A COMMON COCKPIT TRAINING SYSTEM

Jeremy Ludwig
Stottler Henke Associates, Inc.
San Mateo, California

CDR Henry Jackson
Naval Air Systems Command
Patuxent River, Maryland

ABSTRACT

The Naval Air Systems Command is introducing a new helicopter, the MH-60R (Romeo), for anti-submarine warfare and other uses. There are three crewmembers: the pilot, the airborne tactical officer (ATO), and a sensor operator (SO). The SO will be responsible for interpreting and managing a large variety of sensors. These sensors will be used to detect and track all ships, submarines, and possibly planes in the helicopter's vicinity, as well as friendly and enemy missiles and torpedoes. It is imperative to maximize the skills of both the ATO and SO, both operationally and tactically, as they must handle large amounts of information under stressful time critical situations.

However, carrying out anti-submarine warfare (ASW) at expert levels of proficiency requires extensive practice in real or simulated tactical situations under the guidance of experienced instructors. To train sensor operators more rapidly and cost-effectively, the Navy needs advanced software which complements traditional training methods. This software would provide a learning environment where students can practice ASW via free-play simulated tactical situations while receiving feedback and instruction customized to their experience and competency level.

The intelligent tutoring and simulation system software being developed duplicates the Common Cockpit Mission Display and includes free play simulation capability to maximize training. This intelligent tutoring system (ITS) will observe the operator's interaction with their equipment in the context of the ongoing mission situation, and provide appropriate reactive or proactive feedback to the operator in real time. The system is based on an individualized proficiency model of an operator, developed and updated throughout the operator's use of the ITS. This model will allow the software to provide feedback that is customized to the specific operator.

BIOGRAPHICAL SKETCHES

Jeremy Ludwig is an Artificial Intelligence Software Engineer at Stottler Henke Associates, Inc. (SHAI) where he has been focusing on the design and development of the Operator Machine Interface Aid training system for the MH-60R. Previously, Jeremy has held a variety of both research and software engineering positions at the University of Pittsburgh, Iowa State University, ISU Center for Non-Destructive Evaluation, and Engineering Animation. He received a B.S. degree in Computer Science with minors in Psychology and Philosophy from Iowa State University, and a M.S. degree in computer science, specializing in artificial intelligence, from the University of Pittsburgh.

CDR Henry Jackson is an Assistant Program Manager, (Training Systems) APM(TS) at Naval Air Systems Command PMA205 supporting PMA299 developing MH-60R Training System Programs. He is responsible to plan, procure, and manage innovative training solutions for the new MH-60R aircraft and all its highly complex and technical weapons systems and sensors. CDR Jackson has 15 years of operational experience deploying mainly on the Navy's Spruance Class Destroyers and Ticonderoga Class Cruisers as a LAMPS MKIII Detachment Officer in Charge and helicopter pilot. He has flown 6 different types of military aircraft and amassed over 3000 military flying hours. He has had tours in the Navy's training squadrons and as a Helicopter Squadron weapons and tactics officer.

A COMMON COCKPIT TRAINING SYSTEM

Jeremy Ludwig
Stottler Henke Associates, Inc.
San Mateo, California

CDR Henry Jackson
Naval Air Systems Command
Patuxent River, Maryland

INTRODUCTION

The MH-60R helicopter contains an unprecedented array of sensors under the control of a single Sensor Operator (SO). These include a dipping hydrophone, passive and active sonobuoys, electronic support measures, multi-mode radar, and forward looking infrared. Each one of these systems has different modes, settings, and methods of operation that must be calibrated for optimal performance in a particular setting.

For example, in the case of sonar systems several factors must be taken into account when determining settings. These include the current environment and its effects on the signal propagation paths; the physics of the signal propagation; threat tactical behavior and signal source characteristics; and the capabilities, limitations, and processing algorithms of the sensors and processing system.

