KNOWLEDGE-BASED SIMULATION: AN APPROACH TO INTELLIGENT OPPONENT MODELING FOR TRAINING TACTICAL DECISIONMAKING

Azad M. Madni, Ph.D
Perceptronics, Inc.
21111 Erwin Street
Woodland Hills, CA 91367-3713

Robert Ahlers, Ph.D
Human Factors Division, Code 711
Naval Training Systems Center
Orlando, FL 32813-7100

Yee-yeen Chu, Ph.D
Perceptronics, Inc.
21111 Erwin Street
Woodland Hills, CA 91367

ABSTRACT

Modern weapon systems have greatly expanded the range of options that can be exercised by trained tactical decisionmakers. However, tactical training environments today are unable to create the different opponent behaviors necessary to challenge the decisionmaking skills of tactical commanders. Since the use of human opponents is clearly not cost-effective, this training requirement falls under the purview of computer-based simulation. This paper presents a knowledge-based simulation approach for tactical adversary modeling along with an interactive user interface that allows non-programmers to modify simulation models on-line. A laboratory application that addresses a set of training objectives appropriate for surface warfare officer training is also included. The suggested approach is directly applicable to meeting current training simulation requirements generated by the surface Navy. Both the simulation approach and the software implementation are upward-compatible with the modeling of coordinated adversaries and supporting team members.

INTRODUCTION

The ability of modern weapon systems to engage multiple, sophisticated targets is providing a strong impetus for incorporating similar capabilities into related training environments. However, incorporating the different opponent behaviors for challenging modern tactical decisionmakers requires additional sophistication in today's tactical training environments. The use of human controllers continues to be cost-ineffective. Currently, instructors and device operators cannot efficiently represent realistic behavior of multiple targets. This is due to many factors, including conflicting training responsibilities, time overloading and varied familiarity with enemy tactics for the broad range of platforms generally required for training. Consequently, this requirement is a good candidate to be addressed by software models. However, there are several drawbacks to current software implementations of tactical threats/targets. These are outlined in Table 1.

TABLE 1
CURRENT IMPLEMENTATION DRAWBACKS

- Current software implementations of automated targets
  - do not represent decisionmaking behaviors of enemy tacticians (no planning functions; only reaction to situation)
  - do not take into account training objectives of trainee
  - involve lengthy software update cycle

Recent developments in Artificial Intelligence (AI) and cognitive modeling make it possible to develop knowledge-based techniques for modeling and controlling the opponent's behavior. This paper describes such an approach. It presents an intelligent opponent model that responds to trainee's actions in accord with training goals and inferred trainee deficiencies. Specifically, the opponent model consists of both domain-dependent knowledge and general problem-solving heuristics, thus allowing it to initiate and control tactics, maneuvers, and subsystem operations.

KNOWLEDGE-BASED SIMULATION OF TACTICAL ADVERSARIES

Enemy platforms that exhibit complex, purposeful behaviors are necessary to challenge tactical decisionmakers. However, the behavior of automated targets, has, heretofore, been driven primarily by analytical models or by a table of pre-enumerated behavior options, resulting in opponent behaviors that are either purely time and trajectory based, or capable of reacting to the prevailing tactical situations only with little provision for proactive tactical training.

Knowledge-based expert systems technology has opened up a new dimension in tactical simulation in general, and tactical adversary simulation in particular. Specifically, expert systems technology can be harnessed to model and emulate the behavior of an enemy commander who is in tactical control of an adversary platform. In other words, knowledge-based expert systems technology can be exploited to simulate an "embedded enemy tactician." The capabilities inherent in the use of knowledge-based expert simulation of adversary behaviors are summarized in Table 2.

An analysis of the tactical decision making task reveals the following important features:

- While tactical engagements do have generic phases, tactical behaviors tend to be opportunistic, subject to satisfying tactical doctrine and rules of engagement.
- Incoming information goes through a process of fusion and aggregation prior to producing useful intelligence.

- Tactical time horizons produce high time-stress and mental burden.

