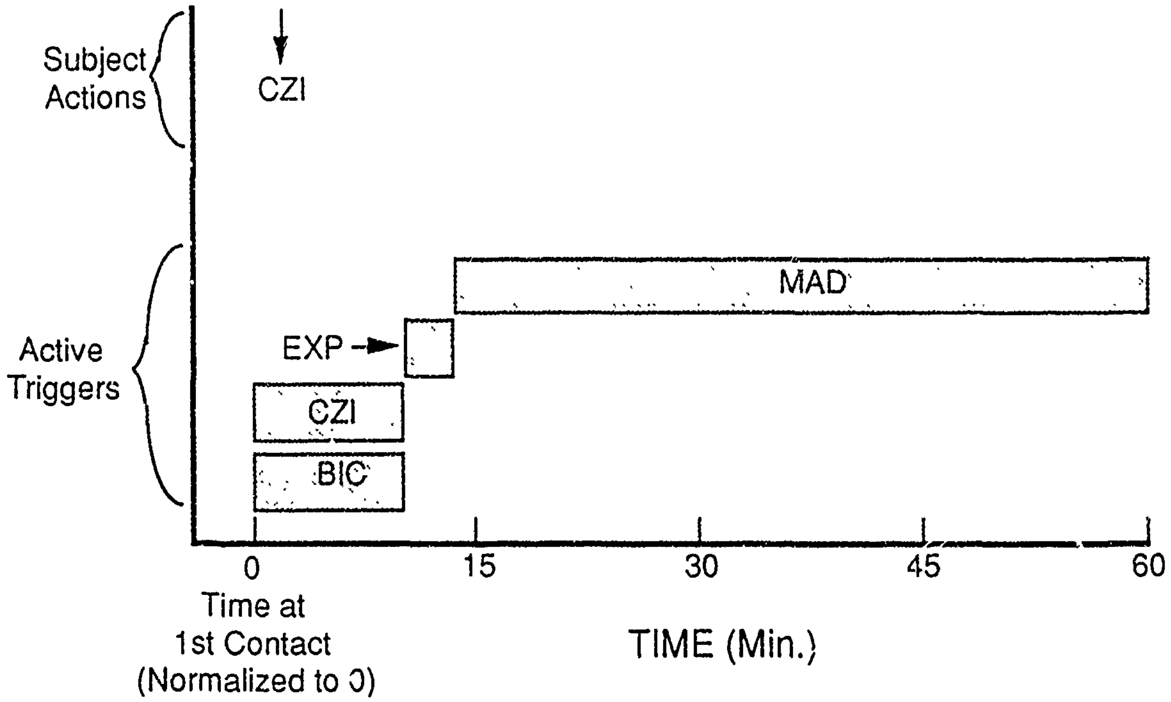


Figure 3-7 Subject 4 Problem 1

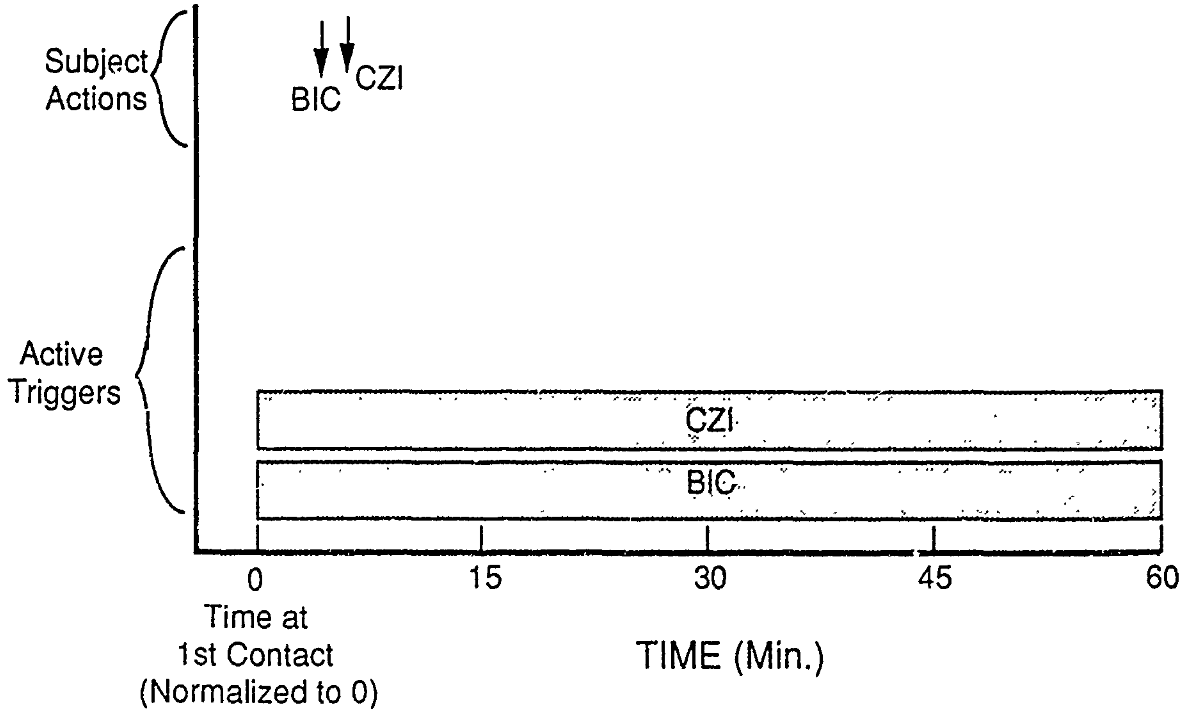


Figure 3-8 Subject 4 Problem 3

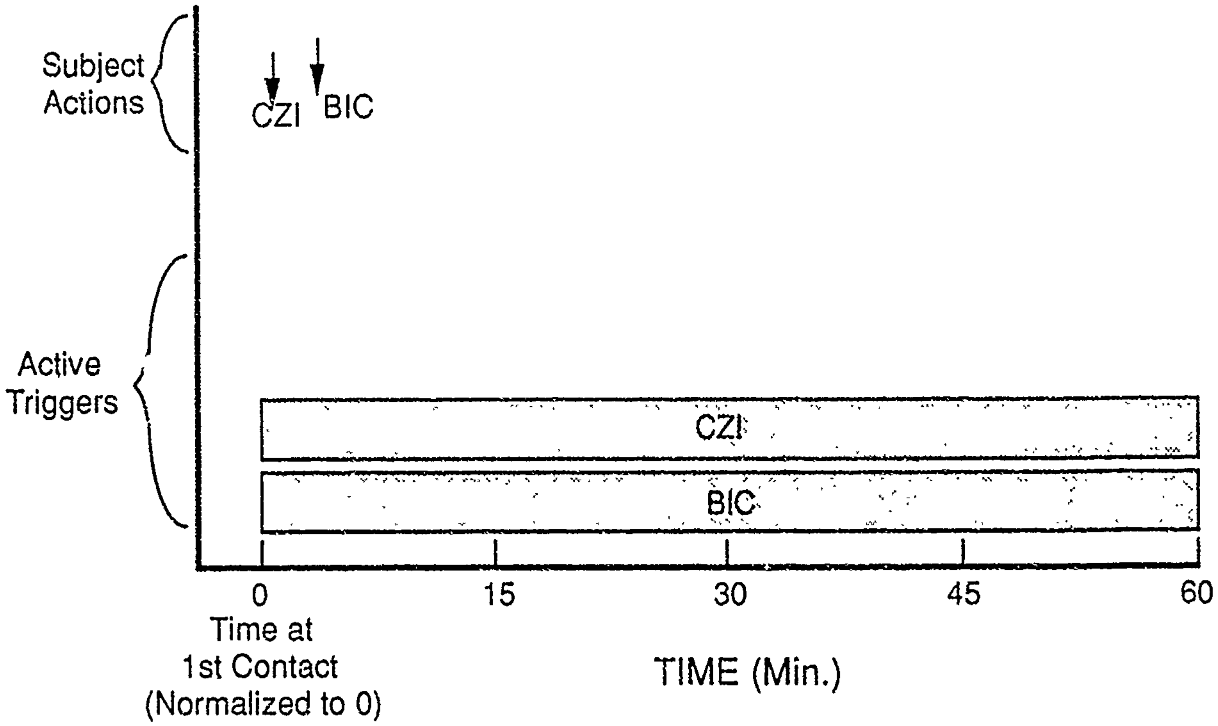


Figure 3-9 Subject 2 Problem 4

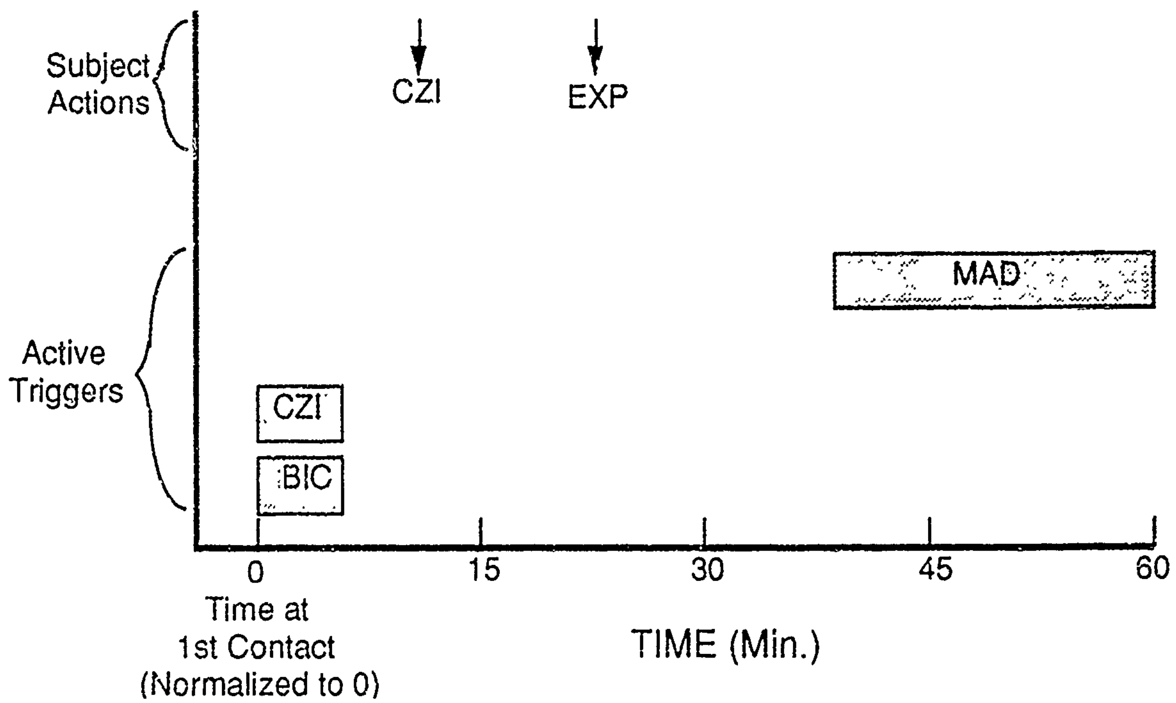


Figure 3-10 Subject 5 Problem 1

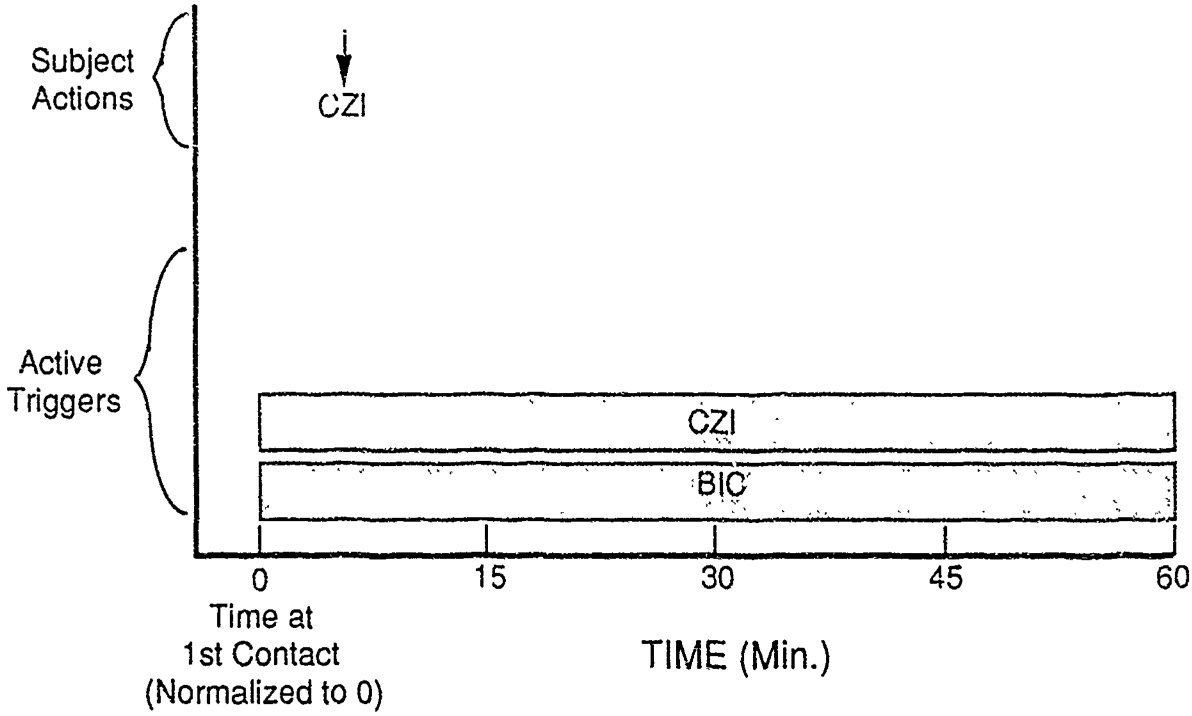


Figure 3-11 Subject 5 Problem 3

Table 3-1 provides a tabular summary of the eleven trials. Three additional data summaries are provided to allow inspection of the contribution of different variables to the model's predictiveness. Table 3-2 summarizes the data for each subject; Table 33 for each problem; and Table 3-4 for each task. Each table includes the number of tasks performed and predicted, a simple calculation to yield the % of tasks predicted, and the mean prediction lead time. The mean prediction lead was calculated using only predicted tasks. In some cases subjects performed more than one instance of a task (e.g., two or three MAD runs); for those cases a second calculation, shown in perentheses, included only the first occurrence of each task. The prediction lead is obviously going to be longer for the second or later occurrence of a task. Thus, the second number is probably more representative of the model's performance. Examination of Table 3-4 indicates that the first MAD run follows the model prediction by an average of 1:05 min, while for all MAD runs the average is 4:38 min after the model initially predicts a possible attention shift.

Table 3-1. Post-hoc Verification Trial Data

Subj # - Prob #	Number of tasks performed	Number of tasks predicted	% of tasks predicted	Mean prediction lead (for predicted tasks)	
Subj 1 - Prob 3	1	5	100%	3:39	(2:56)
1 4	5	5	100%	6:27	(5:19)
2 1	4	3	75%	2:50	(2:06)
2 2	2	2	100%	6:37	
2 3	4	4	100%	2:44	
2 4	2	2	100%	4:48	
3 3	2	2	100%	2:44	
4 1	1	1	100%	0:50	
4 3	2	2	100%	4:01	
5 1	2	0	0%	-	
5 3	1	1	100%	3:27	
Total	30	27	90%	4:06	(3:12)

Table 3-2. Post-hoc Verification Subject Data

Subj #	Number of tasks performed	Number of tasks predicted	% of tasks predicted	Mean prediction lead (for predicted tasks)	
1	10	10	100%	5:03	(3:57)
2	12	11	92%	3:51	(3:48)
3	2	2	100%	2:44	
4	3	3	100%	2:57	
5	3	1	33%	3:27	

Table 3-3. Post-hoc Verification Problem Data

Prob #	Number of tasks performed	Number of tasks predicted	% of tasks predicted	Mean prediction lead (for predicted tasks)	
1	7	4	57%	2:20	(1:40)
2	2	2	100%	6:37	
3	14	14	100%	3:18	(3:03)
4	7	7	100%	5:59	(5:06)

Table 3-4. Post-hoc Verification Task Data

Tasks	Number of tasks performed	Number of tasks predicted	% of tasks predicted	Mean prediction lead (for predicted tasks)
BIC	7	7	100%	2:44
CZI	11	10	91%	5:29
EXP	4	3	75%	1:21
MAD	8	7	88%	4:38
1st MAD	4	3	75%	1:05

For nine of the eleven trials, all tasks performed were predicted by the model. The three unpredicted task occurrences were for two of five subjects and all on one problem. Of the three tasks that were not predicted, one was performed 1:58 min prior to the task trigger (Subj 2-Prob 1); one was performed 4:05 min after the task was "untriggered" (Subj 5-Prob 1); and one was never predicted (Subj 5-Prob 1). In the first case, the subject shifted attention to 'Expand Contact for Continuity' prior to the model prediction, indicating the model actually predicted the attention shift, but just after the TACCO performed it. Subject 5 performing Problem 1 took a long time to analyze the situation prior to beginning prosecution, and by the time he began, he lost the initial contact and never regained control. Thus, the model was not able to predict attention shifts. This suggests elaboration of the model to cover cases in which contact is lost for long enough that a complete re-evaluation of the situation must be undertaken. As it stands now, the model begins from the point of initial contact.

As Table 3-2 indicates, there is some variability in the model's 'goodness' across different subjects. This may, in part, be due to differences in the TACCOs' ability and/or expereince. Subject 5, the subject for whom the model was least able to predict performance,ß had the fewest hours of operational experience of the five subjects and the longest period since active TACCO duties. Those TACCOs for whom the % of tasks predicted is high but the mean prediction lead time is longer, are probably the ones who would benefit the most from an adaptive interface that provided alerts that a particular task is appropriate. Variability in % of predicted attention shifts and prediction lead time across problems (Table 3-3) most likely reflects problem difficulty, amount of deviation from standard situations or tactics, or problems in which multiple tasks are appropriate simultaneously.

Table 3-4 indicates that the model was able to predict attention shifts to all four tasks. Although the % of tasks predicted varies from 75% to 100%, the variability is due to the number of task performances (i.e., there is one failure on each of three tasks). The prediction lead time varies among tasks, with the longest lead for 'Investigate Convergence Zones.' This is probably to to the fact that when both BIC and CZI are appropriate, the doctrinally correct approach is to perform BIC first.

3.3 Independent Model Validation

Further validation of the COGNET model of air ASW mission management involved comparison of the programmed model with the performance of new subjects, whose data was not used to construct the model. The validation methodology is the same as that used in the post-hoc verification. However, in these cases the problem timeline had to be analyzed to identify the task sequence and initiation of each task prior to the validation analysis. Subsequent to the timeline task decomposition, the validation methodology described in Section 3.1 above was followed.

The validation study included four subjects performing one or two problems each (following a practice trial each) for a total of six trials. Subjects 6 and 7 were new subjects; Subjects 8 and 9 had also served in the baseline data collection (Subjects 2 and 4, respectively). The problems used in the validation study included both new problems and problems that had been used in the baseline data collection. There was one instance of a repeat problem -- Subject 8 (old Subject 2) solved Problem 1 again.

