Validation of Human Behavioral Models

Avelino J. Gonzalez
Electrical and Computer Engineering Department
University of Central Florida
Orlando, FL 32816-2450
ajg@ece.engr.ucf.edu

Abstract

Validation of human behavioral models, such as those used to represent hostile and/or friendly forces in training simulations is an issue that is gaining importance, as the military depends on such training methods more and more. However, this introduces new difficulties because of the dynamic nature of these models and the need to use experts to judge their validity. As a result, this paper discusses some conceptual approaches to carry out this task. These are based on comparing the behavior of the model to that of an expert, while the latter behaves normally in a simulated environment, under the same conditions as perceived by the model.

1 Introduction

The field of intelligent systems has matured to the point where significant research is now being focused on modeling human behavior. Earlier research work, mostly in the form of expert systems, concentrated on developing means of representing and manipulating deep but narrow and specialized knowledge efficiently and effectively. Their objective was to provide expert advice in the process of solving difficult and specialized problems. This objective has generally been met successfully, with research in expert systems currently having shifted to more efficient means of knowledge acquisition, and system validation and verification.

Expert systems, however, have three significant limitations in our quest for a truly intelligent system: 1) they are, by definition, generally quite limited in breadth, 2) they typically do not directly control anything, and 3) they do not learn easily. Truly intelligent systems must be able to interact with their real world environment in all of its breadth and scope, as well as learn from it on a continuous or semi-continuous basis. Additionally, they must be able to directly control their own actions if they are to survive in the real world.

Much of the current research effort has centered on developing intelligent systems that can do these things, either as robots in the physical world, or as computer generated entities in a simulation of the real world. The latter are most often used to assist in simulation-based training, but have applications in entertainment and control.

To be successful in modeling human behavior, researchers must look to the human mind and attempt to replicate its mode of operation. Doing this at the biological level has been nearly impossible to do, as the inner workings of the human brain and how that translates into intelligence are not well understood. Furthermore, it is beyond the realm of modern science to be able to reproduce brain tissue artificially.

Simulating the neural function in a digital computer using connectionist approaches has shown significant success when applied to certain classes of problems. They enjoy the distinct advantage of being able to learn when presented with examples. However, neural networks have not shown success in demonstrating general intelligence by themselves. Furthermore, it is clear even to the most naïve that the human brain does not employ the complex mathematical algorithms which neural networks use to learn.

Modeling human behavior at the procedural level has shown significant promise. Through introspection, humans can and have been able to identify several high level techniques used to solve some problems, especially that of interacting with our environment in order to live, thrive and survive in it. Certainly, solving problems in the same high level way as humans do is a step in the right direction.

The field of human behavior representation research is basically divided in two parts: those working in robotics as the embodiment of these models, and those working in intelligent simulated entities for training.

Intelligent robotic research has generally focused on giving robots the ability to handle fairly low level tasks, such as route planning, obstacle avoidance, and searching for specific objects in a closed environment. A second aspect of this work has been in reinforcement learning – learning by experiencing the environment in much the same way that a child learns not to touch a hot stove after she gets burned once.

Research in simulated intelligent entities, on the other hand, has focused on modeling a higher level of behavior, such as that used in tactical behavior modeling. This different focus is largely a result of the need for displaying tactically correct behavior in a wartime simulation. While route planning and obstacle avoidance have also been areas of intensive work, learning has not been a major issue up until recently. More of interest has been to develop efficient and effective models that truly represent tactical behavior at a minimum of cost to develop as well as to execute.

Human behavioral models, known in the military simulation and training community as *Computer Generated Forces* (CGF), have been successfully incorporated in several significant simulation training systems. However, there is a serious need on the part of these users to be able to validate the behaviors demonstrated in order to ensure a sound training process. But such validations are not easy. This paper provides an insight into the type of procedures that would be required to adequately validate these human behavioral models. However, prior to that discussion, a review of the most common and popular means of representing CGF systems would be appropriate as a way to set up the discussion on their validation.