TABLE 2
ADVANTAGES OF KNOWLEDGE-BASED ADVERSARY SIMULATION

- Controllable computer-based opponent behavior
  - reasonable tactics and doctrines
  - rules of engagement
- Inspectable behaviors
  - decisionmaking processes
  - rationale/assumptions behind decisions
- Provision for realtime software update and knowledge base maintenance
  - opponent knowledge base
  - friendly characteristics knowledge base
  - *scenario knowledge base*
  - terrain, weather, geopolitical database
- Usable by computer-naive tactical domain experts and instructors
  - graphical (iconic) interfaces with graphical editors
  - "what if and why" facilities

The tactical environment is generally characterized by time-varying data and event-driven tactical strategies and decisions. The tactical decisionmaking task can be simply characterized as a real-time problem that requires continuous reasoning in the face of time-varying, uncertain data, changing constraints and evolving objectives. To this end, the one desirable approach to implementing controllable intelligent opponents is the use of a modeling framework that represents the task knowledge in a modular, hierarchical, conceptually transparent manner.

A particular class of expert systems architectures that is well-suited for modeling human performance in tactical decision making tasks is the "blackboard model" (9, 10). The blackboard model is both a problem-solving metaphor and a particular rule-based modeling framework appropriate for problems requiring continuous "real-time" reasoning under poorly defined conditions. Based on the metaphor of "experts" sitting around a blackboard, the blackboard model coordinates the activities of a number of different knowledge sources (i.e., "experts" or "specialists") by providing a global data base (i.e., blackboard) among them. The knowledge sources (KSs) cooperate with each other via the blackboard. The partial solutions that are produced by the knowledge sources to the problem under consideration are posted on the blackboard.

The blackboard itself is divided into a number of abstraction levels, each containing hypotheses for partial solutions which are linked to other levels by logical relationships. A monitor process controls access to these hypotheses and inspects any changes to notify "interested" knowledge sources.

In the blackboard model, each KS is independent of the other KSs, i.e., no KS knows which or how many other KSs exist. In general, a KS monitors a specific level of the blackboard for those changes or conditions that are relevant to its area of expertise. When these changes or conditions occur, the KS requests a processing turn by placing an event-related item on the agenda. This agenda is a list of possible processing events from which the scheduler chooses the one most likely to lead the farthest in solving the problem. The decisionmaking process of the scheduler is controlled by an invariant set of rules about problem solving and a dynamic goal structure. As the solution progresses, the goal structure adapts to focus attention in a data-directed fashion. The chosen processing event is passed back to its KS for execution

The blackboard idea has been extended to hierarchical models in which different concepts are assigned to different blackboards and knowledge sources mediate information among the blackboards in the form of expectations, supports, refutations, etc. Nii (14) has presented a summary of blackboard models and applications.

NAVAL SURFACE WARFARE APPLICATION

The knowledge-based simulation approach has been used to develop an intelligent adversary simulation *within the context of Naval surface warfare.* The simulation scenario is staged against the backdrop of ongoing limited hostilities between a Blue and a Red Naval vessel. It commences with the Tactical Action Officer onboard a Blue Cruiser maintaining watch. Intelligence assets have located the Red fleet in the vicinity of the Blue cruiser. The specific assignment for the Blue Cruiser is a Red Destroyer that has been designated a "prime target."

The TAO onboard the Blue Cruiser has been given specific guidance and is assumed to be aware of the local rules of engagement (ROEs). Table 3 provides illustrative ROEs for this application.

TABLE 3
EXAMPLE OF LOCAL RULES OF ENGAGEMENT

- Attack to disable/destroy any prime target that commits a hostile act
- Hostile acts include enemy behaviors such as:
  - Fire Control System emissions associated with surface-to-surface missile (SSM) for a period greater than or equal to 1 minute
- Attack prime targets only if there is a high certainty that prime targets have:
  - Identified Blue capabilities and
  - Identified Blue location and
  - Have engaged in concerted attack
- Concerted attack implies that more than 2 SSMs have been launched against Blue platform

The intelligent opponent simulation is constructed against the foregoing scenario backdrop. The simulation *employs the blackboard model-based problem-*solving software architecture presented earlier. The specific instantiation of this architecture is shown in Figure 1.