The data presentation follows the same pattern as in the post-hoc verification. Actual and predicted performance is shown for each trial in Figures 3-12 through 3-17

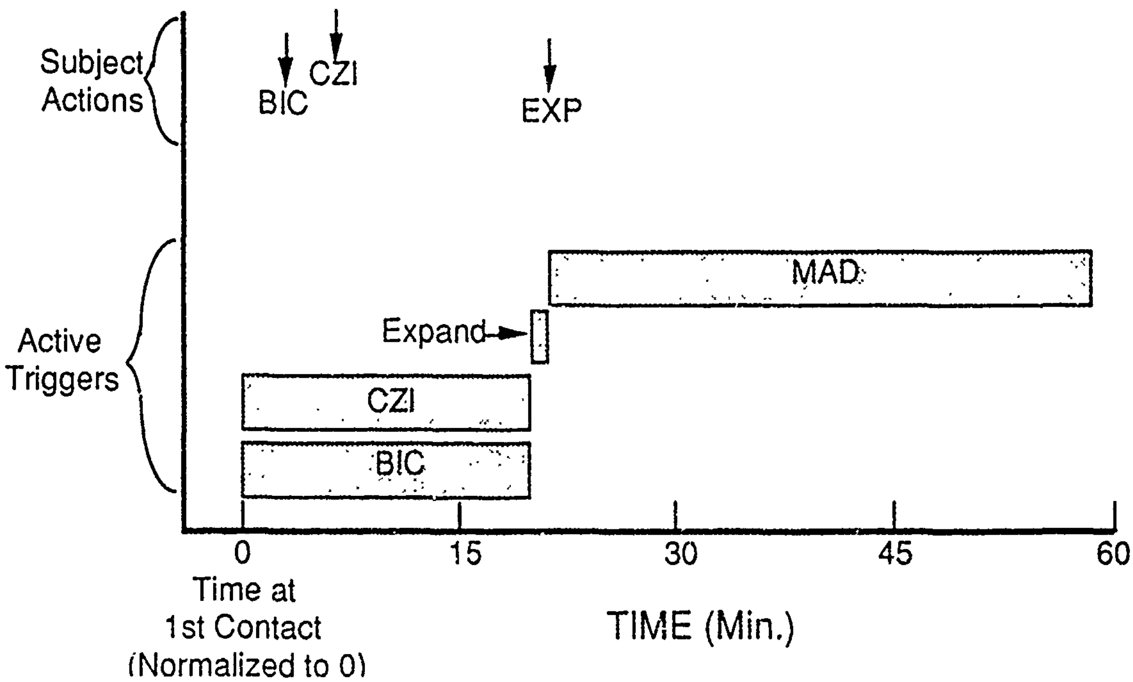


Figure 3-12 Subject 6 Problem 5

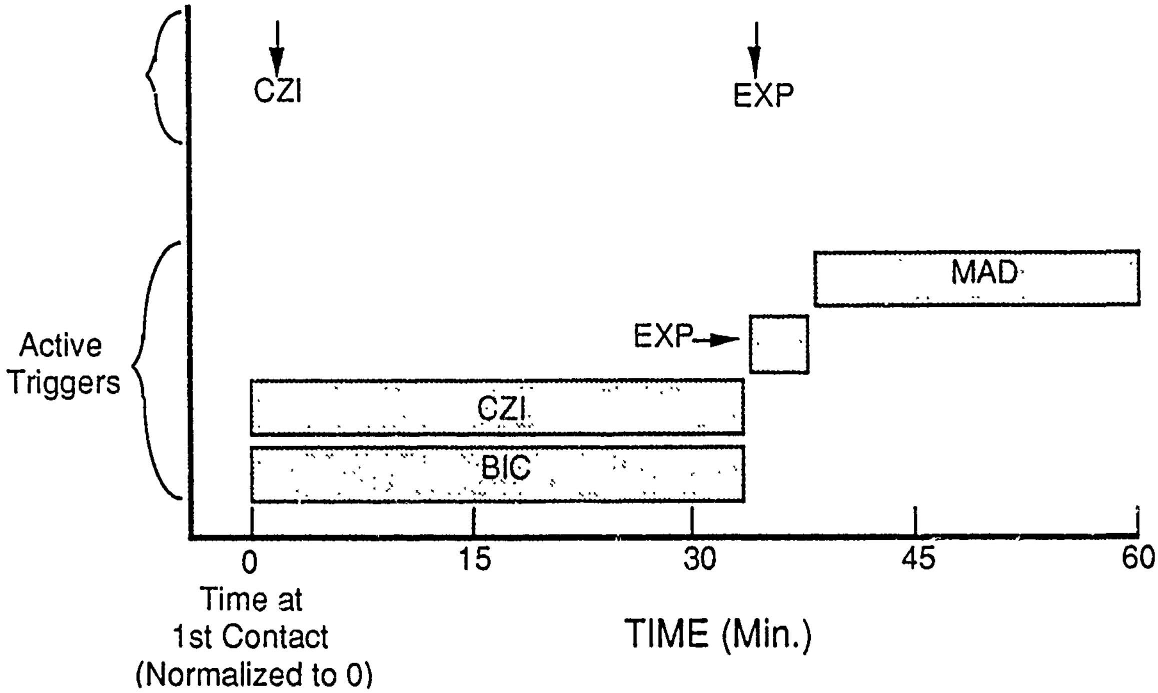


Figure 3-13 Subject 7 Problem 3

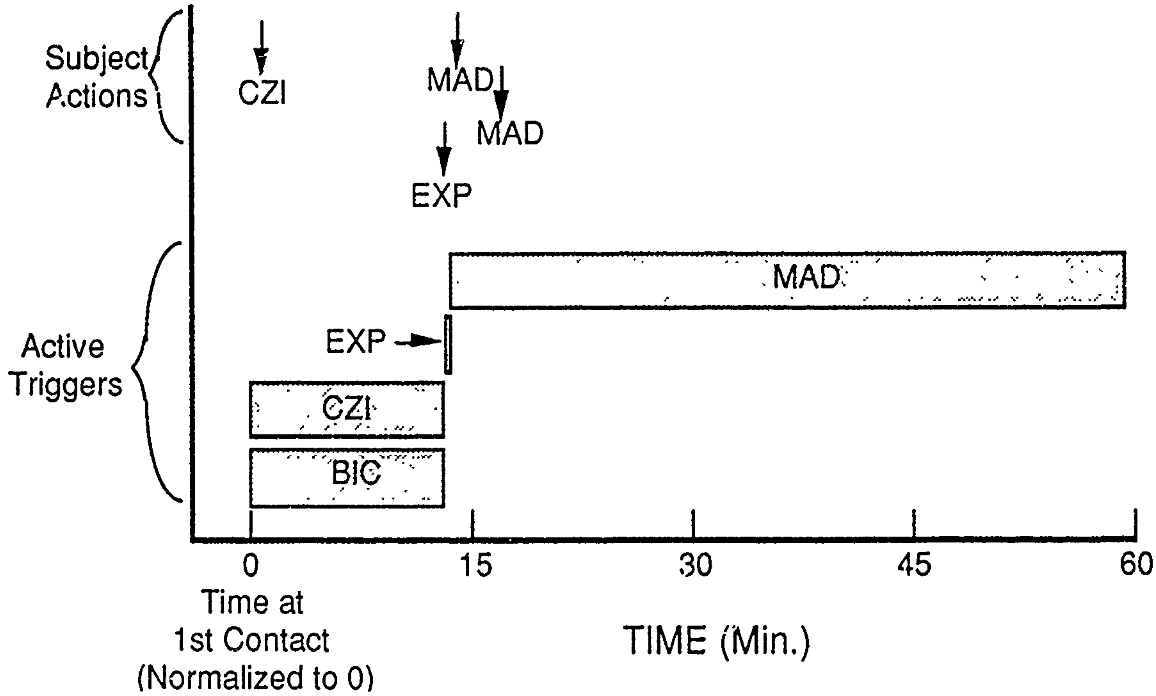


Figure 3-14 Subject 8 Problem 1

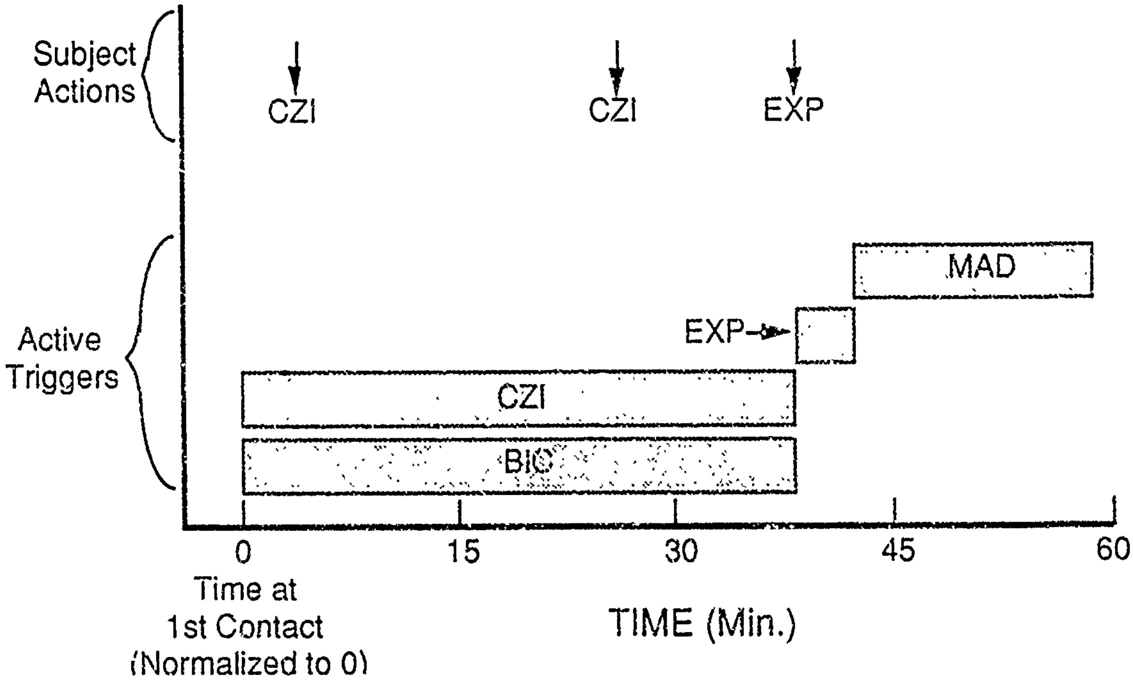


Figure 3-15 Subject 8 Problem 6

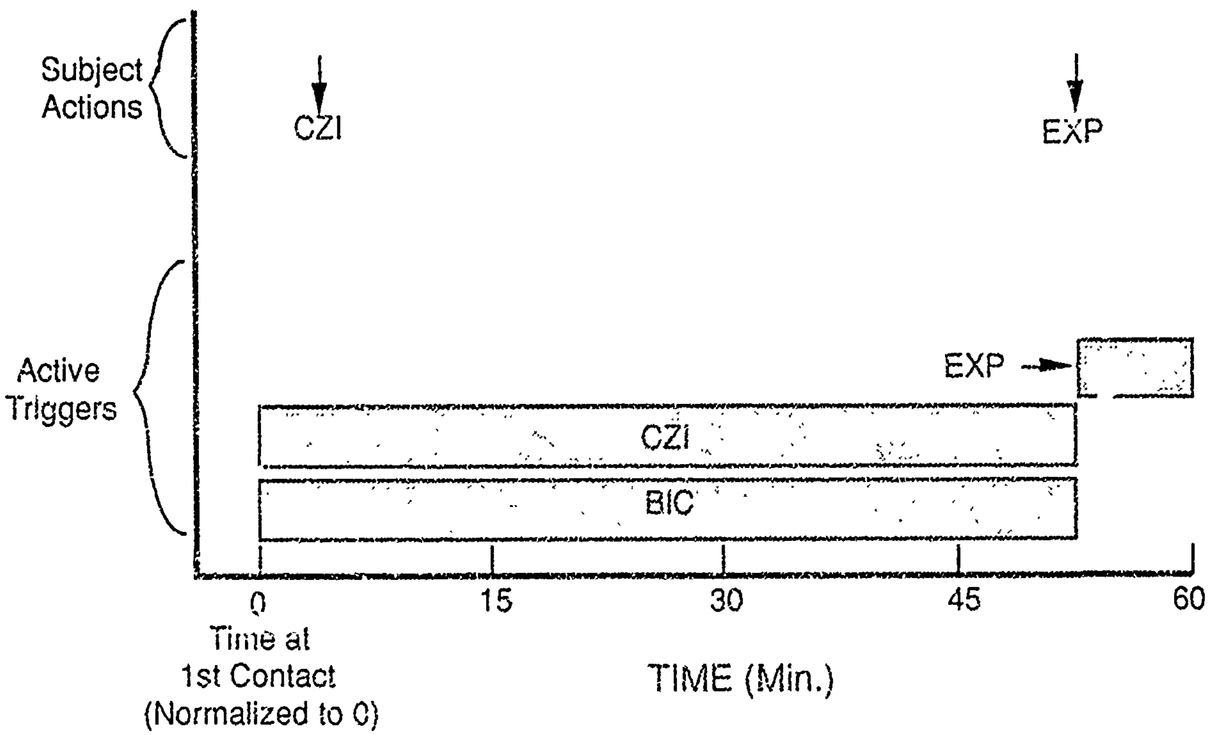


Figure 3-16 Subject 9 Problem 4

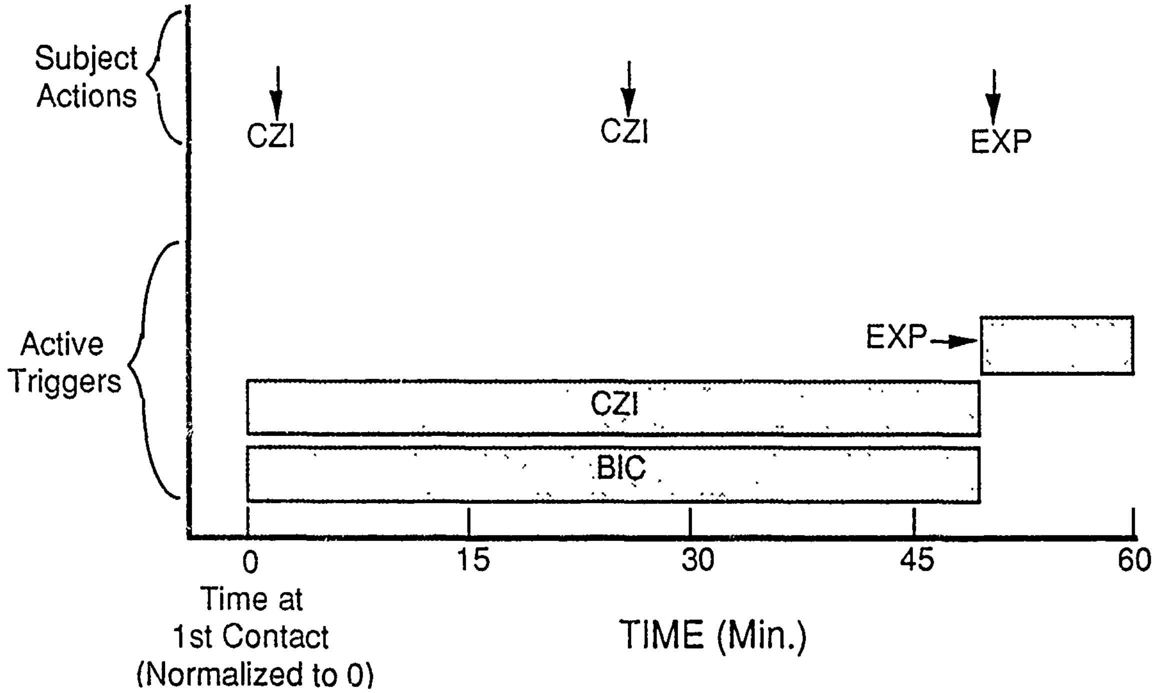


Figure 3-17 Subject 9 Problem 6

Table 3-5 provides a tabular summary of the six trials. Table 3-6 summarizes data by subject; Table 3-7 by problem; and Table 3-8 by task.

Table 3-5. Validation Trial Data

Subj # - Prob #	Number of tasks performed	Number of tasks predicted	% of tasks predicted	Mean prediction lead (for predicted tasks)
Subj 6 - Prob 5	3	3	100%	4:07
7 3	2	2	100%	1:09
8 1	4	3	75%	2:05
8 6	3	3	100%	10:42 (2:16)
9 4	2	2	100%	3:17
9 6	3	3	100%	10:10 (1:04)
Total	17	16	94%	5:38 (2:27)

Table 3-6. Validation Problem Data

Prob #	Number of tasks performed	Number of tasks predicted	% of tasks predicted	Mean prediction lead (for predicted tasks)
6	3	3	100%	4:07
7	5	5	100%	7:25 (2:35)
8	7	6	86%	6:24 (1:44)
9	2	2	100%	1:09

Table 3-7. Validation Subject Data

Subj #	Number of tasks performed	Number of tasks predicted	% of tasks predicted	Mean prediction lead (for predicted tasks)
1	4	3	75%	2:05 (1:12)
3	2	2	100%	1:09
4	2	2	100%	3:17
5	3	3	100%	4:07
6	6	6	100%	10:26 (2:04)

Table 3-8. Validation Task Data

Tasks	Number of tasks performed	Number of tasks predicted	% of tasks predicted	Mean prediction lead (for predicted tasks)
BIC	1	1	100%	3:14
CZI	8	8	100%	9:19
1st CZI	6	6	100%	3:22
EXP	6	5	83%	1:22
MAD	2	2	100%	2:44
1st MAD	1	1	100%	1:39

In the validation study, a total of 17 task instances were performed. Sixteen of these, or 94%, were predicted by the model ($p<.001$). The prediction lead time averaged 5:38 min for all task occurrences and 2:27 min for first task occurrences only. In this study only one task instance was not predicted -- it was an 'Expand Contact for Continuity' that was performed 0:28 min prior to the task being triggered by the model (Subj 8-Prob 1). By the less stringent criteria of including tasks performed within 2 min of the model triggering the task, all 17 task instances, or 100% were predicted.

The fact that model performance at least as good in the validation study as in the post-hoc verification analysis indicates that the model has generalizable predictive power. Examination of the cases in which the model did not predict an attention shift provides data for model elaboration to enhance its predictive power and prediction timeliness.

4. MODEL APPLICATION TO INTELLIGENT COMPUTER-HUMAN INTERACTION

This section discusses the approach and philosophy used to apply the COGNET mission management model to an adaptive interface for the ASW TACCO. General issues regarding the use of, and nomenclature for cognitive models in user interfaces are discussed first. The functionality desired for the adaptive mission management interface is discussed next, followed by the interface system architecture created to implement that functionality.

4.1 Model Based Human-Computer Interaction

The concept of using models of the human operator of a person-machine system to support the design process has been around for a long time. Traditional human factors has long used behavioral models to support system design (e.g. Siegel and Wolfe, 1969). For example, the main purpose of task analysis is to develop a model of the user and the user's requirements as a basis for system design and/or modification. The idea of building and using models of the operator's cognitive process as a basis for system design is of more recent origin. Within this context, designers attempt to model the user's internal representation of the system and its operation, or "mental model", in order to design the interface to support the user's cognitive processes. It has come into increasing importance as human operator roles in systems has moved from 'inner loop' physical control to 'outer loop' supervisory control (see Sheridan and Johansen, 1976; Van Cott, 1985). Cognitive model-based design approaches have been constructed and successfully applied by Rassmussen (1986), Zachary (1986,1988), Rouse (1981), Woods and Hollnagle (1986) and others. A previous report in this project (Zachary et al., 1989) discusses this issue in more detail.

The use of mental models in interface design has suffered from a history of confusing terminology and conflicting claims. As noted in and Wilson and Rutherford (1989) and Dehdashdi (1989) , the concept of mental model has been used indiscriminately to refer to:

- the user's model of system,
- the system's model of user,
- the designer's model of user, and even occasionally to
- designer's model of the user's model of system.

In this research, careful nomenclature is particularly important. The most widely accepted terminological framework for mental models in system design was put forth by Norman (1983). He defined a "mental model", in a person-machine system context, as the user's model of the 'target' machine system. In contrast, he used the term "conceptual model" to refer to the designer's representation of the 'target' system. Wilson and Rutherford (1989) added some terms to this definition to bridge the designer's and user's models. To complement Norman's conceptual model, they added the notion of a "designers conceptual model" as the designer's representation of the target system's user. They then distinguished the "user's conceptual model" from the "user's mental model the user's conceptual representation of the system as defined in arbitrary terms.corresponds to the "user's conceptual model". The "user's mental model" described as the user's internal representation of the system, defined in terms tied to psychological/cognitive theory. To this intricate set of distinctions we must add the idea of a "user model" as a designer's model of the user's mental model.

With these terms, the intended application of this research to interface design can now be clearly stated. A COGNET model such as the mission management model, provides a mental model of the user. It represents the knowledge and procedures the user employs to operate the system and to solve the specific domain problem. The model is expressed in terms tied to a specific general cognitive architecture (the formalized Pandemonium architecture) and notation. It is important to note that this COGNET architecture is based on the assumption that the user's internal representation of the evolving problem context is critical to the problem solving process. Thus, a main component of a COGNET model of the system user is the user's representation of the problem being solved. This adds a distinction not widely made in other literature on user's mental models, a distinction between the system aspect of the model and the problem aspect of the model. To make matters even more complex, the ultimate goal is to embed the COGNET model into a human-computer interface to allow the interface to interact more intelligently and adaptively with the human operator. Thus, we seek to translate the COGNET mental model of the user into a design-oriented user model. The embedded COGNET user model, therefore, becomes the computer's model of the human operator of the system. This includes a representation of the user's mental model of the problem being solved as well as a representation of the user's mental model of the system itself and the procedures for manipulating it.

There are two general applications of user models in human-computer interface design processes. The first and much more common one is as part of the analysis and design process. In this case, the user model forms the basis for critical aspects of the system design, ranging from control flow and data representation to functionality. The design approaches suggested by Zachary (1988) or Rassmussen (1986) are detailed examples of this use. In both methodologies, a model of the user's representation of the system is constructed first, and then analyzed to define feature of the system-under design and to guide critical design decisions. Similarly, Elkerton and Palmiter (1989) have used a user model of Hypercard programming (expressed in GOMS) to design an online help or training system as a way of enhancing the explanation or instruction capability of an interactive system.

A second and more ambitious use of user models is to physically embed them in the interface coupled with some additional reasoning apparatus. The interface can then reason about the actions, goals, and plans of the user and to support, enhance, and adapt the human-computer interaction to the cognitive processes of the human system user. There are specific functions which an embedded user-model-based interface could perform in this vein. Croft and his colleagues at the University of Massachusetts, for example (references) sought to employ an embedded model of the user's planning process as a way of correcting errors and inconsistencies in user performance.