All of these problems are exacerbated in the littoral environments where this helicopter will likely be used. In shallow water littoral environments, near or over-flown land masses, and a large number of commercial and neutral surface and airborne contacts significantly complicate the sensor optimization problem. Because of the clutter and multi-path effects in littoral environments, the SO needs to make good sensor choices to accurately separate threat from non-threat.

One solution to this problem would be to automatically set sensors' parameters for the SO. For a novice of any particular sensor, this would probably be an improvement. Unfortunately, such a solution is too rigid to allow for the superior performance of a SO skilled with the particular sensor. The result of using only a Tactical Decision Aid, or other advising technology, to set sensor parameters is to end up with middle of the road sensor results.

This tells us that optimal sensor performance requires expert users and poses a training problem for the Navy. Expert use of these sensors requires extensive training and practice of anti-submarine warfare (ASW) tactics, and familiarity with the SO controls of a Common Cockpit helicopter. Further, with the large number of available sensors it is likely for a student to be differentially familiar with the various sensors. That is, he might be an expert with passive sonobuoys but only mediocre with the dipping sonar. The Navy's solution to this training problem is based on several cooperating methods: traditional classroom instruction, computer-based training, and helicopter simulators.

The goal of this paper is to extend this solution to include an intelligent tutoring system (ITS) as part of this solution. Currently, we are developing a system called OMIA which integrates an ITS and a desktop-based Common Cockpit simulator. Ultimately, we hope the system will demonstrate that an ITS can be a valuable part of MH-60R training for both the SO and the Action Tactical Officer (ATO; copilot in the MH-60S).

We will briefly describe why an ITS is a valuable educational tool and give an overview of our ITS design. We will then show how an ITS might fit into the MH-60R training program. This discussion will be followed by a quick look at the system we are currently building for the Navy. Finally, we will talk about the realization of benefits from the Common Cockpit initiative and the future goals of this project.

INTELLIGENT TUTORING SYSTEMS

The idea, famously validated by Bloom (1984), is that students learn much better with one-to-one tutoring than in a classroom setting. Given that tutors could greatly enhance learning, the simple solution to the training problem discussed here is to use tutors to train SOs as thoroughly and quickly as possible. Unfortunately, the problem with this solution is quite obvious. The resources (e.g. financial; qualified personnel) required to carry out this task would be enormous, making this plan unfeasible. The current compromise is to provide classroom training and limited one-on-one instruction for helicopter pilots.

A goal of our ITS is to fill the tutoring gap in the above compromise. Basically, an ITS is designed to mimic and automate the relationship between a student and a tutor. By using an ITS to fill the role of a tutor, we hope to improve student learning without the exorbitant costs associated with human tutors. The tradeoff is that current ITSs are generally less effective than human tutors (Training and Personnel Systems Science and Technology Evaluation and Management Committee, 1996). That is, ITSs do not provide the degree of knowledge and flexibility given by interacting with a human tutor. Much research in the area of ITSs is aimed at minimizing this gap.

Intelligent Tutoring Systems are different from both computer-based training (CBT) and simulation. Computer based training is not adaptive to the individual weaknesses and strengths of the students; it is closer to being textbook than a teacher. Likewise, simulators provide an environment where the student can experiment, but do not actively teach the students. Often, simulators require human supervision to coach the students through exercises. Given this pairing, it seems straightforward that pairing an ITS with a simulator should lead to training results superior to those supplied by a simulator alone.

ROLE OF AN ITS IN THE MH-60R TRAINING PROGRAM

The original idea we proposed was using a real-time intelligent "coach" onboard the helicopter. This coach would have a model of the proficiencies of the current operator, which would have been constructed by monitoring the actions of the student and comparing them to the knowledge base supplied by expert SOs. During a mission if the coach deemed that a mistake was likely, a message would be displayed to the user suggesting the correct course of action.