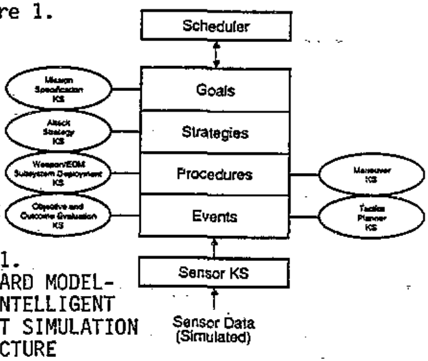


FIGURE 1.
BLACKBOARD MODEL-BASED INTELLIGENT OPPONENT SIMULATION ARCHITECTURE

As seen in Figure 1, several different KSs are involved in representing the knowledge of the tactical opponent. Based on the discussions with Navy Tactical Officers, these KSs are partitioned into tactically meaningful classes that roughly correspond to the generic categories of situation assessment, objective formulation, tactics planning and tactics execution. Some examples of the tactical rules associated with the KS are given in Table 4. Note that the rules consist of a set of conditions and actions or procedures. Details of the software architecture are described in Chu and Shane (^{5}).

TABLE 4
EXAMPLES OF TACTICAL RULES

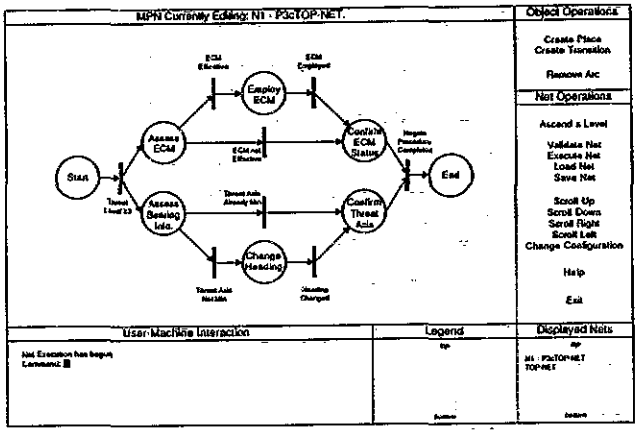
IF	THE BLACKBOARD CONTAINS A NODE: TRY CHANGING SPEED, and THE OBJECTIVE IS: TO NEGATE A POSSIBLE ATTACK
THEN	TAKE PRECAUTION BY MOVING FASTER.
IF	THE OBJECTIVE IS: TO NEGATE POSSIBLE ATTACK and IT HAS BEEN FOCUSED ON FOR AT LEAST 4 CYCLES
THEN	POST INFERENCE NODES ON THE BLACKBOAHD TO i) TRY CHANGING HEADING and ii) TRY LAUNCHING A CHAFF
IF	THE BLACKBOARD CONTAINS A NODE: TRY LAUNCHING A CHAFF, and THE FOCUSED EVENT IS: TO NEGATE A POSSIBLE ATTACK
THEN	if RED SHIP CAN TURN BEFORE A MISSILE COULD ARRIVE then MAKE AN ACTION: LAUNCHA CHAFF
IF	THE FOCUSED ACTION IS : LAUNCH A CHAFF
THEN	COMPUTE THE BEST HEADING FOR RED TO TURN, CALCULATE THE CHANGE IN HEADING TO BEST HEADING (HEADING-CHANGE) if HEADING-CHANGE = 0 then TAKE ACTION TO LAUNCH CHAFF, if HEADING-CHANGE ≥ 20 then ADJUST RED'S HEADING TO BEST HEADING (with high confidence) and LAUNCH CHAFF if HEADING-CHANGE ≤ 20 then ADJUST RED'S HEADING TO BEST HEADING (with little confidence) and LAUNCH CHAFF
IF	A BLUE'S MCS IS ACTIVATED
THEN	POST INFERENCE NODES ON THE BLACKBOARD i) OBJECTIVE IS: TO NEGATE A POSSIBLE ATTACK and ii) TRY CHANGING RED'S SPEED

INTELLIGENT OPPONENT BEHAVIOR MODIFICATION/UPDATE

For training tactical decisionmaking, it is necessary that the student be exposed to a host of opponent behaviors. To this end, the training system contains interface facilities for the instructor, or other subject matter expert, to modify the adversary's tactical rule base in support of required training objectives.

The tactical behavior of the platforms used in training simulations cannot be "coded and forgotten." Not only does our knowledge about opposition tactics change, but the tactics themselves also change. In addition, the emphasis that is placed on specific aspects of the tactics also changes. This results in a requirement to create tactical targets that are not necessarily "tactically accurate" but which are designed to fulfill or complement specific training objectives. To this end, an instructor is provided with interface facilities for straightforward creation or modification of tactical targets.