Rouse *et al.* (1987) have used an embedded user model for intent inferencing in a pilot's associate context. In an intent inferencing concept, the interface infers the intent of the user and adapts the interaction to that inferred intent. User models for intent inferencing have been applied in other domains as well, including simulated sattelight tracking (Rubin et al, 1988), and computer-aided engineering (Finegold, 1984).

4.2 Adaptive Interaction Functionality for the Mission Management Domain

In practice, the use of an embedded model in an interface application is not simply an unconstrained design choice. The functionality possible from an embedded-user-model-enhanced interface is constrained by a tradeoff between three key factors -- technology, need, and implementation factors. Of these three, need is the most straightforward.

In any given RTMT domain, human operators exhibit different difficulties and problems. For example, in Croft's accounting domain, the amount of detail to be mentally managed by the user interacted with the complexity of the computer interface to lead to an increased number of errors and inconsistencies, many of which required remedial action. This, in turn, suggested to the researchers a need for interface support via an embedded user model. In other more real-time domains, operator inability to complete all desired or required tasks suggested a need for streamlined, more intelligent human-computer interaction via an embedded user model. Still other domains suggest other needs for intelligent interface support. It is critical to note that the need for an embedded user model at the interface is relative and dependent on specific features of the domain and operator population involved. The determination of need for intelligent interface support can be made on the basis of empirical analyses of person-machine system performance, by human factors task-analytic methods, or by some combination of expert opinion and case study. For the Air ASW mission management domain, the experimental problem solving data provided a good source of empirical information on the need for intelligent interaction.

Technology provides a different set of constraints on the application of an embedded user model. Different notations and model structures facilitate some kinds of operations and inhibit others. GOMS type models, for example, are useful in predicting performance in goal-directed interactions (John and Rosenblum, 1985), but fail to represent or support real time or multi-tasking domain features. In an ideal world, there would be a clear mapping of user modeling technologies to classes of needs (similar to that provided by Zachary, 1986, for decision support technologies), but at present no such classification exists. Thus, the technology used to model the system user limits the kinds of support that could be added to the interface through embedding the user model into an interface.

Finally, implementation issues will always be an important consideration. Some types of need may be met only with great cost and difficulty, while others can be achieved with much less effort. This issue is intertwined with the organization and structure of the baseline interface into which the user model (and associated functionality) is to be embedded.

A specific set of adaptive interaction functions for the Air ASW mission management interface was chosen against the backdrop of these tradeoffs. It should be noted that, as a research effort (albeit as an applied one), our goal was to demonstrate the application of COGNET to adaptive human-computer interaction. A parallel interest was to identify and address a real need of the Air ASW domain within the constraints of the implementation costs and the technology range of COGNET.

A complete analysis of operational deficiencies in Air ASW decision-making was clearly beyond the scope of this effort. Nonetheless, several issues surfaced repeatedly in the experimental data collection efforts. One was a delay in responding to task opportunities. Operators often needed several minutes to initiate a task after conditions were appropriate for its execution. These were usually clear during the verbal protocols taken during problem replays, when subjects would make such comments as "I could have started to make a MAD run [or some other task] here, but I guess I didn't realize that until later on." In other cases, operators failed altogether to recognize a tactical opportunity because they were focusing on only one or two hypotheses. (Schmidt and Goodson, 1990, show more detailed and compelling experimental data supporting this tendency to focus on only a small number of hypotheses in Air ASW). Taken together, these problems suggested one opportunity for use of the embedded model -- alerting the user to conditions where a specific task execution is appropriate. This use would address the first issue (delayed recognition) directly. Moreover, given the results of the attention flow validation cited above, (i.e., the anticipatory capability of the Air ASW model), the model could be reasonably expected to achieve this goal. This user model in the interface would also address the second issue, although indirectly. By alerting the TACCO to opportunities to perform a task, the interface would occasionally identify a task opportunity that the TACCO had missed, because the opportunity was triggered by a hypothesis that the TACCO was ignoring. Such an alert could force the TACCO to expand his decision strategy to include the hypotheses implied by the interface's recommendation. Task-level alerting was thus chosen as the first function of the model-based interface.

A second set of interrelated issues also arose from the experimental data collection. These dealt with certain kinds of errors and omissions made by the operators during the experiments. Many actions in the Air ASW domain must be tailored on the basis of some aspect of the operator's mental representation of the problem context. Most sonobuoy pattern deployments, for example, are laid out relative to some abstract anchor point such as primary contact buoy, or expected axis of target movement. All operators were observed to make errors in translating their mental representations into physical actions in the complex TACCO interface. That is, the operator may have 'known' that the primary contact buoy was buoy 14, but nonetheless based the pattern on the nearby buoy 15 (particularly when neither had contact at the time) by mistake. Such errors were particularly insidious, because they appeared to be correct, and resulted in patterns that had some limited value. This masked the original error lead the operator to propagate it forward. A seemingly different class of errors of omission resulted when operator's truncated tasks or ignored them because of time limitations. There were many cases where details of tasks were omitted to save time in a time-critical part of the missions, and where these omissions later came back to 'haunt' the future prosecution of the target.

Together these two issues suggested a second adaptive interface function, task aiding. That is, the interface would assist the TACCO in performing any task suggested by the alerting function outlined above. The detailed model of the task from the COGNET model would be retrieved and instantiated using the current blackboard contents. The user could then have the interface perform any part of the task in an automated or semi-automated mode, including the parts of the task which required reference to the current mental representation of the problem. To the extent that the blackboard contents are an accurate representation of the TACCO's mental model of the problem, the interface should be able to perform those representation-specific actions more accurately than the human TACCO, since it is invulnerable to those referential or translational errors which seem to give rise to the human 'slips.' Similarly, this function would allow the interface to speed execution of those tasks, thus also addressing the problem of time-constrained task performance.

These two functions represented areas of reasonable need for the problem domain and were within the technological range of the COGNET representation. They were also amenable to implementation within the scope of the current effort. The next section discussed the architecture and interface implementation used to realize these functions in an adaptive ASW Mission Management Interface.

4.3 COGNET-Based HCI Architecture

Accomplishing the functionality defined above required development of a novel and sophisticated architecture for the human computer interface. This architecture built layers and tools for embedded user models and adaptive interaction on top of the basic displays and controls components of the baseline interface in the Mission Management experimental environment (as described in Zachary and Zubritzky, 1988). The architecture is shown in Figure 4-1. Five separate parts of the COGNET model were extracted and programmed into distinct components in this extended HCI architecture. They were added to and integrated with the Air ASW experimental environment. This environment linked a simulated Air ASW environment with an emulated TACCO workstation with separate display and control software. The added components formed two distinct adaptive interaction subsystems.

The first subsystem monitored the interactions between the user and the workstation and used pieces of the COGNET user model to interpret and simulate the user's problem solving state inside the interface. Within this subsystem, the perceptual demons in the model were programmed to constantly review the display contents for events that would trigger perceptual events. When such events were detected, the perceptual demon would execute itself, resulting in the POSTing of information on programmed representation of the blackboard structure. In addition, the purely cognitive components of individual task models were extracted from the overall set of task models and programmed. These cognitive extracts consisted of attention triggers and the associated POST/UNPOST operations (with their attached execution conditions). Thus, as the perceptual demons began to populate the blackboard data structure with information, the patterns would begin to match triggers for cognitive activities, which would then execute themselves and further transform the blackboard contents. In several cases, it was necessary to link the cognitive task triggers with extended conditions that included both blackboard patterns <u>and</u> observable user actions. For example, the cognitive portions of the Develop Target Fix task would be allowed to execute until the embedded user operation -- the user's placement of a target fix symbol on the display screen -- was also observed by the interface. In this way, the cognitive task components, (POST/UNPOSTs, and perceptual demons) simulated the system user 's cognitive processes. Moreover, this model was linked to the evolving problem context and observable user behavior.

The second subsystem within the adaptive interface used other aspects of the COGNET model to provide the adaptive interaction functions identified above. The attention triggers for each of the tasks in the COGNET model were each programmed separately and placed into a loop in which they monitored the (constantly changing) blackboard contents. When any of them detected a pattern that matched their triggering conditions, they formed a 'hypothesis' that the user's intent would be to begin executing the corresponding task as soon as possible. When multiple triggers detected their triggering patterns on the blackboard simultaneously, they consulted a priority map to establish a (partial order) among these hypotheses. This can be viewed as similar to what Rubin *et al* (1988) and Rouse *et al* (1987) have termed inte inferencing. As new task hypotheses were formed, they were then used to form alerts to the user that the execution of that task might be appropriate (thus providing the task level alerting function). When the partial ordering was able to define a precedence among concurrent task hypotheses, it was use to 'stack' the task-execution alerts in order of decreasing priority.

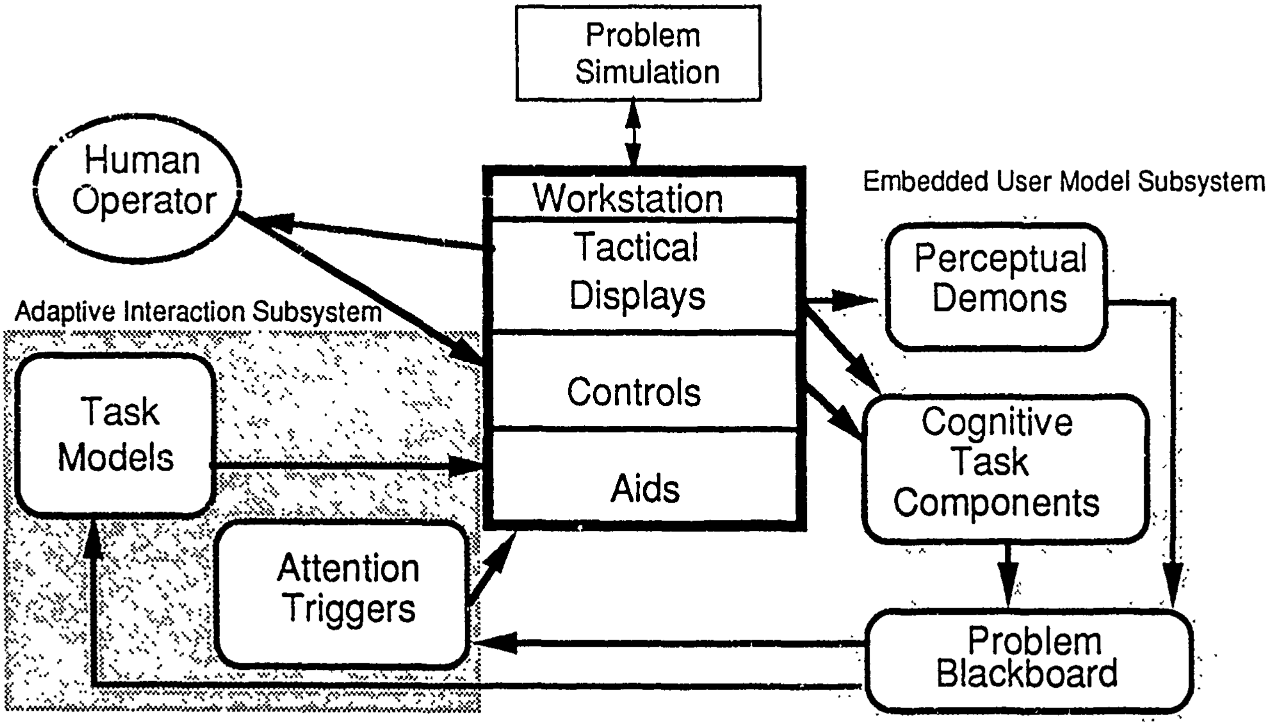


Figure 4-1. Architecture for User-Model Based Adaptive HCI

Any active task hypotheses would be displayed to the TACCO. The TACCO could interact directly with the hypothesis symbol to obtain a display of the subgoal structure of the task, as represented in the COGNET model of that task. These provided a decision structuring function for the user (see Zachary, 1986), by indicating the logical components of the task. The subgoal structures were retrieved from an executable representation of the task-level COGNET model for the task. The model was programmed in executable form, to allow the interface to automatically perform any or all of the subgoals listed for a task. The interface would respond to TACCO requests by retrieving and executing the description of the part of the task model that corresponded to that subgoal. It is important to note that in most cases, this instantiation of a subgoal procedure required use of current problem context, as represented on the blackboard data structure. For example, when the user needed to 'draw environmental features' as part of a task, the interface would use the blackboard representation to infer which features were relevant, and which reference point (usually a specific sensor) to use in drawing them. Thus, the blackboard structure was integral to the context sensitive aiding and task automation provided by the adaptive interface. The implementation of this architecture is discussed in the next Section.

5. IMPLEMENTATION OF THE ADAPTIVE MISSION MANAGEMENT INTERFACE

The proof of the pudding, as the old saying goes, lies in the eating. And so the value of COGNET as a vehicle for embedded user models must lie in the ability to link the model to a human-computer interface in a way that enhances the performance of the human operator as well as the larger system being controlled. This section describes the adaptive extensions to the existing ASW TACCO human-computer interface that were developed to demonstrate the applicability of the ability of the COGNET model. These extensions provide the general intelligent and adaptive interface functions defined in Section 4, as well as implement the architecture presented there.

The interface, as developed, performs four specific adaptive functions:

1 Provides reminders or alerts to the user when the system believes the context is appropriate for initiation of or return to a specific task in the COGNET task network;

2 Indicates the expected priority or order of precedence of the tasks, when multiple tasks are inferred as appropriate;

3 Provides decision structuring assistance, on user request, by identifying the internal organization of goals/subtasks within any given task identified as appropriate for initiation; and

4 Offers automated performance of any or all of the subtasks in a task identified as appropriate for initiation, with the task instance adapted automatically to the interface's understanding of the problem context.

These are not the only ways in which an embedded user model could be applied to enhance computer-human interaction. Rather, they represent an initial demonstration of how a COGNET user model could be applied to extend user interface capability in important ways.

5.1 Screen Design and Layout

In practical terms, it would be an enormous improvement over current interface technology if the system could simply be given access to an approximation of the task-specific, user's mental representation of the problem, and could use this to adapt the interface to the state of the user's problem solving process and current decision making needs. These were the goals of the initial application of the COGNET model of vehicle tracking to enhance the existing HCI used in the initial vehicle tracking simulation.

The actual screen design was guided by a philosophy of minimal intrusion. An attempt was made to minimize the actual visibility of the adaptive interaction functions to the user relative to the pre-existing interface. This was done because the intention of the adaptive interaction functions was to streamline and simplify the interaction with the underlying system. The baseline system was, on the one hand, integral to expert operator's problem-solving procedure. It was difficult for them to separate their knowledge of their current system and its interface from their knowledge of how to solve Air ASW problems. This is quite reasonable since the system and its interface constitutes their view into the ASW world and their only capability to solve problems in that domain. Thus, changing the baseline interface in a fundamental way would require potentially substantial re-learning efforts. On the other hand, the functionality present in the baseline interface represents the capabilities of the underlying hardware/software/electro-mechanical systems that interact with the submarine and oceanographic world. These basic functions cannot be changed without rendering the entire problem moot to both the user and this research. Thus, the only reasonable approach was to build the adaptive interface features 'on top of' the existing interface, but in a highly transparent way, so that:

- the user's pre-existing knowledge about system use was not vitiated,
- the system's underlying functionality was not compromised, and
- minimal additional interaction demands were created by the adaptive interface functions.

The organization of the baseline interface to the TACCO station in the experimental environment is shown in Figure 5-1. The majority of the screen is occupied by a

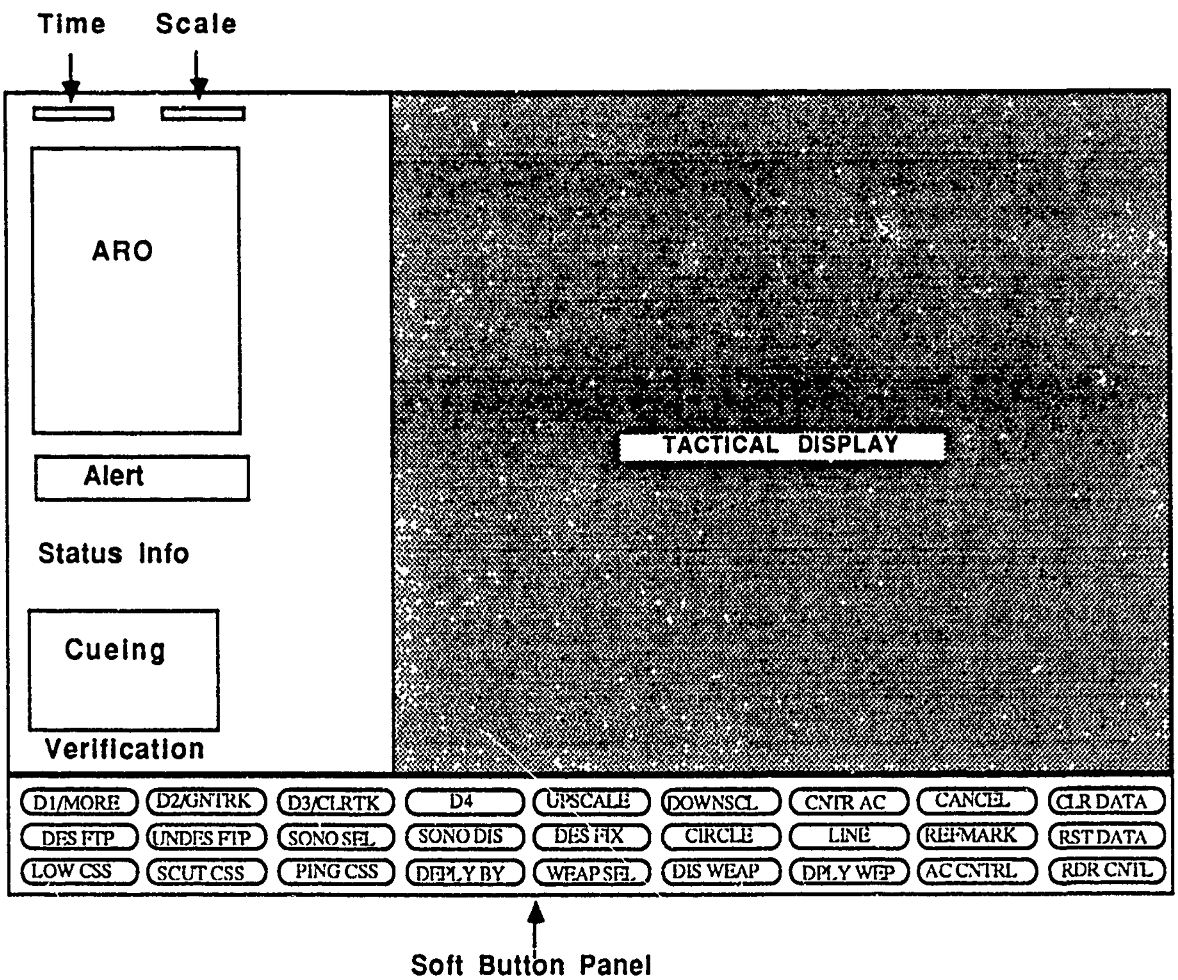


Figure 5-1. Baseline Interface Organization

spatial tactical plot window, with TACCO control functions implemented as 'soft' buttons at the bottom of the screen. The extreme left side of the display was used to display an alphanumeric read out (ARO), plus additional windows for making men selections associated with specific soft button functions. The full interface to the existing TACCO station, and its emulation in the experimental environment used in this study is discussed in Zachary and Zubritzky (1988). The adaptive interface modifications were designed to be as unobtrusive as possible within this basic screen layout.

The basic configuration for the adaptive interface extensions is shown in Figure 5-2. The ARO windows have been moved to an auxiliary alphanumeric display screen (actually making it more like the real workstation). In their place, two aiding windows were added. The top is an alerting window, where task hypotheses are displayed as they are triggered. The hypotheses are labeled to the user as "SUGGESTED STRATEGIES". This is to avoid any implication that the computer is controlling the interaction or is dictating behavior to the user. Any such approach was certain to meet substantial resistance among the user community.