2 CGF Implementation Techniques

When a human tactical expert is asked what he would do under certain circumstances, the response is typically framed as a conditional. "If this was green and that was blue, then I would turn left". Thus, the most intuitive as

well as popular means of representing tactical human behavior is through the use of rules. However, rules have the drawback that they are quite myopic in scope. To develop a system with any kind of tactical realism, a large number of rules need to be developed and executed, as the numerous conditions resulting from the many variations can translate into an explosive number of rules for even relatively simple tasks. This is not efficient. Furthermore, gaps in the behavior can be easily left uncovered. Whereas these gaps can be easily filled with new rules, it is a makeshift process that does not have natural closure. Experts systems suffer from the same deficiency. But the domain of expert systems, being more limited and the inputs more predictable, can easily tolerate the situation.

Because of its intuition and popularity, most of the early systems that implemented CGF's were based on rules. Golovcsenko [1987] discusses some Air Force prototype systems that exhibit certain amount of autonomy in the simulated agents. One in particular, a special version of the Air Force's TEMPO force planning war game system, uses rule-based techniques to replace one of the human teams involved in the game.

One notable CGF system is the *State Operator and Results* system (SOAR) [Laird, 1987; Tambe, 1995]. SOAR takes a goal-oriented approach in which goals and sub-goals are generated and plans to reach them are formulated and executed. These plans are in effect until the goals are reached, at which point they are replaced by new goals that address the situation. However, SOAR is based on the rule-based paradigm, which, as mentioned before, has many disadvantages.

Another popular technique used in CGF systems has been *Finite State Machines* (FSMs). FSM's have been used to implement goals or desired states in the behavior of the AIP [Dean, 1996]. These states are represented as C-Language functions. Three major FSM-based CGF systems are the *Close Combat Tactical Trainer* (CCTT) [Ourston, 1995], ModSAF [Calder, 1993], and IST-SAF [Smith, 1992]. These systems all employ FSMs as the representational paradigm. The knowledge found on FSM's does not necessarily aggregate all the related tasks, actions, and things to look out for in a self-contained module. This makes their formalization somewhat difficult from the conceptual standpoint. Some FSM-based systems allow for the control of one entity by more than one FSM at the same time. This can be dangerous in that an incorrect behavior can be easily displayed.

Other, less popular, alternative representation and reasoning paradigms such as model-based, constraint-based or case-based reasoning, although promising in some respects, (see [Borning, 1977; Castillo, 1991; Catsimpoolas, 1992]) are not "natural" for this form of knowledge since they do not easily capture the heuristics involved in tactical behavior representation.

Another common approach to the AIP problem has been to use blackboard architectures to represent and organize the knowledge. One implementation [Chu, 1986] uses separate knowledge sources to carry out tasks such as situation assessment, planning, formulation of objectives, and execution of the plan. This system also implements adaptive training so that the AIP can modify its action according to the skills on which the trainee needs to concentrate. The system uses a modified version of Petri Nets to represent instructional knowledge. Work carried out in the form of maneuver decision aids at the Naval Undersea Warfare Center [Benjamin, 1993] has also employed a blackboard architecture. The objective of this research is to assist the submarine approach officer in determining the most appropriate maneuver to carry out to counter an existing threat, or to accomplish a mission.

Another approach has come from cognitive science researchers [Wieland, 1992; Zubritsky, 1989; Zachary, 1989]. These efforts do not directly address AIP's, but rather, the closely related problem of cognitive modeling of the human decision-making process. Their efforts also make use of a blackboard architecture. The COGNET representation language [Zachary, 1992] uses a task-based approach based on the GOMS concept [Card, 1983; Olsen, 1990], in an opportunistic reasoning system. In this approach, all actions are defined as tasks to be performed by the AIP. The definition of each task includes a trigger condition that indicates the situation that must be present for that task to compete for activation with other similarly triggered tasks. The use of blackboard system, however, introduces a high overhead and much added complexity.