To provide proper interface facilities, we chose to use the direct, graphical manipulation capabilities to allow an instructor to specify variations of data and procedures that govern an adversary's behavior. Figure 2 shows a sample screen of the knowledge base browsing mode that supports this process. The opponent's tactical knowledge at the highest level is represented using Modified Petri Nets (MPNs), (^{11}; ^{12}) based upon the Petri nets formalism (^{15}; ^{1}). MPNs are used to model opponent task execution at those levels of abstraction where there is a high degree of concurrent task-related activities and tactical decisions (^{20}; ^{11}). An MPN is graphically represented by two types of nodes: places (circles) and transitions (vertical bars). The nodes are connected by directed arcs (arrows). The places, transitions and directed arcs represent the static properties of the network. The dynamic property is represented by a token (dot) that resides in a place. When a token resides in a place, the activity associated with the place is said to be "ongoing." When the event associated with the output transition occurs (i.e., the transition "fires"), the ongoing activity ceases and the token propagates across the transition to the output places (i.e., activities). Thus, the control knowledge associated with a specific intelligent adversary procedure can be parsimoniously represented with the MPNs framework.



Net Execution has begun
Command:

FIGURE 2.
GRAPHICAL INTERFACE FOR VIEWING/EDITING
OPPONENT'S "NEGATE ATTACK PROCEDURE"

The declarative knowledge in the opponent simulation is represented using frames (^{13}). Specifically, the approach employs the Frame Representation Language (^{17}) which features multiple inheritance paths between frames. A frame is associated with each place and transition in the network. Procedural knowledge is embedded within specific frame slots using production rules. The separation of control knowledge from the declarative and procedural knowledge provides an executable opponent simulation that is straightforward to generate and easy to inspect and modify. This multi-formalism approach to representing opponent simulation knowledge makes for a parsimonious, executable tactical expert model. The graphics interface uses multiple windows and a mouse pointing device to allow the instructor to browse a knowledge base. Instructors can also view the execution of specific tactics during real-time opponent decisionmaking in much the same way a student using GUIDON-WATCH (^{16}) can view the reasoning process during diagnostic problem-solving.

The rule-based representation of KSs provides a convenient basis for constructing an interface for modifying tactical rules. This interface provides the necessary flexibility for the instructor to modify the expertise and rules used by the simulated tactical adversary. In other words, this capability is constructed as a knowledge acquisition interface that allows the expert instructor to specify variations of rules and procedures that govern an adversary's behavior. These modifications result in alternative adversary performance models. Use or avoidance of particular tactics or systems, and the creation of particular tactical situations to which a trainee must respond, are all accomplished via the knowledge base interface facilities.

Current interface facilities allow the user to browse through and selectively review portions of an adversary's tactical rule sets including rules of engagement (ROE) and rules for situation assessment and planning.

Current Capabilities of the Intelligent Adversary Simulation

The current capabilities of the intelligent opponent simulation are summarized in Table 5. The trainee has total control of the Blue platform via the trainee station. Specifically, the trainee can select tactics, specify maneuvers and deploy weapons or sensory assets. The trainee maintains situation awareness by monitoring the geopolitical situation, the various subsystems, and the message window, or alternatively by querying the appropriate knowledge bases via the commands menu.

TABLE 5
CURRENT CAPABILITIES OF
INTELLIGENT ADVERSARY SIMULATION

- Trainee can control Blue ship
  - Tactics
  - Maneuver
  - Subsystems
- Trainee monitors
  - Geopolitical situation
  - Subsystems
  - Commands
  - Message window
- Computer opponent can control Red ship
  - Tactics
  - Maneuvers
  - Subsystems, including missiles, sensors and countermeasures
- Effects on World Model of
  - Red maneuver, tactical moves and deployment of assets
  - Trainee's decisions and "non-decisions"

The automated opponent has total control of the Red ship's tactics, maneuvers and the various subsystems including missiles, sensors and countermeasures. The manner in which the opponent controls these assets can be changed interactively by the instructor in the knowledge base browsing/editing mode.

All actions taken by the trainee and the intelligent computer opponent update the world model. For example, Red maneuvers, tactical moves and deployment of assets all update the world view. Similarly, the trainee's decisions and "non-decisions" in terms of use of friendly assets, and the execution of specific tactics also update the world model.