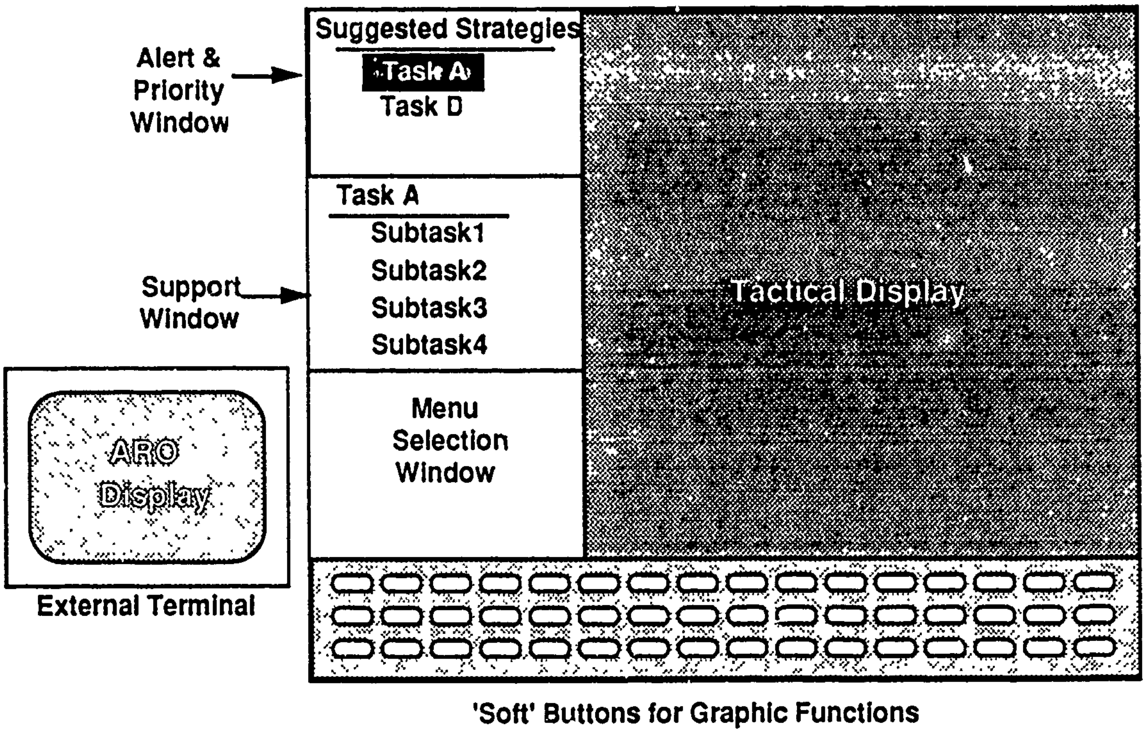


Figure 5-2. Adaptive Interaction Layout

As new hypotheses are triggered, the list may spontaneously reconfigure itself, to move some active hypotheses to higher (or lower priority), or to remove them as their associated blackboard patterns disappear. Within the "SUGGESTED STRATEGIES" window, the user may select any task name by placing the cursor over the name and clicking the mouse button. At that time, a display is created, in the window immediately below it, of the subgoal structure of that task. This window is labeled to the user only with the title of the chosen task from the Suggested Strategies window.

The user may simply review this structure as a sort of procedural checklist, keeping it displayed as the task is performed manually, or the user may close the window directly and focus on a different task. The user can also interact directly with the items listed in this window. Any subgoal in the window can be selected (again, by by placing the cursor over the name and clicking the mouse button). Once selected, the interface will execute that subgoal, following a confirmation by the user.

5.2 Implementation

The initial implementation of the mission management adaptive interface was undertaken as a demonstration of COGNET as a technology for supporting adaptive interaction. Therefore, only a subset of the full COGNET model was used. This subset model consisted of the full blackboard structure (i.e., both panels and all levels within those panels), all perceptual demons, but only five of the 14 cognitive tasks. These five were:

- Identify Area of Interest
- Broaden Initial Contact
- Investigate Convergence Zones,
- Expand Contact for Continuity, and
- Maneuver Aircraft for Magnetic Anomaly Detection (MAD)

These tasks were selected for different reasons.

The first of these tasks, determine area of interest, is a purely cognitive task. It has no behavioral operators within it. It consists of a complex goal/condition structure and POST/UNPOST operators. The mental operations in this task are essential to the proper building and maintenance of the blackboard structure, making this task essential for successful implementation of the "embedded user model" subsystem of the interface (see Figure 4-1 above). This task comprised a major part of the "cognitive task components" module in the adaptive interface architecture. Because it had no behavioral component, it was not included in the adaptive interaction subsystem. That is, the interface made no attempt to alert the user of situations where this task should be performed or to assist the user in performing it.

The other four tasks, on the other hand, were selected only for inclusion in the adaptive interaction subsystem. They were chosen because they:

- represent different phases of the mission, with 'Broaden Initial Contact' and 'Investigate Convergence Zones' (usually) occuring early in a prosecution, and 'Expand Pattern for Continuity' and 'Maneuver Aircraft for MAD' usually occuring later in the prosecution; and
- within these phases, these tasks often compete for attention, thus providing an opportunity to demonstrate both the alerting and the task prioritization functions in the interface.

The adaptive interface architecture discussed in Section 4 was implemented in the context of the ASW mission management experimental environment (Zachary and Zubritzky, 1988). The organization of the experimental environment is shown here in Figure 5-3. The full environment supports scenario/problem development, interactive person-in-the-loop simulation of ASW mission management problems, and recording and analysis of problem data. In this phase of the research only the interactive simulation or testbed portion of the environment was of interest. The ASW mission simulation and emulated TACCO workstation from the experimental environment provided the "problem simulation" and "workstation" components of the architecture in

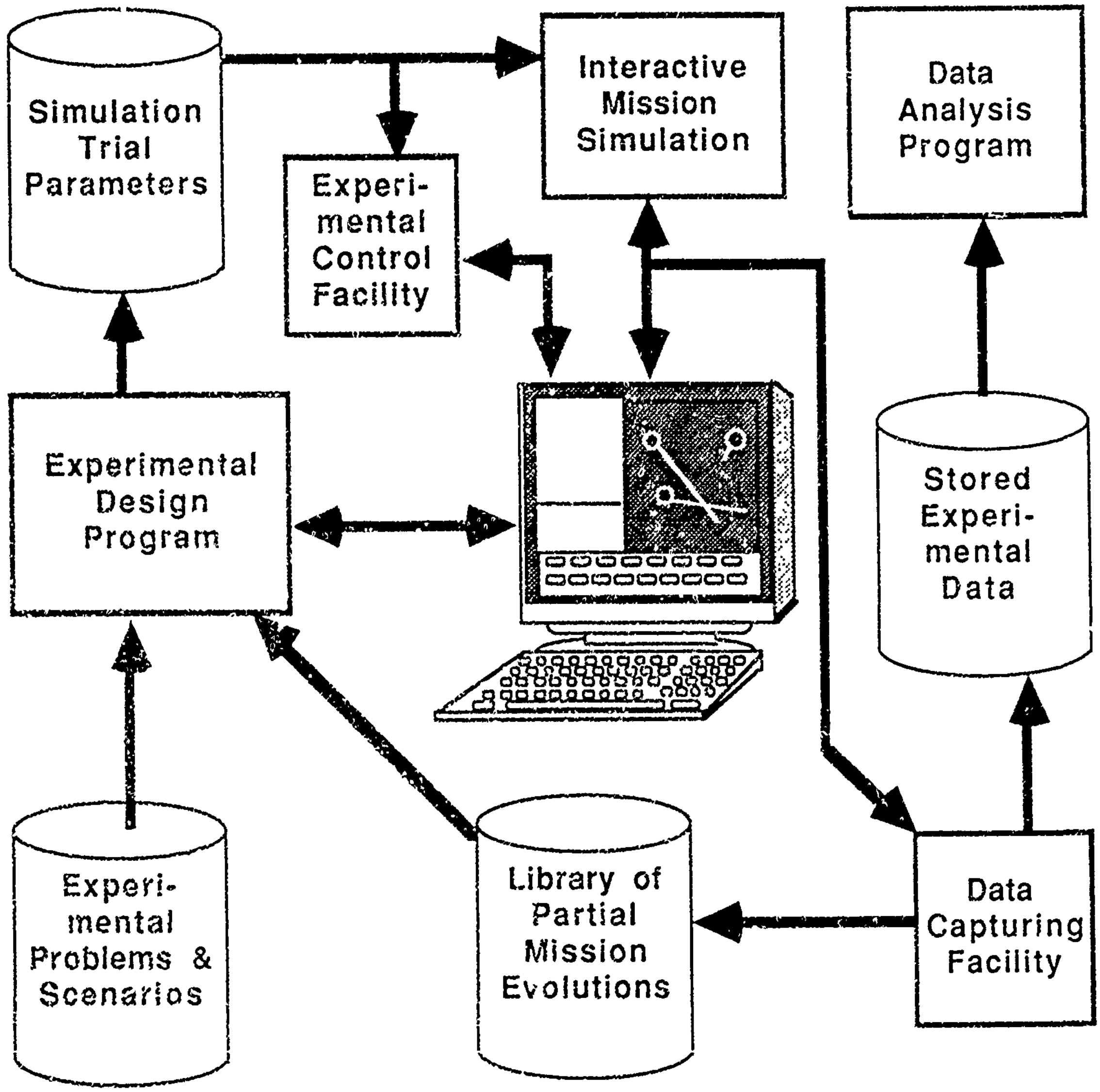


Figure 5-3. Experimental Environment Organization

Figure 4-1. Thus only the components shown as the embedded user model subsystem and the adaptive interaction subsystem had to be added. These pieces were programmed as discussed below, all in the C language as used in the remainder of the experimental environment.

Although the present implementation was intended as a technology demonstration, care was taken to create a set of tools that would support the efficient implementation of the current version of the adaptive interface as well as later extensions to the full system model. Key to this development environment were blackboard-related software tools. The blackboard panels in a COGNET model are extremely complex entities which must be mutable not only during execution in an adaptive interface, but also during the course of software development. This is because efforts at implementation of a model point out hidden ambiguities, conflicts, or omissions, requiring further model revision. Thus, it was necessary to allow for blackboard structure redefinition during the implementation process as well as blackboard contents modification during execution. To do this, we developed a language to define the blackboards which could be used by the model-builder.

The language incorporated the concept of hierarchical layers of items on levels within the blackboard. Each hierarchical layer was addressed by a user-defined name (see the example definition in Table 5-1), in order to make the definition of all the

Table 5-1. Example Blackboard Definition

```
BLACKBOARD Situation
    LEVEL mission_factors
            ITEM mission_factors environment CURRENT ARRAY OF 2 INTEGER;
                    DEFINE NUMBER_CZS BOTTOM_BOUNCE;
            ITEM mission_factors envir_param CURRENT ARRAY OF 5 INTEGER;
                    DEFINE DIRECT_PATH CZ1_INNER CZ1_OUTER CZ2_INNER CZ2_OUTER;
            ITEM mission_factors buoy_resources CURRENT ARRAY OF 2 INTEGER;
                    DEFINE NUMBER_DIFAR NUMBER_ DICASS;
            ITEM mission_factors mission_time CURRENT INTEGER;
            ITEM mission_factors mad_range CURRENT FLOAT;
            ITEM mission_factors mad_ceiling CURRENT INTEGER;
            ITEM mission_factors radar_range CURRENT INTEGER;
            ITEM mission_factors active_mdr CURRENT FLOAT;
    LEVEL expectations
            ITEM expectations contact_expectation HISTORICAL ARRAY OF 3 INTEGER;
                    DEFINE EXP_BUOY_ID EXP_BUOY_TIME EXP_BEARING;
            ITEM expectations captureftp_expectation HISTORICAL ARRAY OF 2 INTEGER;
                    DEFINE EXP_FTP_ID EXP_FTP_TIME;
            ITEM expectations arrive_expectation HISTORICAL ARRAY OF 4 DIFFERENT:
            (FLOAT INTEGER INTEGER INTEGER);
                    DEFINE ARRIVE_RADIUS EXP_ARRIVE_TIME ARRIVEX ARRIVEY;
    LEVEL patterns
            ITEM patterns search_pattern CURRENT ARRAY OF 6 INTEGER;
                    DEFINE S_PATTERN_ID S_PATTERN_TYPE S_PATTERN_CENTERX
                    S_PATTERN_CENTERY S_PATTERN_NUMBUOYS S_PATTERN_STATUS;
            ITEM patterns referent_pattern HISTORICAL ARRAY OF 6 INTEGER;
                    DEFINE R_PATTERN_ID R_PATTERN_TYPE R_PATTERN_BUOY
                    R_PATTERN_REFERENTX R_PATTERN_REFERENTY R_PATTERN_STATUS;
```

cognitive information groupings more readable to the model-builder who was defining the POSTING and TRIGGERING mechanisms. This blackboard definition linked to a parser that translated the high level blackboard definition language into the defined dynamic blackboard data structure. The parser was constructed using the UNIX utilities YACC and LEX. This simplified the modification of blackboards during development. Experimenting with different cognitive structures brought about many enhancements and refinements to the structures and their interactions with the environment. It also supported an iterative process of model development, interface implementation, and model refinement.

The execution-time operation of the adaptive interface revolved around the multi-panel blackboard that represented the TACCO's mental model of the problem being solved. The blackboard is a highly dynamic entity. One of the distinctions in blackboard structure that emerged primarily in attempts to implement the baseline COGNET model was a distinction between blackboard items or messages that are transient and items or messages that are historical. This refereed to the fact that some kinds of reasoning embedded in the various task models refereed to recent rather than current blackboard contents, and in some cases required access to an entire sequence of messages posted in a specific slot on the blackboard. A good example is that of compensating for bearing shift. In several tasks, the user must reason about target location or sonobuoy pattern location. Typically, such reasoning is based on contact messages posted on the blackboard. When there is little movement or random fluctuation in the contact bearing line, the reasoning is straightforward. But in many cases, a locational hypothesis or pattern reference point must be slewed or biased to compensate for perceived systematic movement in the bearing data. Such reasoning requires more information that just the current bearing message posted on the blackboard. It requires the recent history of behavior of that bearing. Thus, it was discovered that some blackboard message types need to carry historical values, while others need only current problem data.

To accommodate this need, two basic types of structures were included in the blackboard: HISTORICAL and STATIC. When a message is POSTed to a STATIC item, the current message in the item is simply replaced with the new message being posted (there is no memory of the previous message). When a POSTing to an HISTORICAL item, the new message group is created and placed at the head of the chronologically linked group of messages which had preceded it. Additional software mechanisms were then developed to support pattern matching with this HISTORIAL type of black board entry. These tools allowed the model-builder to search the blackboard (level and item), from most recent to least recent, looking for the message group which matched the necessary criteria. The model retrieved information which had been stored previously by the user. When an UNPOSTing or a TRANSFORM is performed on an HISTORICAL item, the model would request a search based on one or more desired matching values in the message, or could ask to have the operation performed on the most recently posted message. Data on the blackboard is therefore linked in cognitive groupings (message groups) and in time orientation (memory stack). UNPOSTing a message group makes it unretrievable by the model.

At execution time, all blackboard manipulations are done through a blackboard handler. To TRANSFORM HISTORICAL entries, for example, the blackboard handler searches through the historically maintained list of values to find the one which matches the exact set of requirements, and makes the requested changes. If no matching value exists, it returns a message to the calling routine stating that no match was found.

The blackboard contents are manipulated from two sources -- the perceptual demons and the cognitive task components (see Figure 4-1). In a real-world implementation, the perceptual demons would be true spontaneous computation elements, operating in a closed loop examining the contents of the user's display screen and firing themselves based on new events that matched their 'firing' conditions. In this testbed implementation, however, a shortcut was taken, at no loss of generality to the overall model. Specifically, each perceptual demon was linked to the software event that would generate a display change of the type sought by the demon. Thus, for example, the demon that sought a new DIFAR contact was embedded in the display routine that created new DIFAR contacts, effectively firing the demon every time that display event occured.

The cognitive task components consisted of sequences of POST/UNPOST operations organized into logical groups (corresponding to task-model subgoals). Each group was predicated on a specific blackboard pattern, and its execution was sometimes further conditioned by the specific blackboard contents. The POSTing of various areas of interest from a new contact message is an example. These were groups of operators corresponding to subgoals in the Identify Area of Interest task from the COGNET model. Each subgoal was tied to a blackboard condition, e.g., "any new DIFAR contact". All such groups were evaluated for execution in a closed event loop, by suitable calls to the blackboard handler. When a sought-after pattern was found by the handler, the associated POST/UNPOST operations were allowed to execute, again interacting with the blackboard through the blackboard handler.

The preceding parts of the implementation all concerned the embedded user model subsystem. All remaining portions of the interface implementation concerned the adaptive interaction functions. These include the attention triggers and task models.

<u>Attention triggers</u> are the model component which compared the blackboard contents against the conditions associated with each task in the COGNET model, and alerted the user each time a task's execution conditions were met. In doing this, the trigger uses the blackboard contents as an approximation of the user's mental model of the problem situation. Two sets of conditions were necessary for each trigger -- the condition which would trigger the cognitive task, and the trigger which would cancel the cognitive task. Both were expressed as blackboard patterns, and were evaluated using calls to the blackboard handler. It should be noted that in these cases, the blackboard access was 'read only', in contrast to the embedded user model components described above, which could modify blackboard contents as well. Triggers were checked explicitly by the model when appropriate.

When a given attention trigger was satisfied (i.e., found its desired pattern on the blackboard), two events occurred. First, the associated task name was displayed in the "suggested strategies" window in the interface (see Figure 5-2), and second, the task model component was activated. These task models are the second portion of the adaptive interaction functions. The model had two levels. The first was a listing of the set of subgoals involved in that task, based on the COGNET model of the task. When the the task name was selected from the "suggested strategies" window, the subgoal structure would be displayed to the user in the task support window.

The behavioral activities associated with each task subgoal were also coded as C-language operations and linked with the displayed subgoal in the task support window. By selecting and invoking the displayed subgoal, the user could invoke code that performed the behavioral operations. In most, if not all cases, however, the task could not be carried out without reference to some inferred aspect of the situation, which was represented in the blackboard structure. The task-automation routines would, therefore, adapt their execution to the user's inferred mental model of the situation, as contained in the blackboard structure. For example, the task Investigate Convergence Zones, is meaningful only with regard to a specific buoy contact. The Investigate Convergence Zones subtasks are able to infer the proper reference buoy from the blackboard model, and proceed without further intervention from the user, via calls to the blackboard handler. Thus, these are not simple 'task automation' routines, but complex adaptive automation routines which use the embedded user model to contextualize their execution.

Two types of data recording output were also included in the implementation. All transactions on the blackboard were reported in blackboard.log file. All the activity of the triggers and the user relative to the triggers, tasks, and subgoals was reported in another file called the 'trigger file'. The trigger file was a strict chronological ordering of trigger on/off events and times, and subgoal selections by the user. These were used in the model validation experiment described in Section 3 earlier.

5.3 User Interaction

The actual content and style of interaction afforded by the COGNET-based interface is best understood through an example. This section describes a sequence of interactions between a user and the ASW experimental environment workstation augmented by the adaptive interaction capabilities described above. The example is based on one of the experimental problems created for the experimental data collection phase of this project (see Zachary et al, 1989).

The problem facing the TACCO is to gain contact and prosecute a hostile submarine believed to be transiting in a specific open-ocean area. A summary of the pre-flight information is shown in Table 5-2. It is presumed that some mission planning aid (either on-board the aircraft or at the operations center) has developed a plan for an initial search pattern of sonobuoys, in this case a line or barrier pattern. When the aircraft arrives on-station in this (simulated) mission, the TACCO begins to deploy the recommended line pattern. After deploying the first eight sonobuoys, he gains a reliable contact on the sonobuoy using channel 07. The Sensor Operator has classified this contact as likely hostile.

While these events are occurring, the adaptive interface software, has been POSTING information on the blackboard beginning with the pre-flight information (i.e., Table 5-2) and continuing through the partial deployment of the search pattern and the initial contact. This blackboard information pattern is sufficient to trigger two of the four tasks included in the initial adaptive interaction subsystem. These are the 'Broaden Initial Contact' and 'Investigate Convergence Zone'. These tasks are initially displayed in the "suggested strategies" window of the workstation momentarily after the stable contact is gained. The 'Investigate Convergence Zone' task has been assigned an implicit priority lower then the 'Broaden Initial Contact' so that the latter is displayed first in the suggested strategies window, and the former below it. This situation is shown in the screen snapshot in Figure 5-4.

The TACCO next decides to break off deploying the search pattern, and thus removes the fly-to-points associated with it. (These are part of the 'Deploy Sensor/Pattern' task in the overall model, but this task is not included in the initial

Mission: Detect, Localize, and close Track target submarine. Target is currently transiting southwesterly enroute to onstation. Attempt to gain and maintain attack criteria.

Area: Operating area is 200 NM around datum

Altitude: Unrestricted

Relieve: None

Relief: None

EMCON: Unrestricted

Intel: Latest datum (center of pattern) is a ten hours old at comex. The center of the cold pattern is 120 NM ahead of the last datum at a heading of 225T. The cold pattern covers a target MLA of 200T - 230T and an SOA from 12 - 8.5 KTS.

Tactics: Search pattern is a 16 buoy barrier pattern oriented at 135T with a buoy spacing 4 NM. Buoys are set deep and have a three hour life.

Environmentals: Previous events have been reporting direct path contact out to 3 NM and a CZ at 29 NM with a width of 2 NM.