As can be seen, there are several means of representing the knowledge required to model human behavior as it applies to tactics. The problem remains how to validate the behavior models in a way that makes all the different representational paradigms transparent. The next section discusses some conceptual approaches to the problem.

3 Potential Validation Techniques for Human Behavior Models

Since by definition these models are designed to simulate human behavior, it becomes clear that they must be compared to actual human behavior. Validation of the more traditional expert systems is really no different, as these attempt to model the problem solving ability of human experts. Expert systems have traditionally been validated using a suite of test cases whose solution by human domain experts is known ahead of time. However, these tests are generally static in nature – provide the system with a set of inputs and obtain its response, then check it against the expert's response to the same inputs. Time is typically not part of the equation, unless it is already implicitly incorporated into the inputs (i.e., one input could represent a compilation of the history of one input variable). The techniques suggested by Abel [1997] provide effective and efficient means of generating good test cases based on the validation criteria specified for the intelligent system. The Turing Test approach proposed by Knauf [1998] is a promising way to incorporate the expert's opinion in a methodical fashion for time-independent problems and their time-independent solutions.

Validating human behavioral models, on the other hand, requires that time be explicitly included in the expression of the tactical behavior. Such behavior not only has to be correct, but also timely. Reacting to inputs correctly, but belatedly can result in the decision-maker's destruction in a battlefield. Furthermore, tactical behavior is usually composed of a sequence of decisions that are made as the situation develops interactively. Such developments are generally unpredictable and, therefore, it becomes nearly impossible to provide test inputs for them dynamically.

Certainly the test scenarios could be "discretized" by de-composing them into highly limited situations that would take the time out of the equation. However, several of these would have to be strung together in sequence in order to make the entire test scenario meaningful. This would be artificial, and the expert may have difficulty in

visualizing the actual situation when presented thusly. Furthermore, interaction would not be possible. Therefore, I do not believe this would be acceptable as a mainstream solution.

One alternative would be to observe the expert or expert team while he/they display tactical behavior, either in the real world, or in a simulation specially instrumented to obtain behavioral data. Certainly, using a simulation would make the task of data collection and interpretation much easier, at the cost of having to build a simulation. However, since the models are to be used in a simulation, it is likely that such an environment already exists.

Conceptually speaking, validation can be executed for these types of models by comparing the performance of the expert with that of the system while being subjected to the same initial inputs. The performance of each (the intelligent entity and the expert) can be represented as a sequence of data points for the observable variables over a period of time. Overlaying one on top of the other may provide some indication of validity for the system's performance.

A complete match between the expert's performance and the model's behavior would certainly justify validation of the model. Realistically, however, a significant amount of deviation may exist between the two performance records. While some deviations may be indicative of a serious discrepancy in behaviors, others may simply be a different and equally appropriate way of achieving the same goal. Thus, expert input may be necessary to determine what is correct and what is not correct. Alternatively, an intelligent system could be developed to perform this task of determining what is an acceptable deviation and what is not, but it would also ultimately have to be validated itself, and that would ultimately require human expertise.

Furthermore, due to the interactive nature of the model and the domain, the system being validated may make a different decision from what was made by the validating expert which, although correct, progresses into a different scenario. Consequently, the two performance records could no longer be adequately compared, as their situations may have diverged significantly enough to make them not relevant to each other.

In reality, none of the above techniques provide us with an effective and efficient means to validate the performance of human behavioral models. This leaves us in a quandary, until we take into consideration how the model was built in the first place.

Building the model in the traditional way – interviewing the subject matter experts (SME's) and building the model by hand from their response to the numerous queries made in these interviews, would in fact place us in this quandary. There would be little relationship between the means of model development and that of validation. However, by tying the means of building the model with the validation process, some of the obstacles described above may be overcome. This is described in the next section.