Of course, if the coach is onboard the helicopter, it should also be part of the helicopter simulator as well. An earlier goal of the project was to create the ITS (coach) and connect it to a simulator. Since there was not a desktop based MH-60R simulator slated for development, we added a helicopter simulation to the training system. An interesting side effect of this move to a desktop-based simulator is that now the ITS can do much more than coach the student. For example, if a students are seen to do something wrong, they can receive a remediation to try and correct the flaw in their reasoning.

ITS DESIGN

The OMIA ITS consists of five major parts: the student model, instructor module, expert knowledge module, communication package, and enhancement module. This design leverages the success of SHAI's Tactical Action Officer ITS (Stottler & Vinkavich, 2000).

Expert Knowledge Module

Expert sensor knowledge is represented by a collection of individual principles, arranged in a hierarchy.
These principles are relatively low-level pieces of testable information created by domain specialists. Each principle may also contain material that should be presented to the student as enhancements or remediations. A sample hierarchy of principles is shown in Figure 1.

- Acoustic
  - Active Sonar
    - Dipping Sonar
      - Unintegrated for fast targets
      - Waveform selection
      - Configure before ping
      - Ping after configure

Figure 1. Sample principle hierarchy

An example of an individual principle is "Use the unintegrated setting on the dipping sonar for fast moving targets." In this case, comparing a student's actions to the experts is straightforward. A slightly more complicated principle is "Correct waveform selection for the dipping sonar." For this principle, the sonar settings suggested by the expert vary depending upon the environment and the expected target. A *decision tree* is used to represent the domain expert knowledge for principles such as this.

Decision trees are graphically constructed tree diagrams where at each node a question is asked. The next node is chosen based on the answer to the current question. By traversing through this tree, we eventually end up with the settings recommended by the expert. Figure 2 contains a partial decision tree for determining dipping sonar settings based on the target.

- What is the speed of the sub?
    - Slow
        - What is the distance of the submarine?
            - Short
                - Set …
            - Unknown
            - Long
    - Unknown
    - Fast

Figure 2. Partial decision tree

Student Model

The student model attempts to track the current state of the student's knowledge. This includes both what the student can see in the simulator and how much the student knows about each sensor domain. A student model contains the same principles present in the expert knowledge model with information on the student's familiarity with the principle. A simple example of a principle might be "Configure the dipping sonar before pinging," with the student model noting that this was completed successfully 93% of the time. As the student uses the ITS, the student model will come to more closely represent the actual knowledge of the student on each of the various principles.

Instructor Module (Assessment)

It is the job of the instructor module to compare the current situation with one of the cases in the expert knowledge module. Based on the principles required to formulate the expert solution (in our case sonar settings) the instructor module can decide if it is likely the student will fail or succeed based on his student model. If the student is likely to fail, the enhancement module is notified.

An additional job of the instructor module is to determine when a student has failed. If for example the expert module indicates that a student should search for short range targets before long range targets, but the student does the opposite, he has failed this principle. The instructor will then provide feedback to the student (remediation). This is additional information authored by a human expert with the intent of correcting the student's misconception. In the OMIA ITS, remediations are HTML files presented to the student.

To determine the performance of the student (either success or failure), the assessment module uses the student model to estimate what the student knows about a simulation. An example of this might be that the student was told they were looking for a fast moving submarine. This provides a starting point for the assessment module.

The authors of the ITS content create *finite state automata* (FSA) that describe what principles are active in a given situation. So, given that we know we are 1) looking for a submarine and 2) the submarine is moving at a high speed, what should the response of the student be?

Let us assume that the FSA were authored in such a way that both "Unintegrated for fast targets" and "Configure before dipping" were both active. Both of these principles would be sent to the Enhancement Module (below) as candidates for display.

After a student performs an action, pinging the dipper in the running example, a different an FSA might check the student's waveform selection against that of the expert waveform selection (Figure 2). If the settings are correct, the student's percentage correct on waveform selection would increase. If wrong, his percentage on this principle would decrease and the remediation attached to this principle would be displayed.

Enhancement Module

An enhancement is a one-line information text display on the multi-functional display. The goal of an enhancement is similar to that of a coach; to enhance learning by providing the student with enough guidance so they do not make a mistake. This differs from the instructor module that provides correction only after a mistake is made.