Surface Temperature	55F
Layer Depth	150 FT
Sea State	3
Ambient Noise	85 HZ
Depth Excess	400 FM

Target Data:

Predominate Frequency	500 HZ
Source Level	160 dB
RD	-5 dB
FOM	80 dB

Detection Ranges:

	MDR	CZ1	CZW	CZ2
Target/Source above layer	2.5	29	2	
Target/Source below layer	3	29	2	--
Target/Source cross layer	2	29	2	---

Buoy Load: 17 DIFAR
7 DICASS

Weapons: 4 MK-46

Table 5-2. Summary of Preflight Information

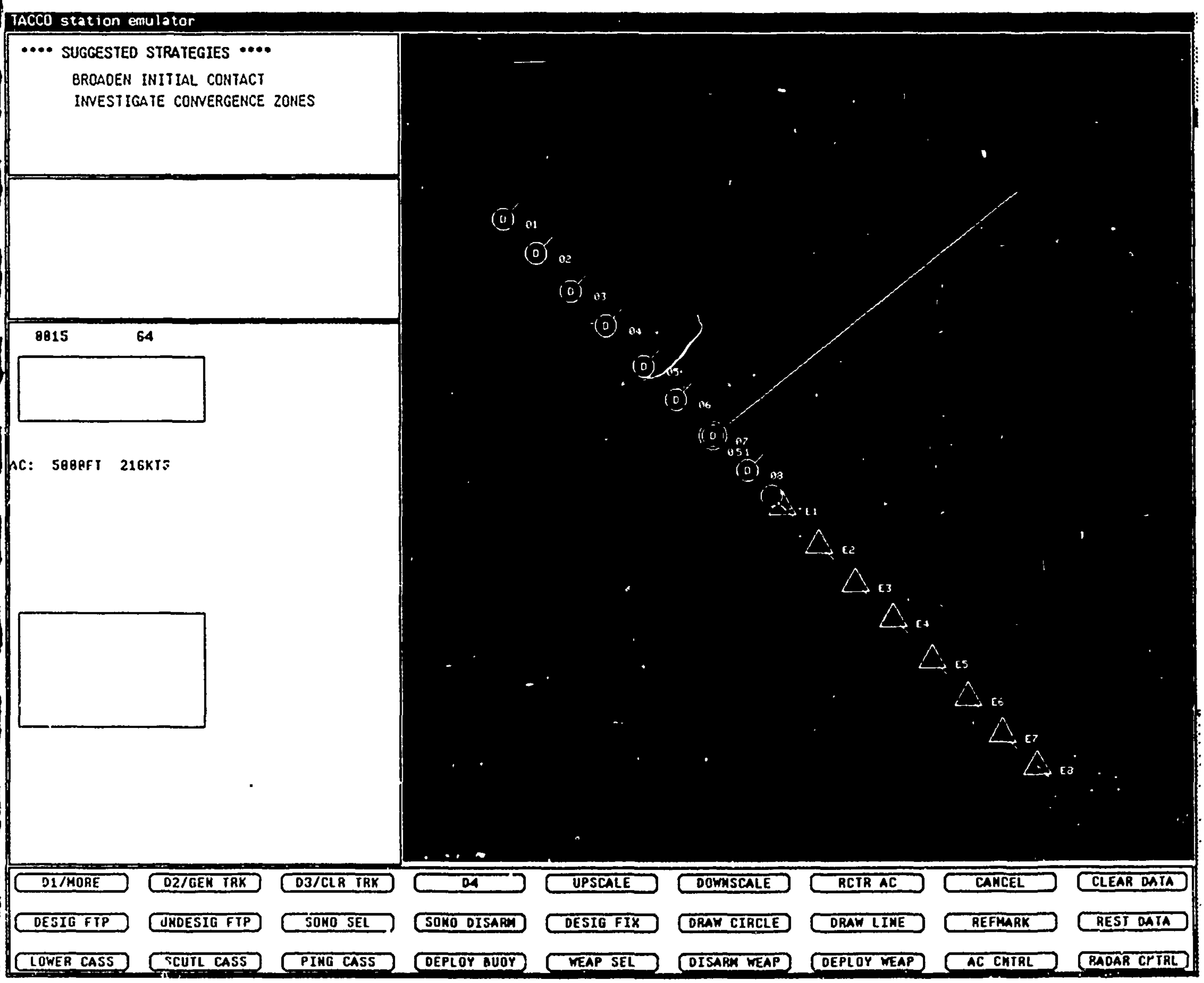


Figure 5-4. Initial Task Triggers Are Fired

implementation; in a complete implementation this task would also have been triggered and would therefore also be recommended in the suggested strategies window at this time). While deleting the pattern of fly-to-points for the remainder of the search pattern, a second stronger contact is obtained on the previously deployed sensor on channel 07, as shown in Figure 5-5.

At this point, the TACCO has still made no overt use of the adaptive interface features, although the display of the suggested strategies may have contributed to the decision to break off the search pattern. Now, however, the use of the adaptive features becomes more direct. In Figure 5-6, the TACCO selects the 'Broaden Initial Contact' task recommendation in the Suggested Strategies window. This results in the display of the subgoals within the task shown in the support window below the suggested strategies. In other words, the TACCO has asked the aid, in Figure 5-6, what is involved in the 'Broaden Initial Contact' task, and it has displayed three subgoals: focus the display on the contact, build a sonobuoy pattern known as a prudential pattern, and arm sonobuoys for deployment when the fly-to-points in the pattern are captured by the aircraft pilot. The final 'exit' item is merely an implementation convenience.

The TACCO then decides to have the interface make an initial attempt at carrying out some of these subgoals. In Figure 5-7, the 'Focus on Contact' subgoal has been selected. The interface has knowledge about the cognitive and behavioral operations involved in building each subgoal in any task because they are programmed into it using the COGNET models of the task. With regard to the 'Focus on Contact' subgoal, the interface uses the blackboard information to determine which buoys are in contact and are relevant to the 'Broaden Initial Contact' task (in this case, these are the sensors on channels 07 and 08). With this knowledge, the interface centers the display on the area of these sensors, and reduces the scale until the sensors and their full bearing line fill the screen. This automated action by the interface saves the TACCO approximately six interactions with the workstation.

The TACCO, satisfied with this choice selects the next subgoal -- 'Build Prudential Pattern'. The result is shown in Figure 5-8. Here, the interface uses even more procedural knowledge from the COGNET model of the task and contextual knowledge from the blackboard. A prudential pattern is defined[1] as a pattern of fly-to-points containing a specific geometry but conditioned to a reference point and the local oceanographic conditions. The reference point is selected, based on the COGNET model, as the sensor which has the best contact. Using the blackboard contents and conditions extracted from the COGNET model, the interface selects sensor on channel 08, although a human TACCO might select either that or the sensor on channel 07. This intention, is displayed immediately in the task detail window via the recommendation

BROADEN CONTACT ON 8.

Knowledge about the prudential pattern geometry is built into the interface, again from the COGNET model of the task, as is the knowledge of how to adapt the geometry to the local oceanographic conditions. These conditions are retrieved from the situation blackboard. Thus, the interface is able to define the reference point and spacing for the elements of this pattern. It also knows the sequence of functions

[1] or 'instantiated' in artificial intelligence parlance.

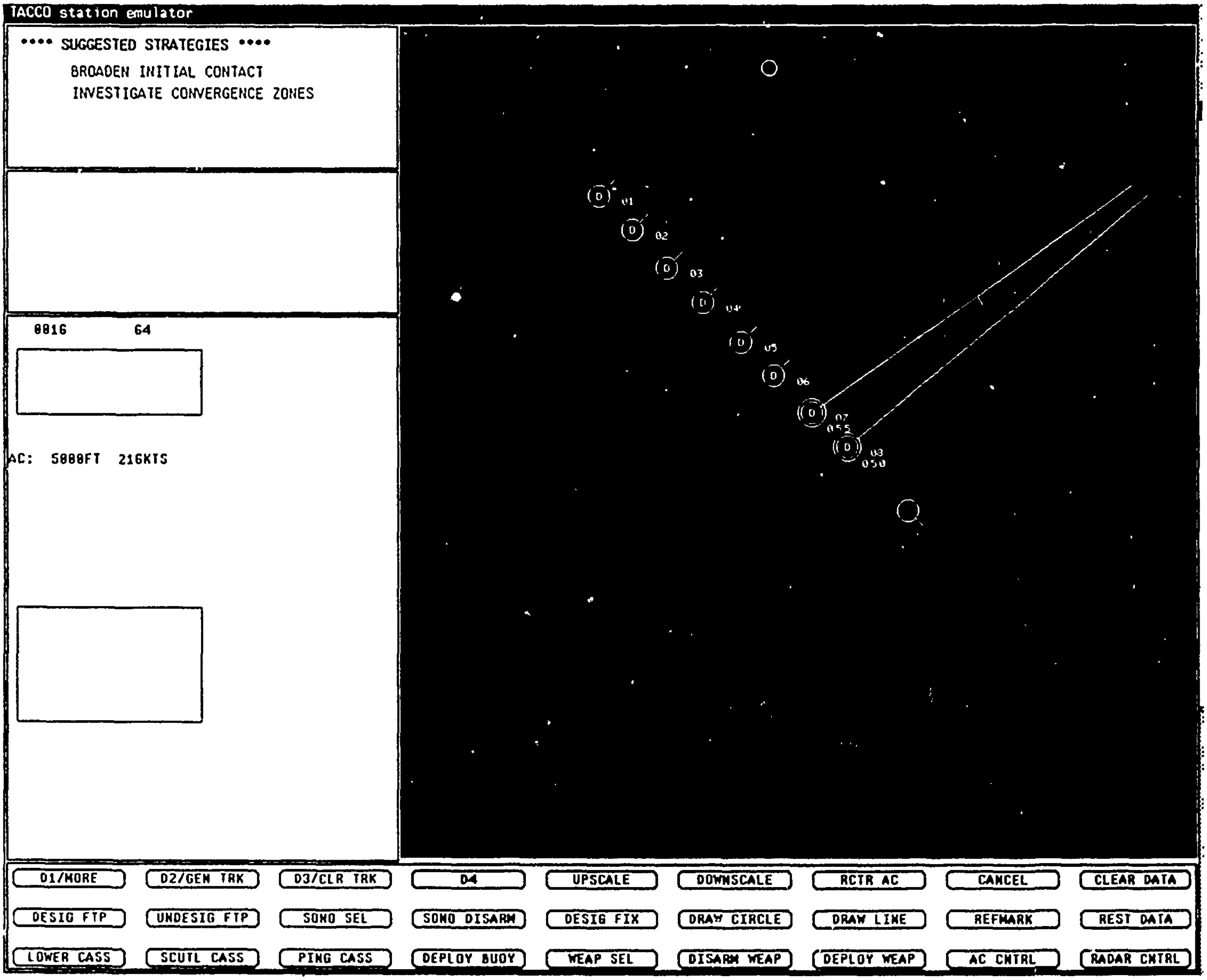


Figure 5-5. Search Pattern Broken Off and New Contact Gained.

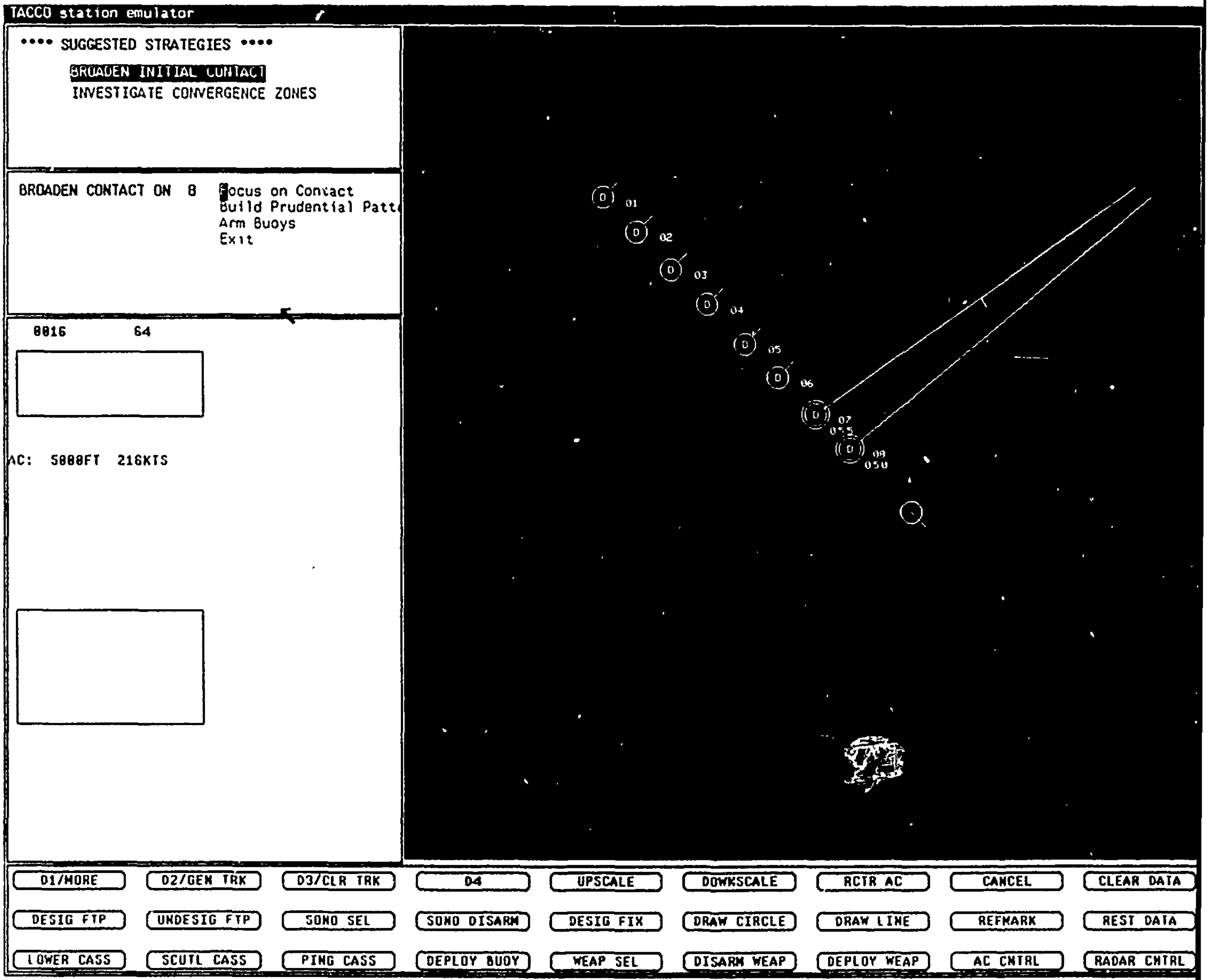


Figure 5-6. Subgoal Structure of "Broaden Initial Contact" Displayed.

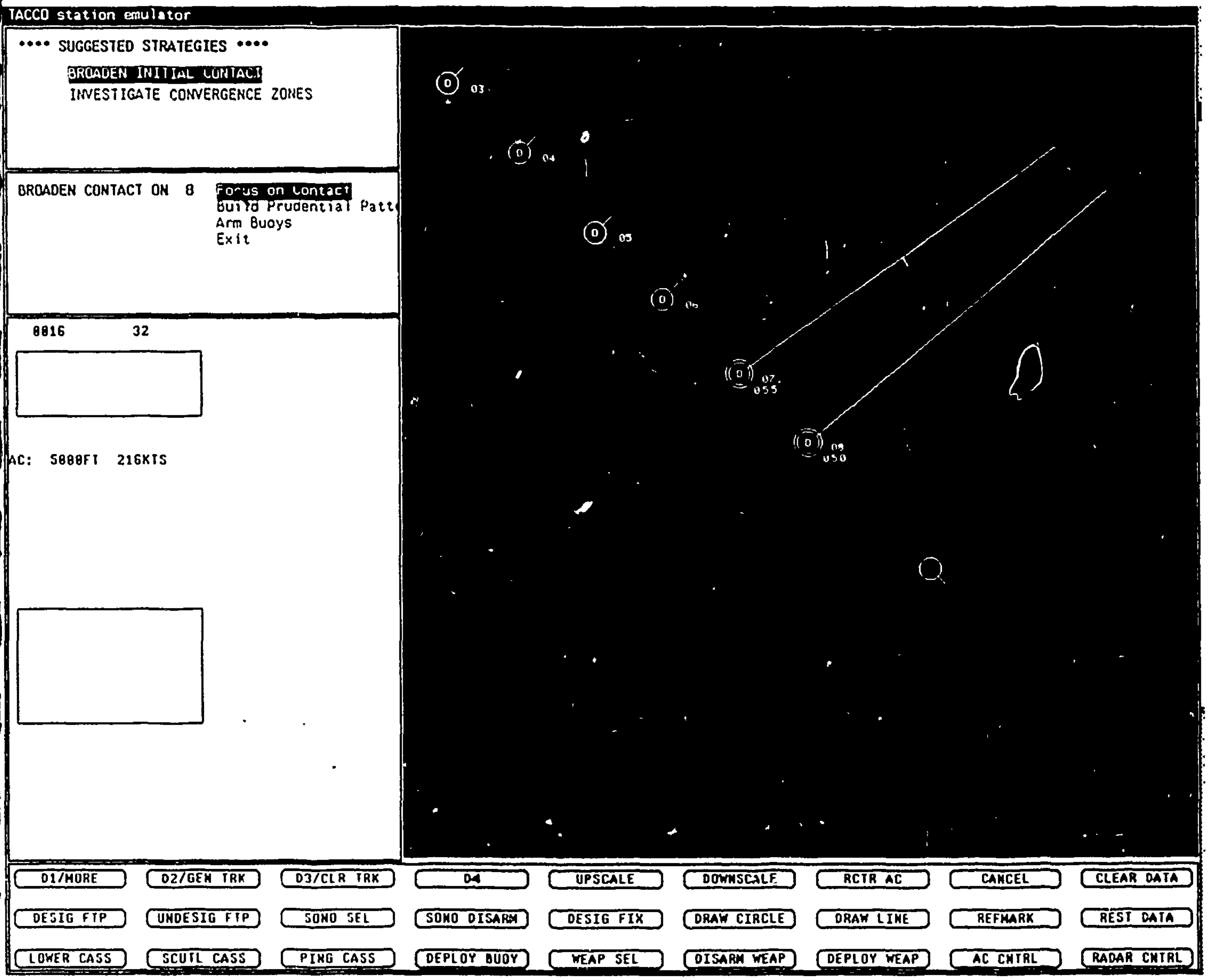


Figure 5-7. Focus on Contact Subgoal Accomplished by Interface

needed to implement this pattern instance as fly-to-points on the screen, yielding the display shown in Figure 5-8.

The TACCO in our example chooses to accept the solution adapted by the interface. The solution has saved him several minutes of detailed interaction and eliminated several opportunities for slips that could have had serious tactical consequences later.[2] However, the TACCO chooses not to have the interface arm the sonobuoys for deployment, but reserves that task for himself and selects 'exit' from the support window.

With the task of broadening the initial contact completed, the user follows the advice in the suggested strategies window and seeks to investigate the convergence zone. This entry in the suggested strategies is selected, resulting in the display of the subgoals in the support window as pictured in Figure 5-9. As in the case of broadening an initial contact, this task has an implicit reference point which is based on the current mission context. The interface has inferred that this should also be the sensor using channel 08. It bases this decision on both the context of the current and recently-past sensor contacts, and the fact that the broadening of initial contact is also using the sensor on channel 08 as a reference.

The first subgoal of this task is the plotting of CZ circles. The TACCO decides to allow the interface to attempt this, and selects that subgoal, creating the display shown in Figure 5-10. In this case, the interface has not only used the information on the situation blackboard panel about the environment (i.e., where the CZs, if any, are located), it has also used knowledge about the display. Specifically, it has determined that if the CZ circles are to be displayed, then the screen must be upscaled to do so. This gives the TACCO a 'big picture' of the convergence zone location with regard to the remainder of the pattern already deployed. The next subgoal, though, is to focus the display on the contact-sensor and convergence zone area where the investigative pattern is to be displayed. This is accomplished by letting the interface perform the second subgoal in the task detail window. This is pictured in Figure 5-11. In this case, the interface aqain uses display-manipulation knowledge from the COGNET model to determine that the primary contact sonobuoys <u>and</u> the relevant convergence zone area should both be visible during this operation. The interface has, therefore, not only changed scale but also recentered the display area to support these visual needs.

The TACCO accepts this action also, and then selects the Build CZ Pattern subgoal to allow the interface to make the initial layout of the fly-to-points that comprise the convergence zone investigative pattern. The resulting display is shown in Figure 5-12. As with the prudential pattern, the convergence zone investigative pattern consists of a fixed geometry and reference point or line, adapted to the specific context of its application. The interface has already selected the sonobuoy on channel 08 as the reference point for the entire task, but this must be more refined in the interaction associated with laying out the convergence zone investigative pattern itself to include an orientation. The orientation must be a specific reference line in relationship to the reference point. The logic for determining it can be complex; in this case the interface has attempted to average the bearing of the two sensors in contact to produce the orientation line. (This interface chose this averaging because, in the context shown here, the two 'good' contact lines do not intersect in the convergence zone, and thus

[2] In following Norman (1983), we define 'slip' as an error of performance in contrast to a 'mistake', which represents an error of knowledge or competence.