4 Learning by Observation of Expert Performance in a Simulation

Humans have the uncanny ability to learn certain tasks through mere observation of the task being performed by others. While physical tasks that involve motor skills do not fit under this definition (e.g., riding a bicycle, hitting a golf ball), cognitively intensive, or procedural tasks can be relatively easily learned in this way. Very often we hear people asking for examples of how to perform a task so they can see how it is done. If such is the case for humans, certainly machines can be made to do the same type of learning.

This idea was seized by Sidani [1994], who developed a system that learns how to drive an automobile by simply observing expert drivers operate a simulated automobile. The system observed the behavior of an expert when faced with a traffic light transition from red to green. It also observed the behavior when a pedestrian attempted to cross the street. Furthermore, it was able to correctly infer a behavior it had not previously seen when faced with both, a traffic light and a pedestrian on the road. Sidani compartmentalized the behaviors by training a set of neural network, each of which was called to control the system under specific circumstances. A symbolic reasoning system was used to determine which neural network was the one applicable for the specific situation.

Further work in the area is currently being carried out by Gonzalez, DeMara and Georgioupoulos [1998a, 1998b] in the tank warfare domain. Using a simulation as the observation environment, behavior is observed and modeled using Context-based reasoning (CxBR). For an explanation of CxBR, see [Gonzalez and Ahlers, 1995]. Supported by the U. S. Army under the Inter-Vehicle Embedded Simulation for Training (INVEST) Science and Technology Objective, this project also extends the concept of learning through observation by including an on-line refinement option. See Bahr and DeMara [1996] for further information on the nature of the INVEST project.

4.1 On-Line Refinement and Validation

The concept of refinement involves improvement of an intelligent system during or after initial validation. The model being developed by Gonzalez, DeMara and Georgioupoulos (which was learned through observation) is intended to predict the behavior of a human combatant in a federated simulation. This is rather different from conventional CGFs that attempt to simply emulate actual entities in a general way. Prediction also carries a much heavier burden when it is regularly and continuously compared to actual behavior, as is the case in the application of the resulting model. However, it provides the opportunity to implement validation relatively easily.

A predictive model is required because an accurate prediction of actual human behavior would reduce the need for real-time on-air communications of the position of an actual vehicle in the field to all others in the simulation. Each live vehicle in the simulated exercise must be aware of the position of all other live, simulated and virtual vehicles in the simulated environment, which may or may not be the same as the physical environment where the vehicle is physically located. Rather than communicating its whereabouts constantly (which is expensive due to the small bandwidth available), each live vehicle has in its on-board computer a model of all other live and simulated vehicles. If the other vehicles all behave as modeled, the personnel in that vehicle can confidently predict its location and would need no update on its position. However, it is unrealistic that a perfect model could be found, in spite of the latest modeling and learning techniques. Each vehicle carries a model of itself in it on-board computer. It compares the actual position, speed and other observable actions with those predicted by the model. If in agreement, no action is necessary. However, if a discrepancy arises, all other models of this vehicle resident

in all the other vehicles (called "clone" models) are not correctly predicting its behavior. Thus, some corrective action must be initiated. Such action can be: 1) discarding the model and providing constant updates through communication, 2) modifying context in which the model is on an on-line basis, or 3) permanently changing the model to reflect reality. This last step can be referred to as refinement.

Therefore, a means to detect deviations becomes necessary. As such, an arbiter system must be developed to determine when the model no longer correctly predicts the behavior so that a correction is initiated. This arbiter, called the Difference Analysis Engine (DAE), basically serves to compare the behavior of the human with that of the model.