The instructor module requests an enhancement on every principle that is applicable to the current situation, but which the student has not demonstrated proficiency on. However, given the limited screen space allotted to enhancements and the fact that we do not wish to overload the SOs with information, only one enhancement is displayed at a time. The goal of the enhancement module is to ensure that the enhancements are displayed in order of importance, and that when displayed they are still relevant to the current situation.

Communication Package

The SO does not operate in a vacuum. While most of the ITS is developed with the idea of the SO interacting with the environment, the SO also interacts with the rest of the helicopter crewmen and perhaps a Tactical Decision Aid (TDA) of some sort.

To simulate this aspect of the SOs training, we introduced a communication package. This allows the ITS to display pre-generated information and prompting displays. An example of information might be "ATO: We have arrived at fly-to-point 1. Dip in accordance with the TDA". A prompt asks a question of the TDA to test their knowledge. For example, after pinging the dipping sonar the SO responds to multiple-choice questions about the results of the ping.

The communication package both provides a more realistic training environment and allows the Assessment module additional opportunities to correct mistakes made by the SO. If an SO misdiagnoses a sonar image, authored remediation material could be displayed which helps the student to correctly read the sonar image in future.

OMIA DESKTOP TRAINING SYSTEM

The OMIA System is made up of three interconnected parts: ITS (discussed above), simulator, and keybuilder. The simulator allows the student to perform "free-play" scenarios with a graphical user interface that matches that of the Common Cockpit Mission Display as closely as possible. This simulator communicates with the ITS to ensure the student model is up to date. The ITS then provides the student with enhancements and remediations as well as simulating discussions with other crew members. Finally, the Keybuilder software is used by instructors to ensure that as the Common Cockpit keysets change so does the simulator interface.

Simulation

The simulator provides an engaging interface between the student and the ITS. A student interacts with the simulator through a computer re-creation of the controls onboard the MH-60R. Additional interface components allow for the communication package of the ITS to have dialogs with the student.

In addition to being an interface, this component also simulates the environment surrounding the helicopters using *scenarios*. Scenarios are authored using a visual editing tool, and determine the elements of the simulation. Sonar returns from pinging depend not only upon the settings chosen by the SO, but also on the environment and target submarine settings in the scenario file.

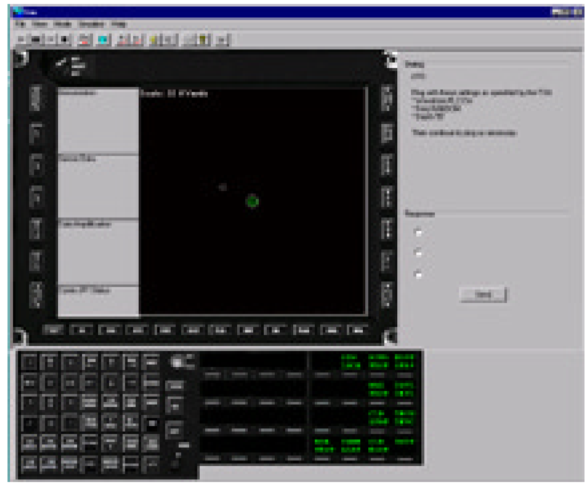

Figure 3. Simulation interface

Other entities, such as an enemy submarine or a friendly ship, can have agendas of their own. For instance, a submarine can be assigned the behavior "flee on detection." Even the fleeing behavior itself is customizable.

Of course, an SO might add objects such as sonobuoys to the simulation in real time. The flip side is that destroyed/sunk objects would be removed from the simulation as well. The ability to act upon the simulation and have it respond gives the students freedom to do the right thing, or to make mistakes. Either way, the system then has a better model of the student than before, and can use this to improve his learning experience.