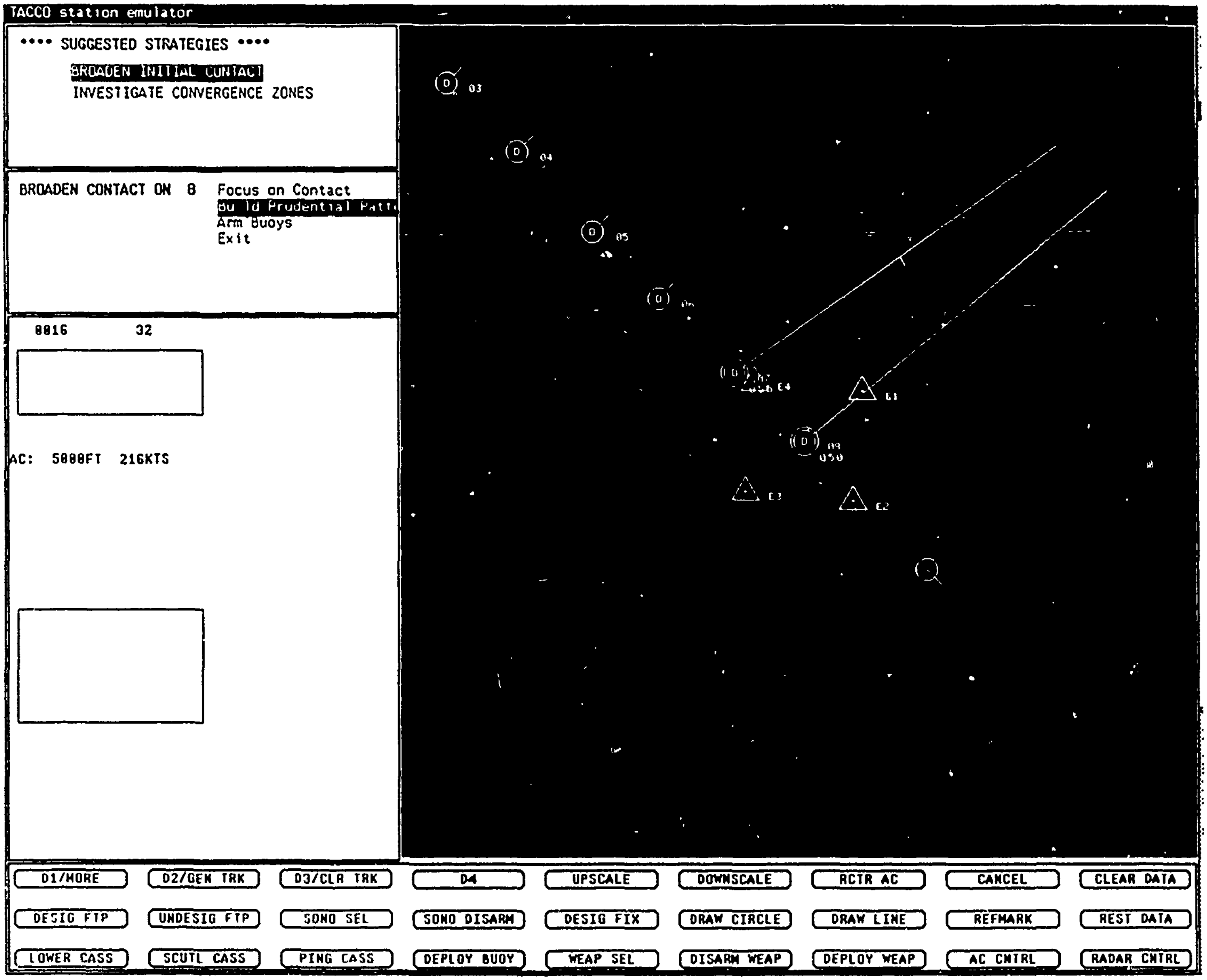


Figure 5-8. Build Prudential Pattern Subgoal Accomplished by Interface

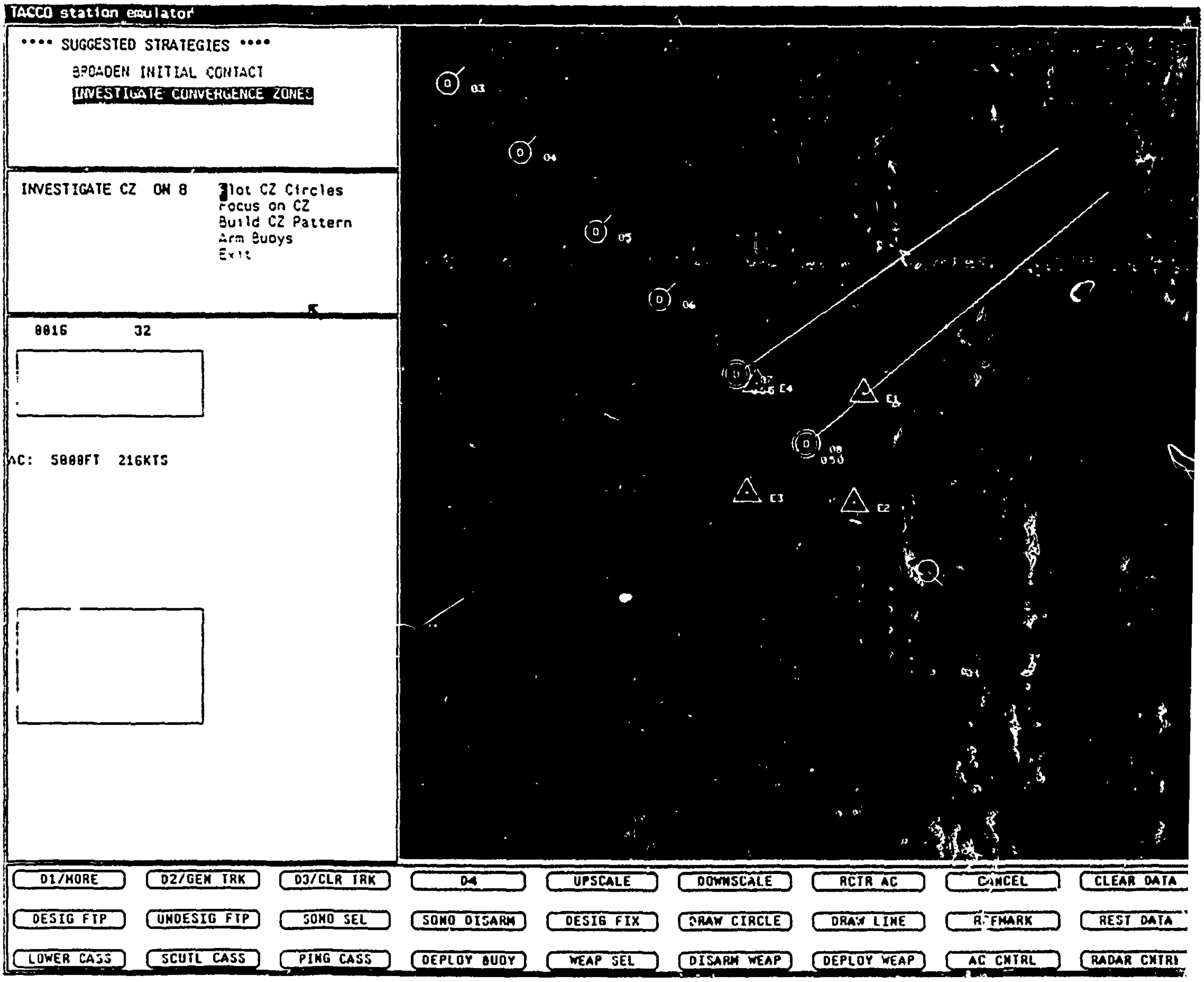


Figure 5-9. Subgoal Structure of "Investigate Convergence Zone" Displayed

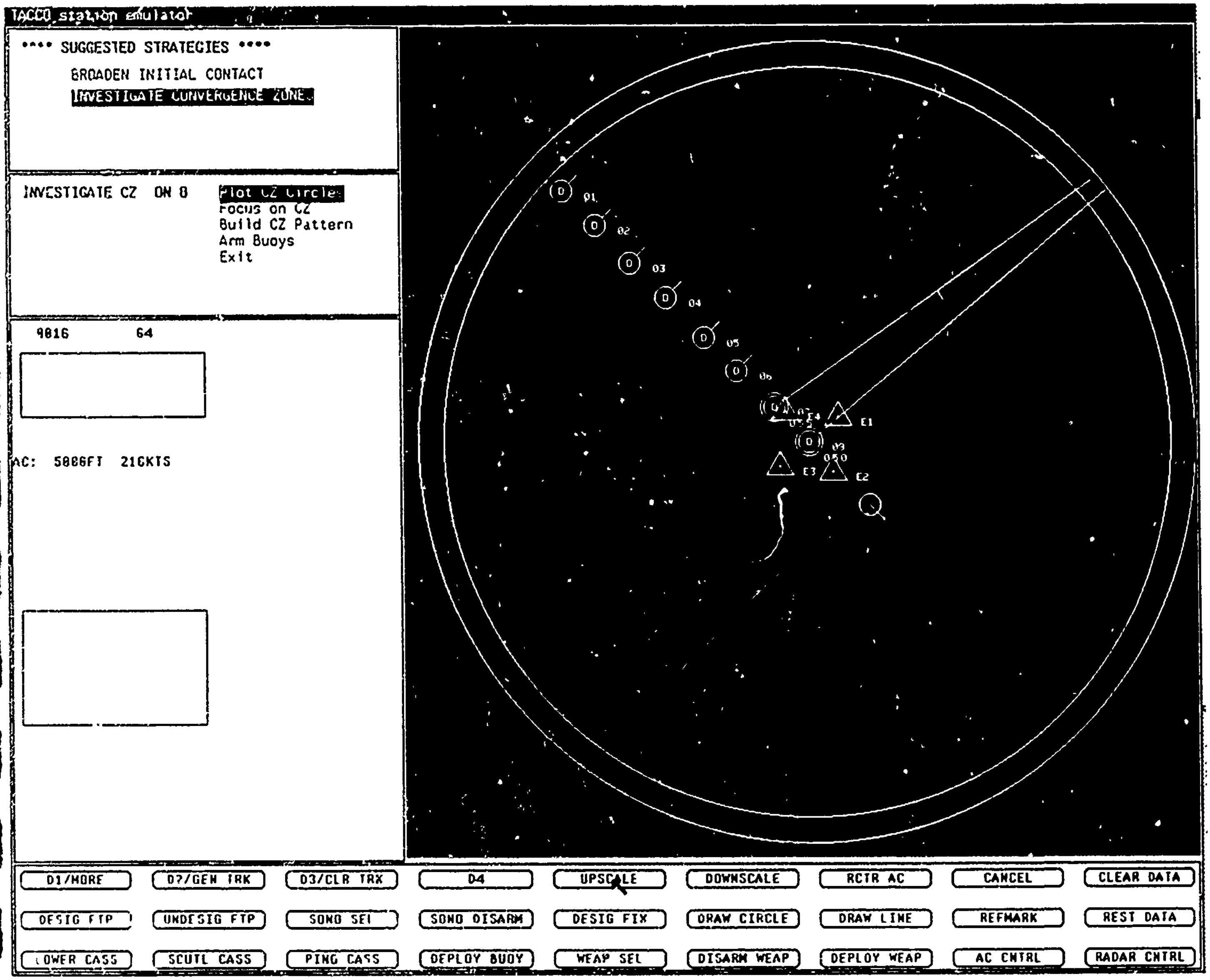


Figure 5-10. Display CZ Circles Subgoal Accomplished by Interface

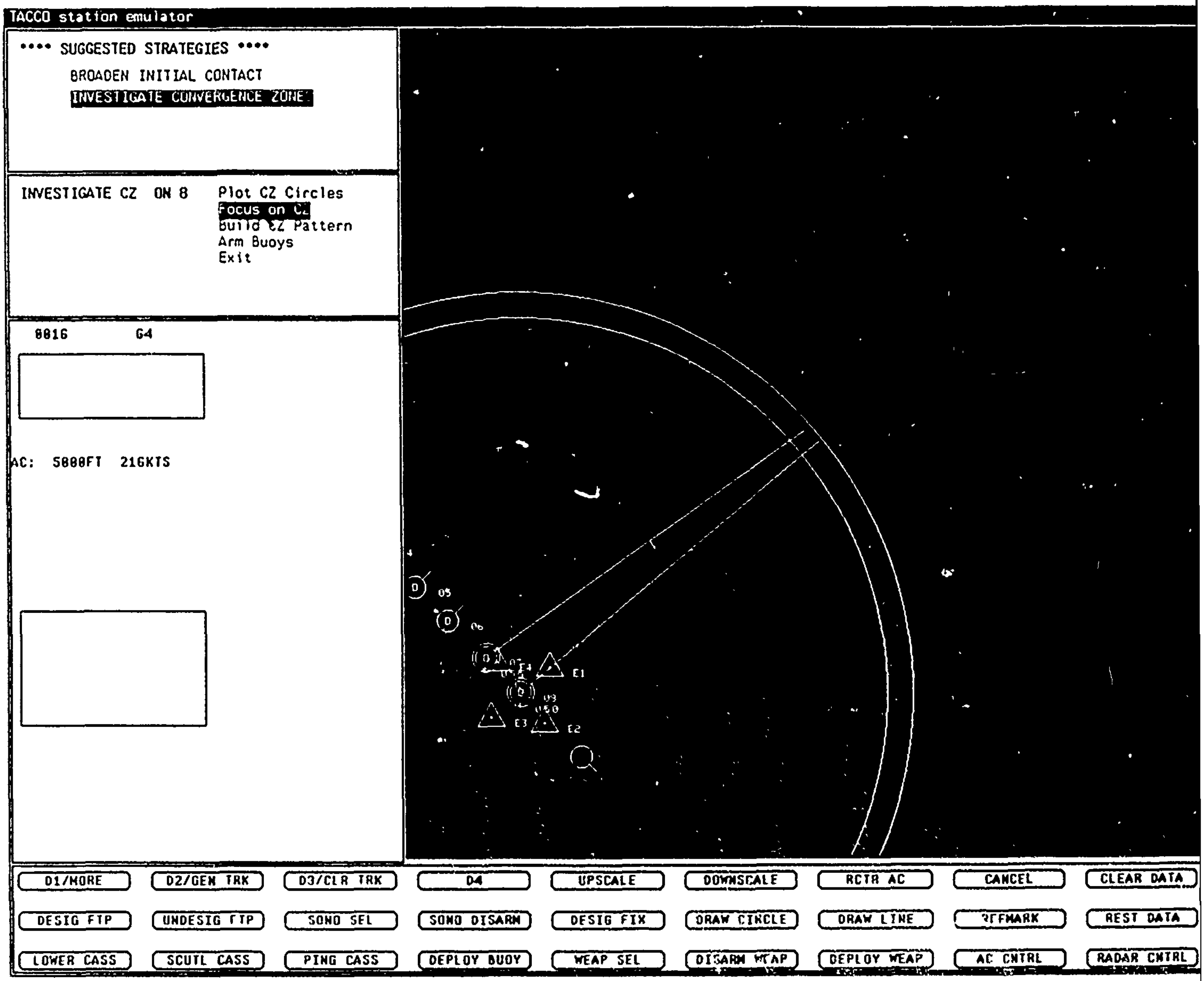


Figure 5-11. Focus on Contact Subgoal Accomplished by Interface

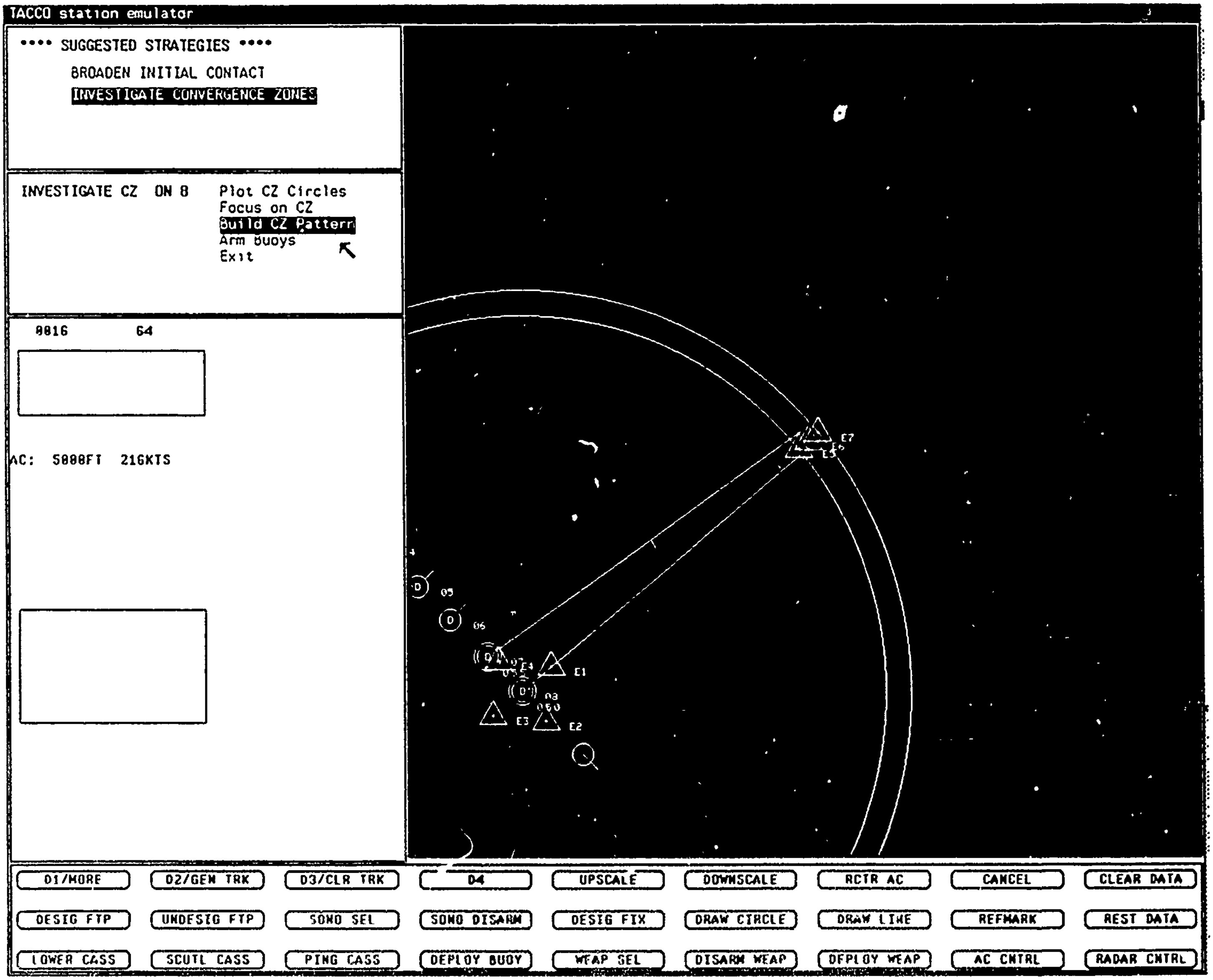


Figure 5-12. Build CZ Pattern Subgoal Accomplished by Interface

suggest that both contain some bearing error.) The orientation line for the pattern is thus derived from the blackboard containing the current context. The convergence zone width is similarly retrieved from the blackboard to define the sonobuoy spacing and pattern center, resulting in the pattern layout shown in Figure 5-12.

As in the broaden initial contact case, the TACCO decides to retain control over the arming of buoys for himself, and thus 'exits' from the investigate convergence zone task after building the CZ pattern. Here again, the interface is unable to determine that this task has been completed, and keeps it on the suggested strategies list for some time.

The TACCO continues with prosecution of the mission for some time forward from this point. The prudential and convergence zone investigative patterns are deployed, and eventually result in additional contacts in the convergence zone, as shown in Figure 5-13. At this point in time, the TACCO has several sensors in contact, including the two shown in the tactical plot in Figure 5-13. Others are in the initial search pattern or the prudential pattern. The TACCO now feels that he has a reliable intersection of the bearings on sensors assigned to channels 11 and 12, and so places a reference mark to indicate a target fix on the screen (represented by the 'X' on the screen). This patten is still consistent with the triggers for both the 'Broaden Initial Contact' and the 'Investigate Convergence Zone' tasks, and so both are still displayed in the suggested strategies window.

After the target fix is placed, however, several things change. The first change is that the pattern associated with the 'Broaden Initial Contact' and the 'Investigate Convergence Zone' tasks is no longer satisfied. The 'untriggers' associated with these tasks are then invoked, resulting in these tasks being removed from the suggested strategies window. At the same time, the pattern for another task -- 'Expand Pattern for Contact Continuity'-- is satisfied by the new blackboard contents (for the first time). The display from Figure 5-12 therefore transforms itself into that shown in Figure 5-14. Almost immediately the sensor on channel 13 gains a strong contact, leading the TACCO to follow the suggestion of expanding the current pattern to maintain continuity of this contact.

The TACCO selects the 'Expand Pattern for Contact' task from the suggested strategies window, creating the subgoal structure shown in the task detail window in Figure 5-15. Many of these subgoals are mutually exclusive in the full COGNET model of this task, and the interface, in fact, will not perform any that are not appropriate for the current context.

