ON THE FIDELITY OF SAFs: CAN PERFORMANCE DATA HELP?

Amy Henninger, Avelino Gonzalez, William Gerber, Michael Georgiopoulos, and Ron DeMara
School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL

ABSTRACT

A recent report developed by the National Research Council (NRC) for the Defense Modeling and Simulation Office (DMSO) encourages the use of real world, war-gaming, and laboratory data in support of the development and validation of human behavioral models for military simulations. Also encouraged in this report is the use of interdisciplinary teams embracing the disciplines of the psychological, computer, and military sciences to create such models. This paper describes the use of an artificial intelligence modeling framework, observational learning, to support these objectives. This framework combines the research methods of experimental psychology with the machine learning methods of computer science to develop behavioral models from data generated by military experts participating in live and/or simulated exercises.

To date, research has demonstrated that behavioral models developed through this framework can be integrated into popular Semi-Automated Force (SAF) systems to enhance their performance. However, there has been no known investigation as to what the benefits of this approach are with respect to behavioral model fidelity. This paper introduces the interdisciplinary nature of observational learning by briefly surveying its history with respect to computer science and psychology and by illustrating how it can be used in conjunction with military experts. Next, this paper examines experimental evidence to determine whether a significant difference exists between SAF performance and human performance for a low-level, skill task. Finally, this paper demonstrates how behavioral models developed through human performance data generated by military SMEs can be used in conventional SAF systems to make SAF performance more "human-like".

ABOUT THE AUTHORS

Amy Henninger is a doctoral candidate in computer engineering at the University of Central Florida (UCF). She has worked in the military simulation community for 10 years. Her research interests lie in the areas of artificial intelligence, mathematical modeling, and simulation.

Avelino Gonzalez received his B.S.E.E and M.S.E.E from the University of Miami in 1973 and 1974 and his Ph.D. from the University of Pittsburgh in 1979. He is currently a professor in the School of Electrical Engineering and Computer Science (SEECS) at UCF, specializing in human behavior representation.

William Gerber received his B.S.E.S. from the United States Air Force Academy. He has an M.S.E. from the University of California at Los Angeles and a M.S.Cp.E. from UCF. He is currently a Ph.D. candidate in computer engineering at UCF.

Michael Georgiopoulos is an Associate Professor at SEECS UCF. His research interests lie in the areas of neural networks, fuzzy logic and genetic algorithms and the applications of these technologies in cognitive modeling, signal processing and electromagnetics. He has published over a hundred papers in scientific journals and conferences.

Ron DeMara is a full-time faculty member in the SEECS at UCF. Dr. DeMara received the B.S.E.E.. degree from Lehigh University in 1987, the M.S.E.E. degree from the University of Maryland in 1989, and a Ph.D. degree in Computer Engineering from the University of Southern California in 1992.

ON THE FIDELITY OF SAFs: CAN PERFORMANCE DATA HELP?

Amy Henninger, Avelino Gonzalez, William Gerber, Michael Georgiopoulos, and Ron DeMara
School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL

INTRODUCTION

A recent report developed by the National Research Council (NRC) for the Defense Modeling and Simulation Office (DMSO) encourages the use of real world, war-gaming, and laboratory data in support of the development and validation of human behavioral models for military simulations. Also encouraged in this report is the use of interdisciplinary teams embracing the disciplines of the psychological, computer, and military sciences to create such models. This paper describes the use of an artificial intelligence modeling framework, observational learning, to support these objectives.

To date, research has demonstrated that behavioral models developed through this framework can be integrated into popular Semi-Automated Force (SAF) systems to enhance their performance (Henninger et al, 1999). However, there has been no known investigation as to what the benefits of this approach are with respect to behavioral model fidelity. This paper introduces experimental evidence to determine whether a distinguishable difference exists between SAF performance and human performance for a low-level, closed-loop, skill task and demonstrates how behavioral models developed through human performance data generated by military SMEs can be used in conventional SAF systems to make SAF performance more "human-like".