4.2 The DAE as a Validation Engine

The DAE is designed to ascertain when deviations between the human and the model are significant enough to warrant initiation of corrective action (i.e., communication sequence). However, it would be relatively easy to convert the DAE into a validation engine instead. The model and the human expert could be reacting to the simulated environment simultaneously, with the DAE continuously monitoring them for discrepancies. Upon discovering a serious enough discrepancy, the DAE could note it and either continue, or modify the model so that it agrees with the human's behavior. Through the use of contexts as the basic behavioral control paradigm, the requisite action would merely be to suggest a context change in the model. This new suggested context would agree with the expert's action and the validation exercise could continue in a synchronized fashion. While this would represent external manipulation of the model, a team of experts could afterwards, in an after action review, determine whether the model should be permanently modified to reflect that change.

Alternatively, the change could simply be noted, recorded and presented to the panel of experts at an after action review. The drawback to this is that unless rectified, a discrepant decision by the model could serve to make the rest of the validation exercise irrelevant, as the model may be faced with situations that are different from that of the human.

5 Summary and Conclusion

It is clear that validation of human behavioral models introduce a new level of difficulty in the validation of intelligent systems. Conventional validation techniques, such as the ones used for expert systems, may not be effective for such a task. A promising alternative exists with the concept of learning and refining a model through observation of expert behavior. While this concept is relatively immature and requires significant further investigation, I feel that it represents a very viable approach to this very difficult problem of validating human behavior models.

References

[Abel, 1997] Abel, T. and Gonzalez, A. J., "Enlarging a Second Bottleneck: A Criteria-based Approach to Manage Expert System Validation Based on Test Cases", Proceedings of the 1997 Florida Artificial Intelligence Research Symposium, Daytona Beach, FL, May 1997.

[Bahr and DeMara, 1996] Bahr, H. A. and DeMara, R. F.: "A Concurrent Model Approach to Reduced Communication in Distributed Simulation" Proceedings of 15th Annual Workshop on Distributed Interactive Simulation, Orlando, FL, Sept. 1996.

[Benjamin, 1993] Benjamin, M., Viana, T., Corbett, K. and Silva, A., "The Maneuver Decision Aid: A Knowledge Based Object Oriented Decision Aid", Proceedings of the Sixth Florida Artificial Intelligence Research Symposium, April, 1993, pp. 57-61.

[Borning, 1977] Borning, A., "ThingLab - An Object-oriented System for Building Simulations Using Constraints," ACM Transactions on Programming, Languages and Systems, 3 (4) October, 1977, pp. 353 - 387.

[Calder, 1993] Calder, R., Smith, J., Coutemanche, A., Mar, J. M. F. and Ceranowicz, A., "ModSAF Behavior Simulation and Control", Proceedings of the Third Conference on Computer Generated Forces and Behavioral Representation, Orlando, FL, March 1993, pp. 347-356.

[Card, 1983] Card, S., Moran, T. and Newell, A., The Psychology of Human-Computer Interaction, Hillsdale, NJ: Lawrence Erlbaum Associates, 1983.

[Castillo, 1991] Castillo, D., "Toward a Paradigm for Modelling and Simulating Intelligent Agents," Doctoral Dissertation, University of Central Florida, December, 1991.

[Catsimpoolas, 1992] Catsimpoolas, N. and Marti, J., "Scripting Highly Autonomous Behavior Using Case-based Reasoning", Proceedings of the 25th Annual Simulation Symposium, Orlando, FL, April 1992, pp. 13-19.

[Chu, 1986]] Chu, Y. Y. and Shane, D., "Intelligent Opponent Simulation for Tactical Decision Making," Report PFTR-1120-11/86, sponsored by the Office of Naval Research and the Naval Training Systems Center, 1986.

[Dean, 1996] Dean, C. J., "Semantic Correlation of Behavior for the Interoperability of Heterogeneous Simulations", Master's Thesis, Department of Electrical and Computer Engineering, University of Central Florida, May 1996.

[Golovcsenko, 1987] Golovcsenko, I. V., "Applications of Artificial Intelligence in Military Training Simulations", Master's Degree Research Report, University of Central Florida, 1987.