In Figure 5-16, the TACCO selects the first subgoal in the support window -- 'Focus on Contact' -- allowing the interface to attempt to accomplish this subgoal. The interface has determined from its COGNET model of the task and the current blackboard contents that the pattern should be expanded using the sensor on channel 13 as a reference point. Note at this point that the interface must upscale in order to accomplish its strategy for the 'Focus on Contact' goal. Its strategy seeks simultaneous display of the primary sensors in contact, the full bearing lines, and the aircraft symbol. Accepting the results, the TACCO selects a tritac tactic to maintain contact continuity. In Figure 5-17, he invokes the interface to define and build a tritac pattern. As in previous pattern related cases, the interface has used its blackboard representation to adapt the canonical pattern geometry to the specific tactical and environmental context of this situation.

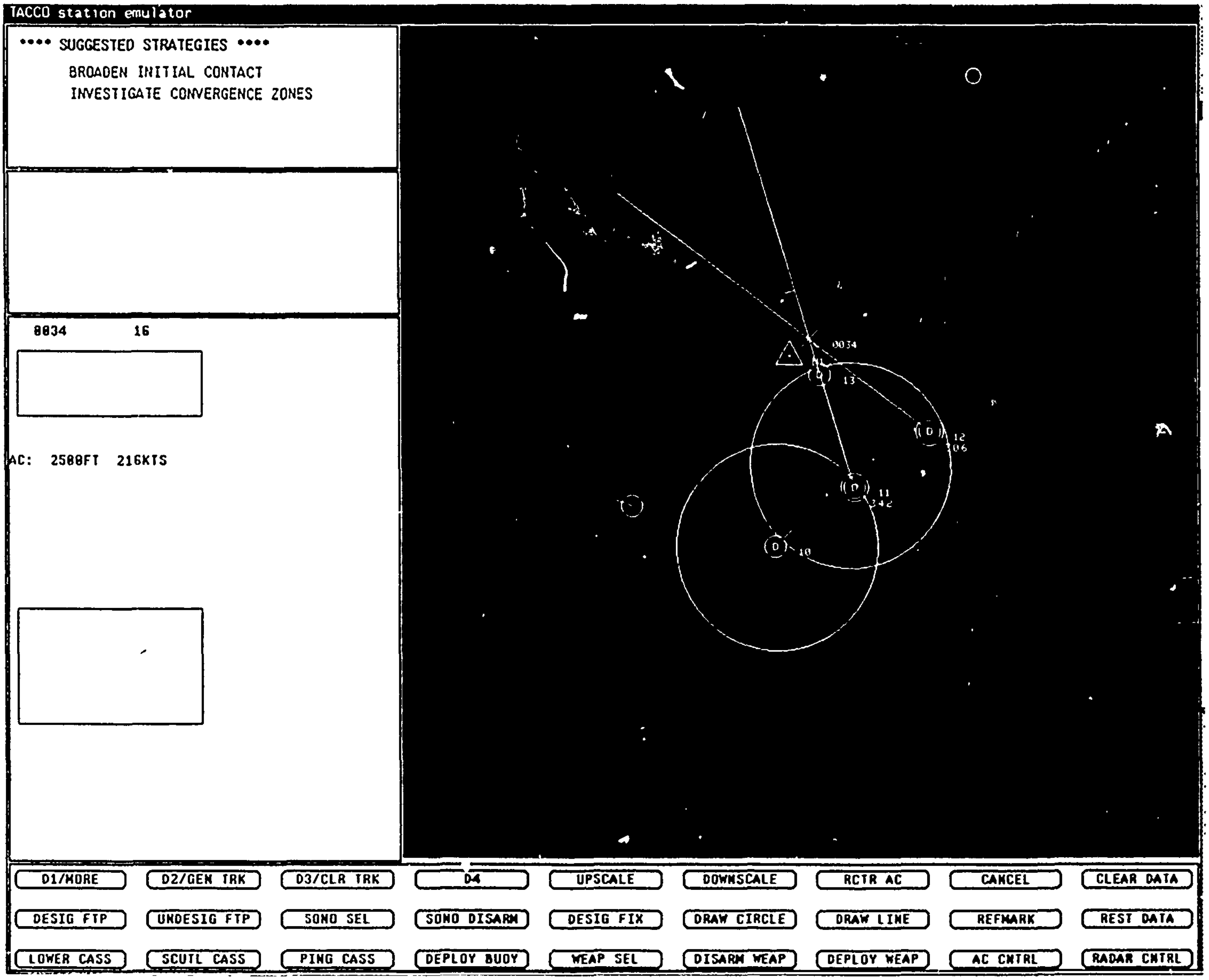


Figure 5-13. Later Contacts in Convergence Zone

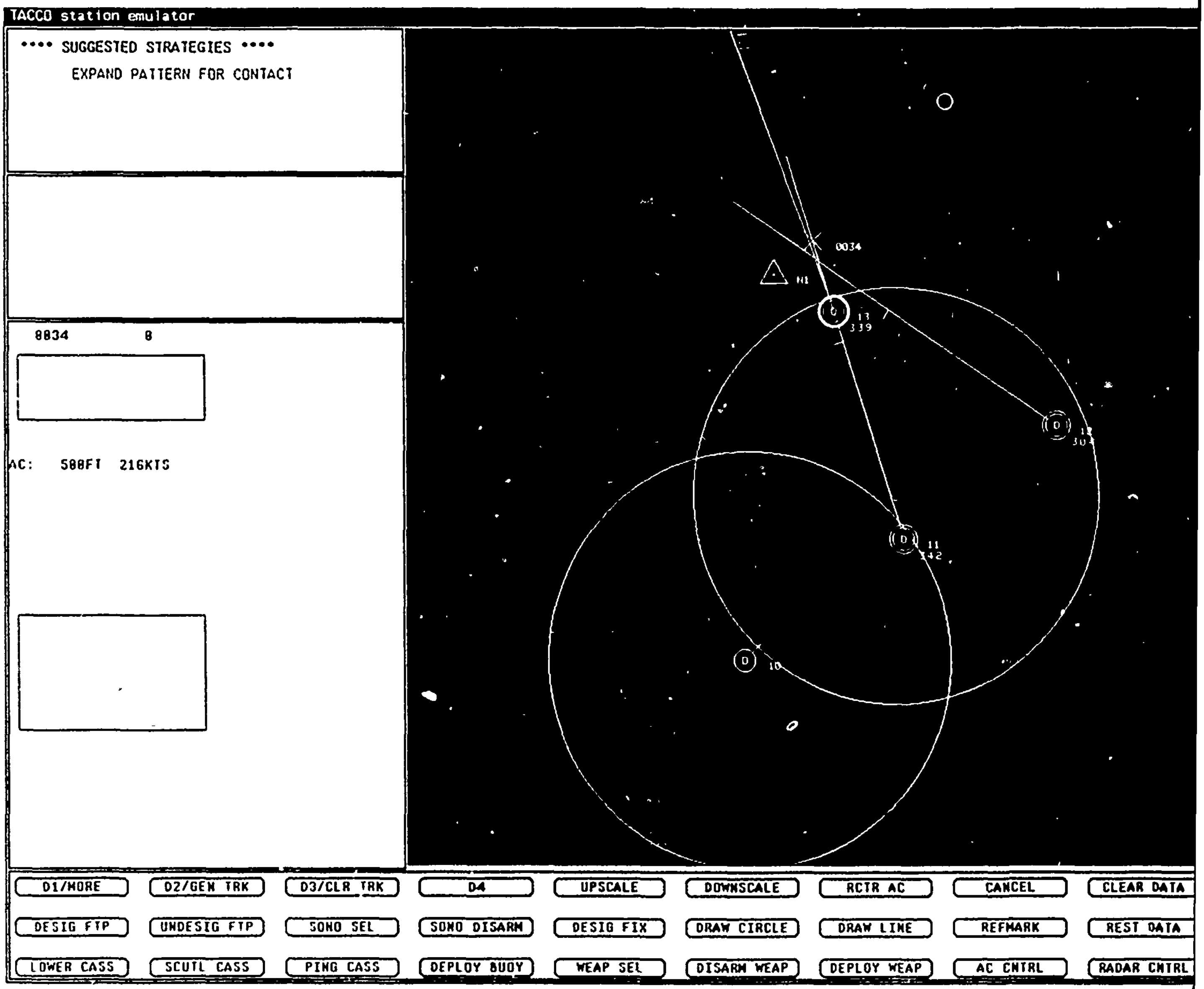


Figure 5-14. Expand Pattern For Contact Triggered

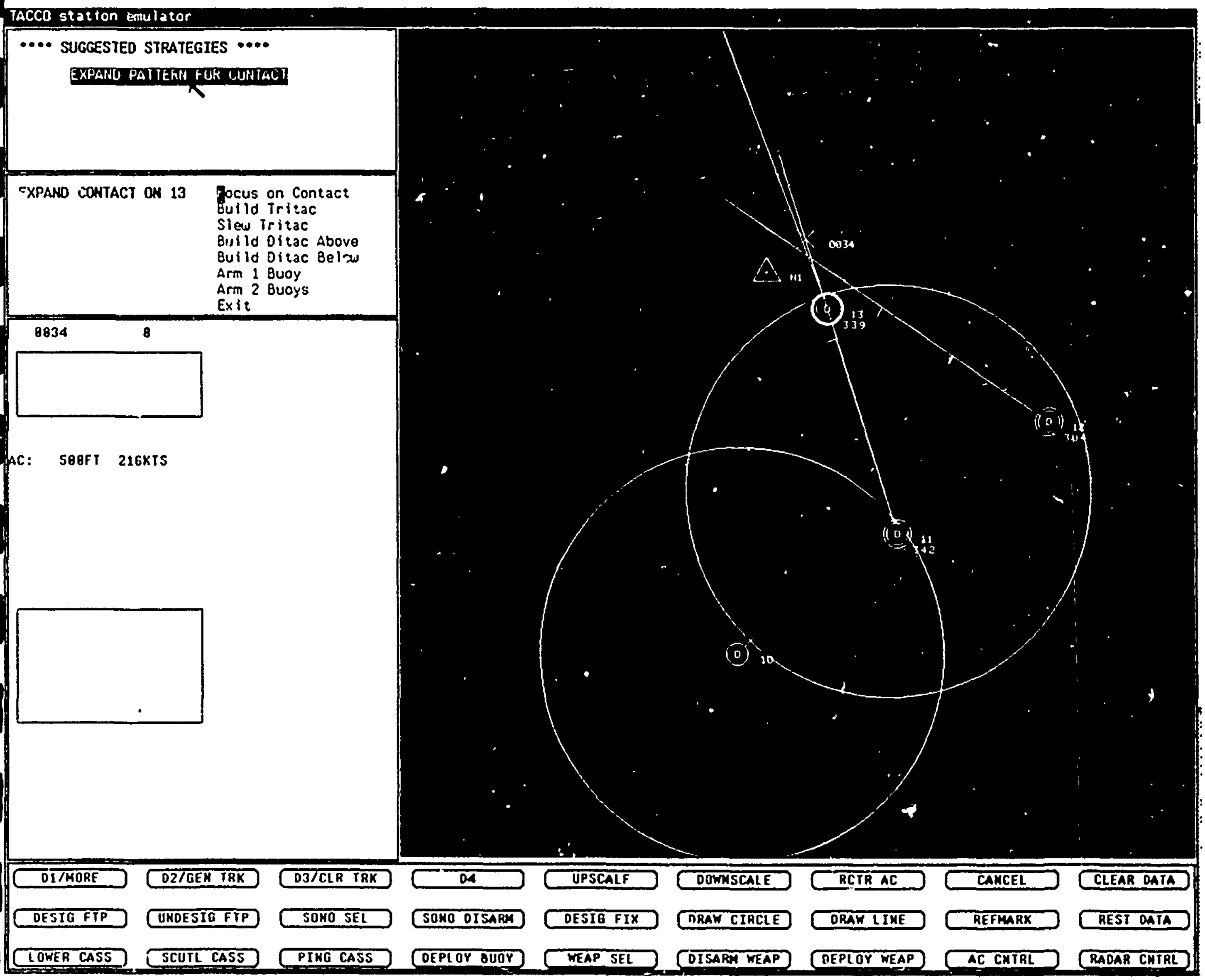


Figure 5-15. Subgoal Structure of 'Expand Pattern For Contact' Displayed.

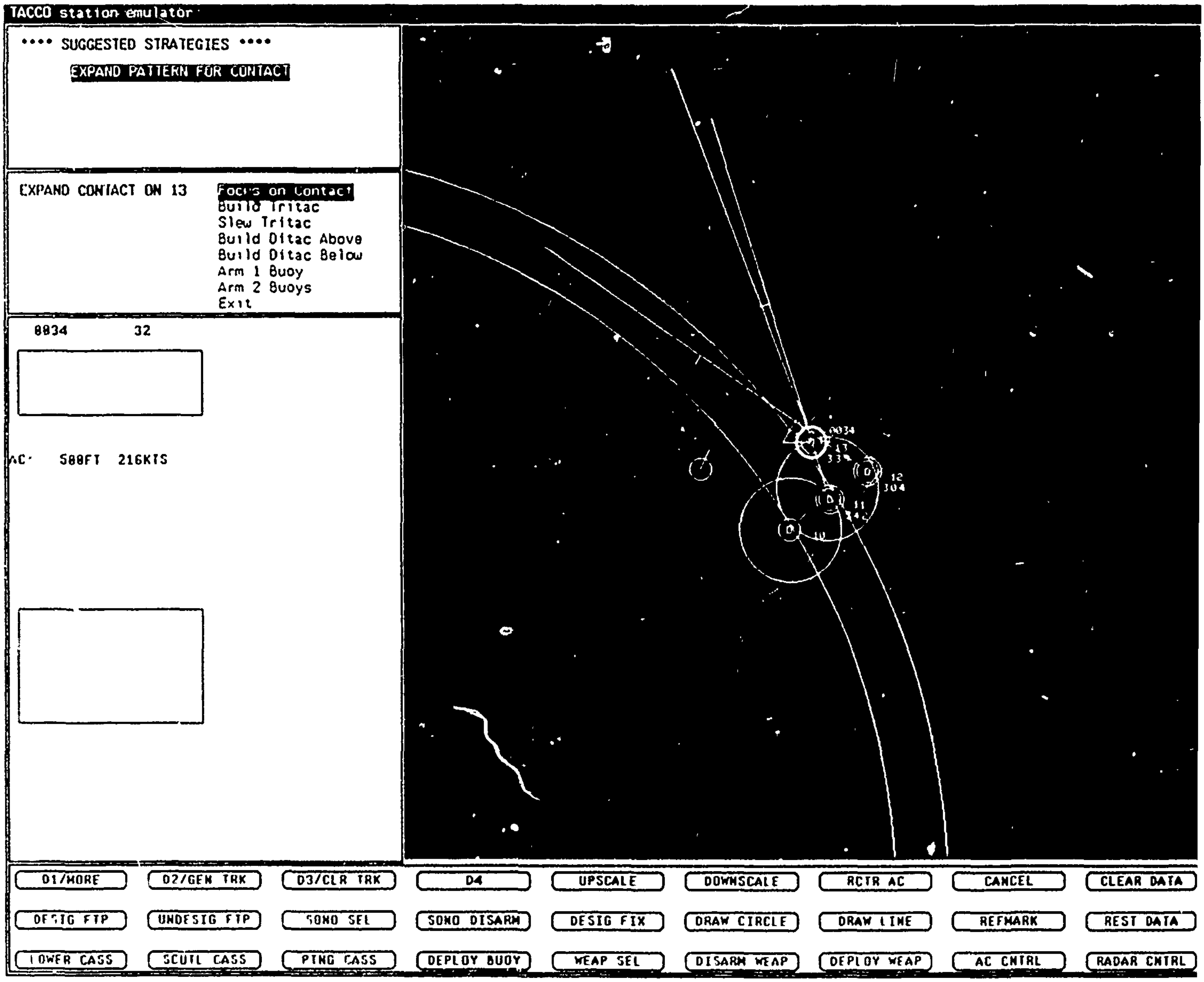


Figure 5-16. Focus on Contact Subgoal Accomplished by Interface

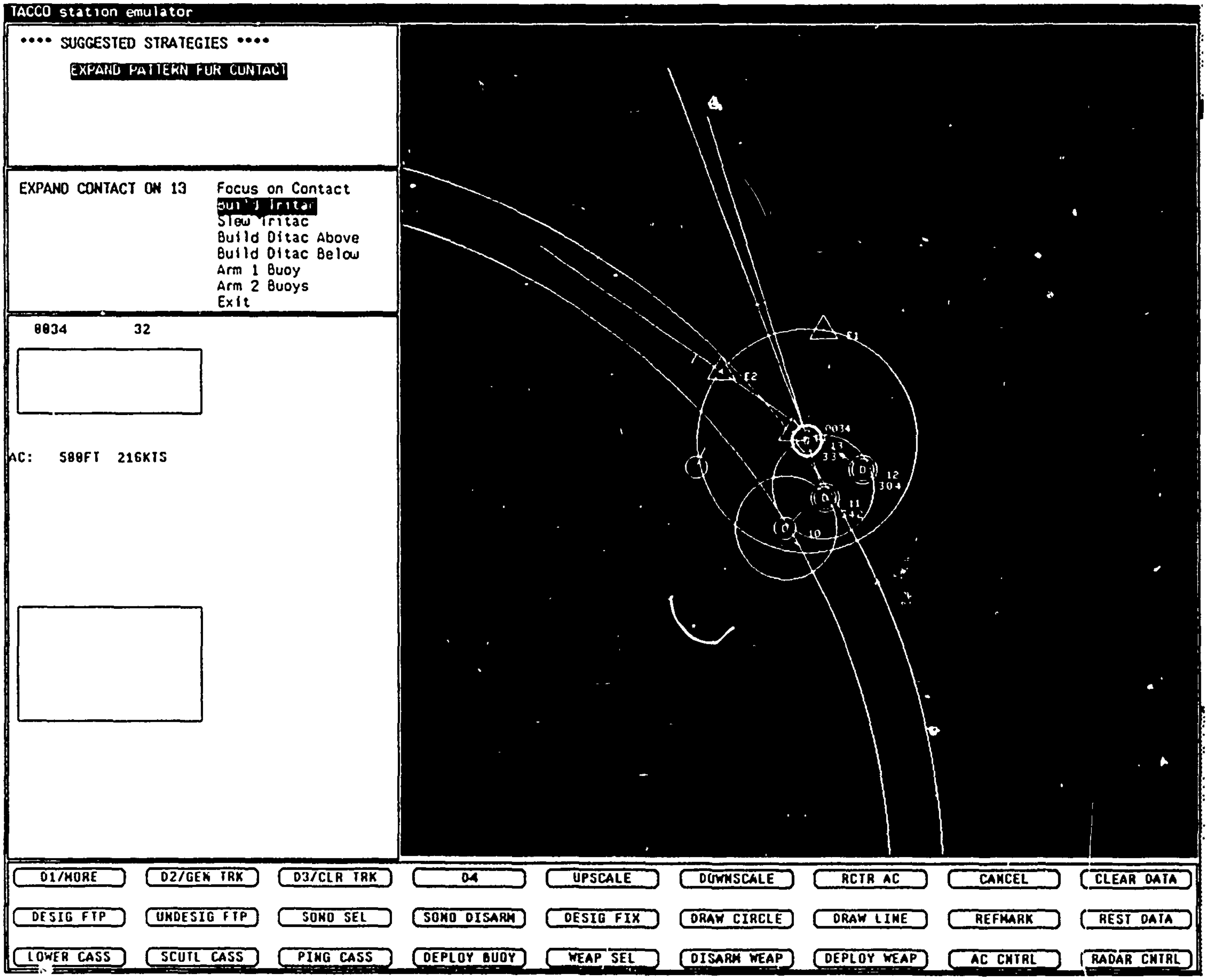


Figure 5-17. Build Tritac Subgoal Accomplished by Interface

The interaction continues for some time. Opportunities to apply Magnetic Anomaly Detection (MAD) sensors arise later in the interaction, but are not shown here. In general, the initial adaptive interface is not hampered by having only four of the possible 14 tasks implemented. It is able to make recommendations and provide interaction support based on the evolving mission context and estimations of the TACCOs relationship to that situation. Such a feature is positive, in that is suggests that the interface will not be easily 'side-tracked' in cases or situations where it can not maintain a constant model of precisely what the TACCO is doing. This is highly likely in real-world applications of this interface concept.

6. CONCLUSIONS

This research effort to investigate the cognitive basis for human-computer interaction and decision making in RTMT environments has resulted in the development of COGNET, a new approach for modeling RTMT information processing. COGNET provides both theoretical and methodological advances for developing human-computer interfaces in RTMT environments. The three major advances are:

- development of the COGNET framework for modeling human RTMT information processing;
- development of a COGNET cognitive task analysis toolkit and methodology; and
- demonstration of a novel adaptive human-computer interaction concept and architecture.

The COGNET framework has been developed and applied to a vehicle tracking domain based on Naval Air ASW. Within this context, a COGNET model of mission management was developed. Initial validation of the model as a behavioral predictor has been accomplished, and an adaptive interface developed. While additional research is suggested by the results and technologies discussed in this report, the COGNET toolkit is sufficiently well developed and validated to support real-world system development efforts in Naval ASW or other RTMT domains.