SAF Background

Currently, distributed battlefield simulations use computerized behavioral models of combatants to serve as opponents against whom trainees can fight, or as friendly forces with whom the trainees can fight. These computer-controlled combatants are known as Computer Generated Forces (CGFs) and usually generate multiple battlefield entities (e.g., tanks, aircraft or infantry) using computer algorithms rather than a human crew to control the actions of those entities. The software used to control the behavior of the CGF entities is flexible enough to react to what is happening in the simulated battle and produce intelligent and realistic actions. The behavior of the CGF may be generated by a human operator assisted by software, in which case the class of CGF is referred to as a semi-automated force (SAF), or they may be generated completely by software, in which case they are known as autonomous forces (AFs). At a minimum, the behavior generated by CGFs should be feasible and doctrinally correct. For example, CGF behaviors should be able to emulate the use of formations in orders, identify and occupy a variety of tactical positions (e.g., fighting positions, hull down positions, turret down positions, etc.), and plan reasonable routes.

Traditionally, SAF behaviors have been implemented in procedural languages (e.g., Ada or C) and organized around state transition constructs such as finite state machines (FSMs) or Petri Nets. For example, a SAF behavior such as “Occupy a Battle Position” might be constructed around states such as: “Start FSM”, “Travel”, “Calculate Position”, “Move Into Position”, and “End FSM”. Any one of these states, in turn, could be (1) an embedded FSM, (2) a simple function call representing some low-level primitive action, or (3) any combination of the two. This type of organization provides a useful means for structuring and communicating the intricacies of the behavior.

SAF Knowledge Acquisition

Although the behavior of SAF entities can be quite complex, they merely provide feasible behavior. Since these behavioral models are fashioned entirely by doctrine that is acquired from declarative knowledge (e.g., manuals and interviews), they simply emulate standard procedures (Ourston, et al., 1995; Calder, et al, 1993; and Smith and Petty, 1992), and provide no representation for the intrinsic performance characteristics that make live entities unique from one another. For example, the current SAF behavioral models used in DIS exercises may simulate the movement of a vehicle to a given location by some standard movement model

(Smith, 1994), but they do not individualize that movement method by either assigning or simulating human performance characteristics to it (e.g., tendency to hug the side of the road, propensity to maintain speed above speed limit, etc.). Thus, behavioral models fashioned entirely by doctrine are often characterized as yielding responses that are "too perfect" or "not human-like". However, the fact that these behaviors are not consistently humanlike in no way suggests that these behaviors are simplistic. Prevalent SAF systems have integrated hundreds of thousands of lines of code to successfully simulate the command and control hierarchy of a military unit and its operation on the battlefield. By providing planned behaviors (e.g., "Conduct a Tactical Road March", "Attack By Fire", "Service Station Re-supply", etc.), situational awareness and assessment capabilities, and reactive behaviors (e.g., "Breach a Minefield", "Call for Indirect Fire", "Actions on Contact", etc.), they have accomplished their objective of providing adequate friendly and enemy forces to populate the battlefield.

A recent report by the National Research Council (Pew and Mavor, 1998) has recognized that the doctrinally driven behavioral models used by the SAF community like "many social and organizational theories are expressed as verbal descriptions of institutional, social, and political processes" (p. 15). However, because modeling human behavior on the battlefield is a highly non-linear, dynamic task and because it is difficult to determine a consistent set of predictions through the sole use of verbal models, there has been a call by the National Research Council to develop and validate human behavioral models through the use of human performance data. According to Pew and Mavor, central to the successful execution of this task is the use of interdisciplinary teams embracing the disciplines of psychological, computer, and military sciences. For those with a specific interest in this topic, the next section reviews the interdisciplinary nature of observational learning by briefly surveying its history with respect to computer science and psychology and by illustrating how it can be used in conjunction with military experts. Others may proceed to the section following the next with no loss of critical information.

MERGING THE DISCIPLINES AND USING PERFORMANCE DATA

Essential to the problem of how to make behavior models more human-like is a definition of what the term "behavior" represents. The term behavior is usually used to refer to anything a human being does: that is, any act or succession of acts that are objectively observable. A key word in the definition of behavior is "observable." Behavior refers to movement, activity, or action that is overt.

Broadly defined, psychology is the study of the behavior of intelligent organisms (Hilgard, et al, 1979). Since artificial intelligence (AI) and machine learning (ML) are concerned with the automation of intelligent behavior (Lugar, 1993), it would make sense then that they borrow and/or learn from the annals of psychology. This, in fact, is evidenced in a variety of AI/MLconcepts. For instance, a number of first generation knowledge acquisition systems embraced as a part of AI's history (Shaw and Gaines, 1987; Boose and Bradshaw, 1987) make use of "repertory grids", which are based on the theory of personal constructs developed by George Kelly, a well-known clinical psychologist. In his research, Kelly employed introspective methods to build templates that modeled an individual's perception of the world. Introspection involves examining one's own thoughts and feelings and making inferences based on this examination, and, at one time, it was the prevalent form for obtaining psychological data.