[Gonzalez, DeMara and Georgiopoulos, 1998a] Gonzalez, A. J., DeMara, R. F. and Georgiopoulos, M. N., "Vehicle Model Generation and Optimization for Embedded Simulation", Proceedings of the Spring 1998 Simulation Interoperability Workshop, Orlando, FL March 1998.

[Gonzalez, DeMara and Georgiopoulos, 1998b] Gonzalez, A. J., DeMara, R. F. and Georgiopoulos, M. N., "Automating the CGF Model Development and

Refinement Process by Observing Expert Behavior in a Simulation", Proceedings of the Computer Generated Forces and Behavioral Representation Conference, Orlando, FL, May 1998.

[Gonzalez and Ahlers, 1995] Gonzalez, A. J., and Ahlers, R. H.: "Context-based Representation of Intelligent Behavior in Simulated Opponents" Proceedings of the 5th Conference in Computer Generated Forces and Behavior Representation, Orlando, FL, May 1995.

[Knauf, 1998] Knauf, R,. Jantke, K. P., Gonzalez, A. J., and Philippow, I., "Fundamental Considerations for Competence Assessment for Validation", Proceedings of the 11th International Florida Artificial Intelligence Research Society Conference, Sanibel Island, FL, May 1998, pp. 457-461.

[Laird, 1987] Laird, J. E., Newell, A. and Rosenbloom, P. S., "Soar: An Architecture for General Intelligence", Artificial Intelligence, 33(1), 1987, pp. 1-64.

[Olsen, 1990] Olsen, J. R. and Olsen, G. M., "The Growth of Cognitive Modeling in Human-Computer Interaction since GOMS", Human Computer Interaction, 5(2&3), 1990, pp. 221-265.

[Ourston, 1995] Ourston, D., Blanchard, D., Chandler, E. and Loh, E., "From CIS to Software", Proceedings of the Fifth Conference on Computer Generated Forces and Behavioral Representation, Orlando, FL, May 1995, pp. 275-285.

[Sidani, 1886] Sidani, T. A. and Gonzalez, A. J.: "IASKNOT: A Simulation-based, Object-oriented Framework for the Acquisition of Implicit Expert Knowledge" Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Vancouver, Canada, October 1995.

[Smith, 1992] Smith, S. and Petty, M., "Controlling Autonomous Behavior in Real-Time Simulation," Proceedings of the Southeastern Simulation Conference, Pensacola, FL, 1992, pp. 27-40.

[Tambe, 1995] Tambe, M., Johnson, W. L., Jones, R. M., Koss, F., Laird, J. E. and Rosenbloom, P. S., "Intelligent Agents for Interactive Simulation Environments", AI Magazine, Spring, 1995, pp. 15-39.

[Wieland, 1992] Weiland, M. Z., Cooke, B., and Peterson, B., "Designing and Implementing Decision Aids for a Complex Environment Using Goal Hierarchies," Proceedings of the Human Factors Society 36th Annual Meeting, 1992.

[Zachary, 1989] Zachary, W. W., "A Context-based Model of Attention Switching in Computer Human Interaction Domain," Proceedings of the Human Factors Society 33rd Annual Meeting, 1989, pp. 286-290.

[Zachary, 1992] Zachary, W., Ryder, J. and Ross, L., "Intelligent Human-Computer Interaction in Real Time, Multi-tasking Process Control and Monitoring Systems", in M. Helander and M. Nagamachi (Eds.) Human Factors in Design for Manufacturability, New York: Taylor and Francis, 1992, pp. 377-402.

[Zubritsky, 1989] Zubritsky, M. C., Zachary, W. W. and Ryder, J. M., "Constructing and Applying Cognitive Models to Mission Management Problems in Air Anti-Submarine Warfare," Proceedings of the Human Factors Society 33rd Annual Meeting, 1989, pp. 129-133.