Keybuilder

Intuitively, it seems that a computer-based training tool would still be quite useful even if the program interface is not the exact same as the interface of the helicopter. That is, it is the functionality that is important to learn, not the interface. As it turns out, this is not the case. Navy instructors reported that students had much less confidence in, and did not like to use, outdated training software (personal communication, 2001). Basically, they (the students) view outdated systems as a waste of their time. This lack of motivation is likely to decrease the effectiveness of any training system (Bransford, Brown, & Cocking, 1999).

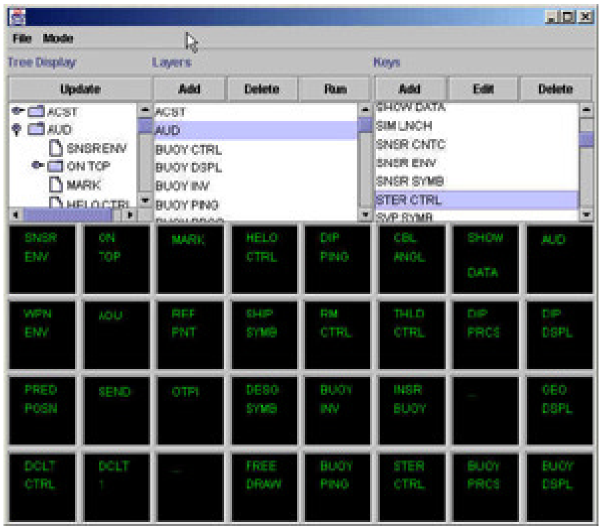

Figure 4. Snapshot of Keybuilder

This poses a problem for the OMIA system since the set of programmable keys used in the helicopter has been changing throughout the life of this project and are likely to keep changing. The solution was to design software that easily allows non-programmers to create, delete, and re-arrange the programmable keys (Figure 4). For example, if a key has a function (perhaps it brings up a menu) and is moved to a new location it will still perform its function at the new location. This program does have limitations however. If the instructor adds a new key, there is no way to place functionality behind the key without additional work from a computer programmer. While this only ensures the face validity of the programmable keys, it offers a way to ensure that the training system looks up to date without additional funding. This increases the chances of the system being well accepted by students.

COMMON COCKPIT BENEFITS

Throughout this project, we have concentrated on the helicopter capabilities that would be required by the sensor operator. Recently, we moved towards making the simulation interface more customizable to allow the system to simulate the interfaces of both the SO and ATO setups. This creates the added benefit of the system being immediately applicable to other Common Cockpit helicopter training. For instance, we are currently working with the MH-60S Fleet Introduction Team to determine to what degree their copilot training goals coincide with those of the MH-60R ATO. Since many aspects of the two roles are similar (e.g. they both work with fly-to points), this system promises to realize some of the pooling of resources enabled by the Common Cockpit initiative.

CONCLUSION

The complexity and number of the sensors under control of the SO on the MH-60R helicopter poses a difficult training task for the Navy. We discussed why intelligent tutoring systems are a promising technology for improving SO training and we have provided an overview of the system we are currently building for the Navy. Of significant importance is the idea of a computerized "coach" which helps students to learn an unknown, or little known, procedure correctly without first doing it incorrectly. Finally, we discussed that by taking advantage of the Common Cockpit, the developed system is useful for MH-60S training as well as MH-60R, thereby realizing some of the Common Cockpit benefits.

REFERENCES

Bloom, B. S., (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. Educational Researcher, 13(6): 4-16.

Bransford, J. D., Brown, A. L., & Cocking, R. R. (Eds.) (1999). How People Learn: Brain, Mind, Experience, and School. Washington D. C.: National Academy Press.

Stottler, R. H., & Vinkavich, M. (2000). Tactical action officer intelligent tutoring system (TAO ITS).

Training and Personnel Systems Science and Technology Evaluation and Management Committee. (1996). Functional Area Analysis of Intelligent Computer-Assisted Instruction. [WWW Page]. URL http://train.galaxyscientific.com/icaipage/litlfaa/litlfaa.htm