The prototype intelligent adversary was done on the Symbolics 3670 using the Heuristic Control System, a rule-based derivative of AGE (2) system developed at Stanford University. A simulated environment of one-on-one engagement was also developed to codify and test expert rules for tactical planning. The combined system (5) has demonstrated that (1) the rule-based representation of tactical knowledge provides a convenient basis for constructing the interface facilities for modifying the tactical rules of the computer adversary, and (2) the blackboard model-based architecture provides an effective framework for representing tactical decisionmaking behavior. In simulation exercises, the modification of adversary parameters can be based on a library of instructional outlines designed to focus attention of the trainee on the appropriate skills prior to playing the modified scenarios. In this setting, the trainee is able to exercise and tune specific target skills individually and in relevant combinations. It appears that such a training, authoring and delivery environment is not only motivating to the user but can also be expected to greatly shorten the software update cycle involved in supporting specific types of tactical training exercises.

SUMMARY

Future Research Directions

Future research will involve additional work in regards to three major themes: the graphical interface that the computer-naive instructor was to specify/modify tactical targets, the simulation of multiple coordinated adversaries, and the simulation of supporting teams.

Intelligent Opponent Simulation Language. The case has been made above for having an interactive interface for specifying/modifying the characteristics and behavior of tactical opponents. The extensive use of graphics and abstract symbology at varying levels of abstraction is expected to make the interface more natural to use, and the targets even easier to access and modify. The graphical interface provides the necessary flexibility for the instructor/experimenter to modify the expertise and rules used by the simulated tactical adversary.

Multiple Coordinated Adversaries. The work described above for single targets will be extended to allow the simulation of opposing forces consisting of multiple platforms. The modeling of such composite forces is extremely complex since the behavior of each platform must be specified in terms of its relationships to all other platforms within the force. The complexity of each individual model therefore is also a function of the constraints that result from the need to maintain coordination with other platforms. To achieve the goals of field implementation and online instructor modification of tactical targets, requires a method for reducing this complexity. Work in AI has addressed the distributed decisionmaking problem (3; 6; 7; 8; 18; 19) and will be applied to the implementation of a generic mechanism for dealing with the multiple constraint environment of coordinated tactical adversaries.

Simulation of Supporting Teams. The products from the above two efforts (i.e., the graphical interface for tactical targets and multiple coordinated adversaries) will be applied to the simulation of supporting teams. For example, in submarine combat systems team trainers, the activities associated with the sonar, radio rooms, and helm control are now performed by training device operators. A development effort will be undertaken to simulate these teams' activities in order to attempt to reduce the manning requirements for the trainers associated with the current and next generation fire control systems. The approach will again be to provide a high-level graphical interface for implementing the actual models. The goal will be to provide not only the simulation models but also the tools to allow the models to be updated in the field.

Conclusions

Current implementations of automated targets are deficient because they do not represent the decision making behaviors of enemy tacticians. Specifically, automated targets react to the prevailing tactical situation only; they do not engage in higher level tactical planning functions. The approach taken in this paper addresses this deficiency by modeling the decision making behavior of the opposing tactical commander. In addition, an interface is described which has been designed to facilitate the maintenance of the opponent's tactical rule bases. Plans are underway to extend this overall approach to the modeling of coordinated tactical adversaries and supporting teams.