The main theoretical and methodological accomplishments are summarized below in Subsection 6.1. Additional operational benefits are summarized in Subsection 6.2. Finally, a set of future issues raised by this research is introduced in Subsection 6.3.

6.1. Main Theoretical and Methodological Accomplishments

The first and foremost product of this research is COGNET, a theoretical model of human problem solving and human-computer interaction in real-time multitasking domains. COGNET is a general framework for decomposing and understanding human information processing in this important class of human-computer systems. The framework has applicability far beyond the Air ASW case used in this research to test and apply it. COGNET is also a powerful tool for cognitive task analysis that can be used for modeling, interpreting, and analyzing human performance in RTMT computer-based environments. As noted earlier, this includes virtually all tactical naval decision-making roles. The COGNET modeling techniques provide a novel integration of two previously unrelated cognitive analysis methods, the GOMS notation of Card, Moran and Newell, and the blackboard architecture that originally arose from the 1970's HEARSAY research program. In linking these two powerful notations, COGNET is able to represent features of both goal-directed and data-directed reasoning processes, and to relate them uniformly to a stream of human-computer transactions.

Along with the notational formalisms that constitute the COGNET description language, a new methodology for analyzing cognitive processes has been developed. This methodology incorporates a range of techniques in a layered approach that parallels the layered structure of the COGNET framework. In the methodology, timelines of key-stroke level human-computer interactions are developed and then analyzed with a verbal, question-answering, protocol technique to segment the timeline into perceived segments of task activity and to develop a vocabulary of mental representations of the domain. These two (the protocol and vocabulary) approaches are used to develop an initial blackboard structure. The task timelines and verbal protocol data are then iteratively analyzed to develop formal representations of each task type and to refine the blackboard structure. The overall methodology provides a novel and replicable way of integrating simulation, verbal protocol, and keystroke-level timeline data in a cognitive task analysis.

Validation and verification of any cognitive architecture or theory is a difficult undertaking (see Pylyshyn, 1984, for an excellent discussion of this subject). One specific key aspect of COGNET -- the RTMT flow of divided attention among tasks -- was experimentally investigated as a preliminary effort at validation. This is the aspect of COGNET which motivated its development, and which most differentiates it from other approaches (e.g., from MHP/GOMS).

A major feature of COGNET is that it is a context-based framework. The flow of attention and human-computer interaction are dependent on the evolving context of the human-computer problem. Initially this dependence is relatively low, but increases as the problem representation (as modeled by the blackboard contents) assumes specific patterns which trigger tasks, subgoals, and task-operator conditions. Moreover, the attention model in COGNET is driven by the trigger/condition elements in a specific model, which themselves represent pieces of knowledge that are held by humans who are experts in the domain. Thus, COGNET's attention framework can only be assessed through a specific model of a specific domain, and against the performance of human experts. The experimental data showed that in the operational domain of Air ASW, a COGNET-based model was able to predict 90% of instances of four tasks in the model. If the basis is included to consider cases where the model indicated a task shortly after the human initiated it, and/or those where instances of opportunities for task-execution were also indicated by the model and accepted as valid opportunities by domain experts but not acted upon by the human subjects in the experiments, then the 90% figure rises further. Even more impressive, perhaps, is the fact that the 90% figure represents <u>predicted</u> task-instances. That is, the model was able to <u>anticipate</u> the task execution 90% of the time for both post hoc verification and external validation data. This provides strong support for the validity of the approach, as well as for the likely usefulness of a COGNET model to support adaptive human-computer interaction.

The research has provided a novel concept and architecture for adaptive human-computer interaction. This concept builds on the pre-existing but highly abstract concept of embedded user models, and is based on the use of a COGNET model of a specific domain as an embedded model of an RTMT system user. The concept is to use the RTMT user model to:

- alert the real system user to situations where an attention shift is warranted,
- help prioritize competing attention demands,
- provide structuring guidance for specific tasks on which the user wishes to focus, and
- support automated and semi-automated execution of the task, including tailoring its performance to the momentary tactical context.

A COGNET-based model, because of its structure and organization, is able to support all of these functions. The research has gone further, however, by implementing a HC architecture that actually incorporates a specific COGNET model and performs the above functions. Thus, the architecture and its implementation have served as proof of concept for COGNET-based adaptive interface.

6.2. Operational Benefits and Results

As an applied research program, this effort has also addressed significant operational problems in the Naval Air ASW domain, and contributed to their long-term understanding and solution, including:

- a COGNET model of Air ASW mission management;
- an experimental environment for exploring human performance and decision-making in Air ASW; and
- an adaptive interface enhancement for the current TACCO workstation.

The COGNET cognitive task analysis tools were tested and refined through application to the human-computer interaction that occurs between an Air ASW Tactical Coordinator (TACCO) and his computer workstation. The results are a highly detailed model of the Air ASW mission management process, that represents the most complete set of expert operator knowledge captured in this domain. The mission management model provides a rich data base for future studies individual differences in TACCO performance and evaluating different tactics. Studies of these types could lead to improved methods for selection or evaluation of TACCOs or to better training techniques.

Early emphasis in this research was placed on developing an experimental environment that could support this and future research into human-computer interaction and problem solving in Air ASW. The experimental environment developed here has proven invaluable in all stages of the research:

- collecting baseline data on human-computer interaction in the domain;
- supporting manipulation, summary, and analysis of the data via playback and automatic data reporting;
- implementing the COGNET mission management model for validation and refinement; and
- implementing an adaptive mission management interface and seamlessly integrating it into the baseline TACCO interface.

This development environment makes future research in this domain more efficient, cost effective, and powerful.

Finally, the research has provided a working implementation of an adaptive mission management interface for the Air ASW TACCO. This interface provides both a proof of concept (as noted above) and points towards possible refinement and operational use in ASW systems. The fact that the adaptive interface is integrated into a realistic TACCO workstation emulation makes its applicability to operational Naval concerns direct and obvious. Of course, substantial advanced development would still be required before the implementation in the experimental environment could be implemented in fleet settings. It is nonetheless an example of how applied research can benefit real-world operational issues and theoretical concerns beneficially and simultaneously.

6.3 Additional Research Issues

Several new areas or issues of research emerged from the research reported here. This fall into two categories, those dealing with the adaptive interface, and those dealing with COGNET and its validation.

6.3.1 Adaptive Interface and Interaction Issues

Further work remains to be done in evaluating the adaptive interaction concept and in achieving a more robust range of functionality. The initial implementation demonstrated a range of adaptive interface functions using an architecture based on the COGNET model. However, further research and development is necessary to complete implementation and validation of the full set of mission management tasks.

A number of specific research issues were identified during implementation of the adaptive mission management interface. One issue involves when and how the task triggers should be evaluated. Initially, the triggers were checked cyclically, but this produced side effects which could not be controlled in a straightforward manner. The immediate problem was solved by checking the triggers only when appropriate. Additional research is needed on this, as the cyclic checking approach is theoretically preferable for maximum generality.

Another area for future research and expansion of the adaptive interface involves the conditions for removing tasks from the suggested strategies list. It should be noted that in the current implementation, recommendations sometimes remain in the suggested strategies window after a task has been performed. The interface is not yet sophisticated enough to recognize that the task has been performed and should be removed from the list. On the other hand, each task trigger has been associated with an 'untrigger condition,' which does remove this task from the list when the pattern that triggered it in the first place no longer exists on the blackboard. All tasks in the initial implementation use this 'untrigger' mechanism in lieu of a more sophisticated mechanism to recognize when the task is actually completed. Furthermore, some tasks can be performed more than once while others should only be performed once. The interface has to be sophisticated enough to determine not only when a task has been completed, but also whether the conditions exist in which the task could or should be performed again.

Many of the sensor patterns (e.g., prudential, convergence zone investigation) can be implemented in more than one way. For example, when two nearby sensors indicate a convergence zone contact, the convergence zone investigative pattern can be based on either sensor or some estimation procedure taking both sensors into account. In the current minimal version of the interface, only one version of each pattern is implemented. There is no mechanism by which the TACCO can direct the software to adapt the pattern to a different set of assumptions. This is an important subject for future research. In addition, there are several variant methods for adapting the canonical convergence zone investigative pattern to the current context. The interface currently only supports the most common, and doctrinally standard, approach. Exploration of mechanisms to allow alternative approaches including some that are adaptive to individual users is another issue for future research.

As discussed in Section 5 above, some tasks contain subgoals that are mutually exclusive. Although the current interface does not perform any subtasks that are not appropriate for the current context, it has no way of conveying this to the the TACCO. This, too, must be a subject for future research.

6.3.2 COGNET Elaboration and Further Validation Issues

One major area for further research is the need for more extensive validation of COGNET, particularly aspects other than the attention flow aspect considered here. Additional validation data and research is needed to validate both the representation and the underlying concept of cognitive tasks and their interaction with the blackboard representation. The blackboard as a basis for integrative task representation also requires further validation. There is some implicit validation of these constructs in the validation discussed in Section 3 above, because the attention flow aspect of the model builds on these constructs. Still, the COGNET architecture, which deals with large segments of goal-directed problem solving in a overall process which is data directed and opportunistic, is a novel one. Major existing cognitive architectures such as the Model Human Processor (Card et al., 1983) or the generalized blackboard architecture (Hayes-Roth, 1983) tend to represent human problem solving either totally goal driven or totally opportunistic. The SOAR architecture of Newell and colleagues (Laird, Newell, and Rosenblum, 1987) is an exception and does seem compatible with COGNET, but further work is needed to relate these two.

A more detailed need for further COGNET elaboration research was identified during the validation analysis. There were two cases -- one in the post-hoc verification and one in the independent validation -- when a task was indicated by the model but not counted as a prediction because the subject performed the action just prior to the task trigger. Upon replay of these two cases, it became clear that there were changes in the tactical situation that the TACCOs acted on more quickly than the model. These cases point out areas of potential improvement in the model. What is required are elaborations of COGNET that provide more sophisticated rules for transformation of information on the blackboard, corresponding to the types of inferences an operator makes about new perceptual information. An example is calculation of location from two direct path contacts on different buoys that give bearing reversals. The current implementation of the model infers a locational hypothesis when the TACCO enters a 'designated fix' command. However, as the two cases above indicate, the TACCO shifts attention to the new task a locational hypothesis leads him to undertake, and only subsequently takes the direct action indicating his hypothesis. Implementation of this and other similar problems involves mathematical calculations integrating data from multiple sensors to provide the model with the capability to trigger tasks based on changes in the tactical situation prior to the TACCO acting on them. Not only will this added capability enhance the model's predictive power; it will also allow better decision aiding by pointing out opportunities for action to the TACCO. The aiding capabilities could be particularly helpful in fast-paced or complex situations when the TACCO cannot process all situational changes as quickly as they occur.

There are also other applications of embedded user models that are possible for COGNET, including for training and embedded training, and/or decision aiding. In addition, COGNET has substantial potential for application in knowledge elicitation and knowledge representation for artificial intelligence systems. Finally, COGNET as a cognitive architecture has potential applicability to the measurement of cognitive processes, since it provides a highly computable mechanism for real-time multi-tasking problem solving. Thus, this research has opened a potentially rich set of research questions for the future in addition to its concrete results.

REFERENCES

Bruce, B. and Newman, D. (1978) Interacting Plans. Cognitive Science, 2, 196-233.

Card, S., Moran, T. and Newell, A. (1983) The Psychology of Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Dehdashdi, Parto (1989) A Study of Mental Models Through Communication Between Users and Advisors Solving Software Manipulation Problems. Unpublished Doctoral Thesis, Drexel University.

Eisenberg, E. M (1990, April). Transcendence Through Organizing. Communications Research, 17, 2, 139-164.

Elkerton, J. and Palmiter, S.(1989). Designing help systems using a GOMS model. Part 1 - An information retrieval evaluation (Technical Report C4E-ONR-3). Ann Arbor: University of Michigan, Center for Ergonomics.

Finegold, A. (1984) The engineer's apprentice. In P.H. Watson and K.A. Prendergast Eds., The AI Business: The Commercial Artificial Intelligence. Cambridge, MA: MIT Press.

Hayes-Roth, B. (1983). A Blackboard Model of Control. Technical Report, Computer Science Department, Stanford University, Stanford, CA.

John, B., Rosenblum, P. & Newell, A. (1985) A Theory of Stimulus-Response Compatibility Applied to Human-Computer Interaction. Proceedings of CHI '85. 213-219. New York:ACM

Malone, T.W., (1990). Organizing information processing systems: Parallels between human organizations and computer systems. In S. Robertson, W. Zachary, & J. Black (Eds.), Cognition, Computing and Cooperation. Norwood, NJ: Ablex Press.

Nii, P.H., (1986) Blackboard Systems. The Blackborad Model of Problem Solving and the Evolution of Blackboard Architectures , AI Magazine, 7, 36.

Norman,D. (1983) Some Observations on Mental Models. In D. Gentner and A.L. Stevens (Eds.) Mental Models, Erlbaum, Hillsdale, NJ

Norman, D. (1983a) Design Rules Based on User Errors, Communications of the ACM.

Pylyshyn,Z. (1984) Computation and Cognition. Cambridge,MA: The MIT Press.

Rasmussen, J. (1986) Information Processing and Human-Machine Interfaces, Amsterdam North Holland

Robertson S., Zachary W. and Black J.,Eds. (1990) Cognition, Computing, and Cooperation, Norwood, NJ:Ablex Press.

Rouse, W. B., and Morris, N. M. (1986) On looking into the black box. Prospects and limits in the search for mental models. Psychological Bulletin, 100, 349-363.

Rouse , W. B. (1980) Systems Engineering Models of Human-Machine Interaction. New York: North Holland.

Rubin, K., Jones, P., Mitchell, C.(1988) OFMspert: Interference of Operator Intentions in Supervisory Control Using a Blackboard Architecture. IEEE Transactions on Systems, Man, and Cybernetics, 18, 618-636.

Schmidt, C. F. and Goodson, J. L. (1990) The Design of Cooperative Person-Machine Problem Solving Systems: A Methodology and Example. In S. Robertson, W. Zachary, & J. Black (Eds.), Cognition, Computing and Cooperation. Norwood, NJ: Ablex Press.

Selfridge, Oliver G.(1959) Pandemonium: A paradigm for learning. In Proceedings of the Symposium on the Mechanization of Thought Processes, 511-529.

Sheridan, T. B., and Johannsen, G. (1976). Monitoring Behavior and Supervisory Control. New York: Plenum.

Siegel A. and Wolf, J (1969) Man-machine simulation models.Psychosocial and Performance Interaction. New York:John Wiley & Sons.

Suchman, L. A. (1990) What is Human-Machine Interaction?, In S. Robertson, W. Zachary, & J Black (Eds.), Cognition, Computing and Cooperation. Norwood, NJ: Ablex Press.

Van Cott, H. P. (1984) From Control Systems to Knowledge Systems. In Human Factors, 26, No. 1, 115-122.

Wilson and Rutherford (1989) Mental Models. Theory and Application in Human Factors. Human Factors, 31, 617-634.

Woods, D. D., and Hollnagle, E. (1987) Mapping cognitive demands in complex problem-solving worlds. International Journal of Man-Machine Studies, 26, 257-275.

Zachary, W. (1986) A Cognitive Based Functional Taxonomy of Decision Support Techniques. In Human-Computer Interaction, 2, 25-63.

Zachary, W. (1988). Decision Support Systems. Designing to Extend the Cognitive Limits. In Handbook of Human-Computer Interaction, Martin Helander (Ed.), New York: North Holland, 997-1028.

Zachary W.W. and Zubritzky M.C. (1988). An Experimental environment and Laboratory for Studying Human Information Processing in Naval Air ASW (Technical Report 881020.8704), Blue Bell, PA: CHI systems, Inc.

Zachary,W., Ryder, J., & Zubritzky,M (1989). Information Processing Models of Human-Computer Interaction in Naval Air Anti-Submarine Warfare. Technical Report 891215.8704. Spring House, PA:CHI Systems.

DISTRIBUTION LIST

OUSDR (A)/R&A(E&LS)
Pentagon 3D129
Washington, DC 20301-3080

Dr. Bruce Barnes
Program Director, Div. Info
Robotics, & Intell. Systems
National Science Foundation
1800 G Street, N.W.
Washington, DC 20550

Mr. Charles Bates
Director, HE Division
USAF AAMRL/HE
WPAFB, OH 45433

Dr. Kenneth R. Boff
AAMRL/HE
WPAFB, OH 45433

Mr. Luiz Cabral
Naval Underwater Sys. Ctr.
Code 2212
Bldg. 1171/1
Newport, RI 02841

Cdr. Robert C. Carter USN
Naval Research Laboratory
Code 5532
Washington, DC 20375-5000

Dr. Marvin S. Cohen
Dec. Sci. Consortium, Inc.
1895 Preston White Drive
Suite 300
Reston, VA 22091

Dr. Stanley C. Collyer
Office of Naval Technology
Code 222
800 N. Quincy Street
Arlington, VA 22217-5000

Defense Tech. Info. Center
Cameron Station, Bldg 5
Alexandria, VA 22314
(2 copies)

Prof. Jay Elkerton
University of Michigan
Dept. of I & O Engineerg
Center for Ergonomics
1205 Beal Avenue
Ann Arbor, MI 48109-2117

Dr. Robert A. Fleming
Naval Ocean Systems Center
Human Factors Support Group
1411 South Fern Street
Arlington, VA 22202-2896

Dr. Alfred R. Fregly
AF Office of Sci. Res.
Life Sciences Directorate
Bldg. 410
Bolling AFB, DC 20332-6448

Mr. Jeffrey Grossman
Naval Ocean Systems Center
Code 4402
Bldg. 334
San Diego, CA 92152

Dr. Thomas E. Hanna
Naval Sub Med Res Lab
Submarine Systmes Dept.
Naval Submarine Base
Groton, CT 06349-5900

Dr. Edgar M. Johnson
Technical Director
US Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333-5600

Dr. William Rouse
Search Technology, Inc.
4725 Peachtree
Corners Circle
Suite 200
Norcross, GA 30092

Capt T. E. Jones, MSC, USN
Aviation Medicine & Human
Performance (Code 404)
Naval Medical R & D Cmd
National Capital Region
Bethesda, MD 21814-5044

Dr. Michael Kaplan,
Director, Office Basic Res
US Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333-5600

Mr. Keith Kramer
Naval Research Laboratory
Code 5532
Washington, DC 20375-5000

Dr. Richard Loutitt
National Science Foundation
Division of Behavioral &
Neural Sciences
1800 G Street, N. W.
Washington, DC 20550

Capt. William Moroney, USN
Naval Air Development Ctr.
Code 602
Warminster, PA 18974

Naval Aerospace Medical
Research Laboratory
Sensory Division Code 23
Pensacola, FL 32508

Commanding Officer
Naval Air Systems Command
Crew Station Design
NAVAIR 5313
Washington, DC 20361

Commanding Officer
Naval Biodynamics Lab
Michoud Station
Box 29407
New Orleans, LA 70189

Commanding Officer
Navy Health Research Ctr.
P.O. Box 85122
San Diego, CA 92138

Commanding Officer
Navy Personnel R&D Center
San Diego, CA 92152-6800

Director
Technical Information Div.
Code 2627
Naval Research Laboratory
Washington, DC 20375-5000

US Naval Test Center
Aircrew Systems Branch
Systems Engineering Test
Directorate
Patuxent River, MD 20670

Commanding Officer
Naval Weapons Center
Human Factors Branch
Code 3152
China Lake, CA 93555

Dr. Michael Letsky
Office of the Chief of
Naval Operations
OP-01B7
Washington, DC 20350

Office of the Chief of
Naval Operations
OP-933D3
Washington, DC 20350-2000

Office of Naval Research
Perceptual Science Program
Code 1142PS
800 N. Quincy Street
Arlington, VA 22217-5000
(3 copies)

Dr. Jesse Orlansky
Inst. for Defense Analyses
1801 N. Beauregard Street
Alexandria, VA 22311

Dr. E. Salas
Head, Human Factors Div.
Naval Training Systems Ctr
12350 Research Parkway
Orlando, FL 32826-3224

Dr. H. Wallace Sinaiko
Manpower Research &
Advisory Services
Smithsonian Institution
801 N. Pitt St. Suite 120
Alexandria, VA 22314-1713

Mr. James G. Smith
Office of Naval Research
Code 121
800 N. Quincy Street
Arlington, VA 22217-5000

Cdr. S. Snyder
Office of Naval Technology
Code 222
800 N. Quincy Street
Arlington, VA 22217-5000

Dr. John Tangney
AF Office of Sci. Res.
Life Sciences Directorate
Bldg. 410
Bolling AFB, DC 20332-6448

Dr. Douglas Towne
University of Southern
California
Behavioral Technology
Laboratory
1845 South Elena Avenue
Fourth Floor
Redondo Beach, CA 90277

Dr. Harold P. Van Cott
Committee on Human Factors
NAS-National Res. Council
2101 Constitution Ave, N.W.
Washington, DC 20418

Dr. John Weisz
Technical Director
US Army Human
Engineering
Laboratory
Aberdeen
Proving Grounds
MD 21005

Prof. Christopher D. Wickens
Department of Psychology
University of Illinois
Champaign, IL 61801