REFERENCES

1. Agerwala, T. Putting Petri Nets to Work. Computer, pp. 85-94, December 1978.
2. Aiello, A., Bock, C., Nii, H.P., and White, W.C. AGE Reference Manual, Technical Manual. Computer Science Department, Stanford University, CA, October 1985.
3. Cammarata, S., McArthur, D., and Steeb, R. Strategies of cooperation in distributed problem-solving. In Proceedings of the Eighth International Joint Conference on Artificial Intelligence, pages 767-770, The International Joint Conferences on Artificial Intelligence, Karlsruhe, West Germany, August 1983.
4. Chu, Y. and Madni, A.M. Network-Based Human Behavior Modeling for Adaptive Training. Proceedings of 1985 International Conference on Systems, Man, and Cybernetics, Tucson, Arizona, November 1985.
5. Chu, Y. and Shane, D. Intelligent Opponent Simulation for Tactical Decision Making. Final Report, PFTR-1120-11/86, Perceptronics, Inc., Woodland Hills, CA, December 1986.
6. Corkill, D.D. An Organization Approach to Planning in Distributed Problem Solving Systems. Technical Report TR 80-13, Department of Computer and Information Science, University of Massachusetts, Amherst, Massachusetts, May 1980.
7. Dalkey, N.C. Group Decision Making. Technical Report UCLA-ENG-7749, School of Applied Science, University of California, Los Angeles, July 1977.
8. Davis, R. A model for planning in a multi-agent environment: steps toward principles for teamwork. Working paper 217, Massachusetts Institute of Technology AI Laboratory, October 1981.
9. Erman, L.D., Hayes-Roth, F., Lesser, V.R., and Reddy, R.R. The Hearsay-II Speech Understanding System: Integrating Knowledge to Resolve Uncertainty. ACM Computing Survey, 12: 213-253, 1980.
10. Hayes-Roth, B. The Blackboard Architecture: A General Framework for Problem Solving? Technical Report HPP-83-30, Heuristic Programming Project, Computer Science Department, Stanford University, May 1983.
11. Madni, A.M. and Chu, Y. Network-Model-Guided Expert Knowledge Elicitation, Proceedings of the 1985 IEEE International Conference on Systems, Man, and Cybernetics, Tucson, Arizona, November 1985.
12. Madni, A.M., Chu, Y., Purcell, D. and Brenner, M. Design for Maintainability with Modified Petri Nets (MPNs): Shipboard Propulsion System Application. Perceptronics Report PFTR-1125-84-11, prepared for Office of Naval Research, November 1984.
13. Minsky, M. "A Framework for Representing Knowledge," in the Psychology of Computer Vision, P.H. Winston (ed.), McGraw-Hill Book Company, New York, 1975.
14. Nii, P.H. Blackboard Systems: The Blackboard Model of Problem-Solving and the Evolution of Blackboard Architectures (Part One), The AI Magazine, Summer 1986.
15. Peterson, J.L. Petri Net Theory and the Modeling of Systems, Inglewood Cliff: NJ: Prentice Hall, 1980.
16. Richer, M.H. and Clancey, W.J. GUIDON-Watch: A Graphics Interface for Viewing a Knowledge Base, Report No. STAN-CS-85-1068, Department of Computer Science, Stanford University, August 1985.
17. Roberts, R.B. and Goldstein, I.P. The FRL Primer, Report AIM-408, Artificial Intelligence Laboratory, MIT, Cambridge, MA, July 1977.
18. Rosenschein, J.S. and Genesereth, M.R. Communication and Cooperation. Technical Report HPP-4-5, Heuristic Programming Project, Computer Science Department, Stanford University, October 1984.
19. Smith, R.G. and Davis, R. Frameworks for cooperation in distributed problem solving. IEEE Transactions on Systems, Man, and Cybernetics, SMC-11(I):61-70, January 1981.
20. Zisman, M.D. Use of Production Systems for Modeling Asynchronous Concurrent Processes in D.A. Waterman and F. Hayes-Roth (eds.), Pattern-Directed Inference Systems, Academic Press, New York, NY, 1985, pp. 53-68.

ABOUT THE AUTHORS

Dr. Azad Madni is a Senior Vice-President and Chief Scientist of Perceptronics. He received his Ph.D. from the University of California, Los Angeles in Engineering and Computer Science, specializing in Human-Computer Systems. His current research interests are in intelligent tutoring systems, cockpit automation, adaptive decision aiding, and intelligent man-machine interfaces. He is a member of IEEE, AAAI, AIAA and ADPA.

Dr. Robert Ahlers is a Research Psychologist with the Human Factors Division of the Naval Training Systems Center. He received his Ph.D. from North Carolina State in Human Factors. His recent work has addressed the use of artificial intelligence techniques to provide performance diagnosis and tactical adversary capabilities.

Dr. Yee-yeen Chu heads the Knowledge Systems Group at Perceptronics. He received his Ph.D. in Mechanical Engineering with emphasis in human-computer interaction from the University of Illinois. His research interests are in intelligent tutoring and aiding systems, robotics, and intelligent interfaces. He is a member of IEEE, AAAI and ACM.