As the discipline of psychology matured, it was agreed that introspection was not entirely open to scientific analysis and that the workings of the mind could only be known through the observation of the behavior it controls. As a result, in modern times, experimental psychologists seldom use introspection. Instead, they rely on a method of collecting data called observation. Observation, or naturalistic observation, has always been a tool of science and is a well-respected and scientific method for accumulating information. This transition has also been evidenced in the AI/ML communities where computational intelligence and fuzzy modeling approaches using observational data are becoming increasingly popular behavior modeling paradigms.

The difference between introspective and observational techniques for acquiring knowledge has also been an issue of concern to the SAF community. Because SAF models are driven by human behaviors, the value of the simulation whether for training, operations, or policy analysis depends in large part on the validity of the human knowledge on which those models are based. All SAF developers recognize the importance of acquiring valid tactical knowledge from creditable subject matter experts (SMEs). Some developers (Velt, 1993; Deutsch, 1993) assert that while acquiring introspective knowledge from SMEs may be beneficial and/or necessary, it alone is not sufficient. This view is corroborated by Pew and Mavor (1998) who assert that, “humans, unassisted by a computer, are simply not good at thinking through the implications of such complexity” (p. 15). According to these researchers, conventional CGF knowledge acquisition techniques capture the SME's perceived decision, and this perception may not be consistent with the SME's actual decision. In other words, the decisions people think they would make are not necessarily identical to the decisions people would actually make. To better understand this phenomenon, Deutsch offers a concrete domain example taken from Dreyfus and Dreyfus:

“In the Air Force, instructor pilots teach beginning pilots how to scan their instruments. The instructor pilots teach the rule for instrument scanning that they themselves were taught and, as far as they know, still use. At one point, however, Air Force psychologists (DeMaio et al. 1976) studied the eye movements of the instructors during simulated flight and found, to everyone’s surprise, that the instructor pilots were not following the rule they were teaching. In fact, as far as the psychologist could determine, they were not following any rule at all. ... the instructors, after years of experience, had learned to scan the instruments in flexible and situationally appropriate ways.”

Deutsch continues by asserting that introspection is unreliable for determining the deliberative aspects of performance and cites psychological literature which reports that during the process of introspection, influential stimuli are not only inaccurately reported, but that they are frequently missed entirely. Furthermore, he cites psychological literature which suggests that it is not unusual for subjects to be unable to report that a cognitive process has occurred at all, and that even responses are not always easily reported.

Ultimately, Deutsch suggests that distributed simulation environments and CGFs provide a unique framework in which to design scenarios and conduct experiments to explore skilled behaviors of domain experts. That is, by working with subject matter experts in a synthetic CGF environment, simulated scenarios which approximate the environment in which an expert interacts can be used to allow the expert to communicate by "doing" instead of by "reflecting". As a result, the models are based on human performance data acquired by observation instead of data acquired through introspection.

This paper builds on the recommendations and assertions offered by Pew and Mavor (1998) and Deutsch (1993). In the next section, the performance of a low-level skill task by a SME is compared with the performance of the same task by a SAF to determine whether a noticeable difference exists. Then, the section after the next demonstrates how the use of SME-generated performance data can be used to make conventional SAF performance more human-like.

COMPARISON OF HUMAN MOVEMENT TO SAF MOVEMENT MODEL

This section presents the method used to compare the near-term movement behavior of a ModSAF M1A2 entity with the movement behavior of a trained tank driver. Since maneuvering in the battlefield is a complex behavior depending on many factors, the problem was scoped to focus on the examination of the vehicle’s speed and orientation over one section of a road march route. Three scenarios were generated for the M1A2 entity and three scenarios were considered for the SME, who was a former Army officer with Armor experience. Also, the SME was provided ample opportunity to become familiar with the road march order parameters and the M1A2 table top driver simulator controls (see Figure 1).

Figure 1. Driver at M1A2 Table Top Simulator

The NE_Bosnia terrain database was used for the experiment, and the route used for the road march may be seen in Figure 2. Both the ModSAF entity and the human tank drivers started the route in identical locations with identical orientations and order parameters. Also, the performance of both the ModSAF entity and the human driver was evaluated over the same section of the route.

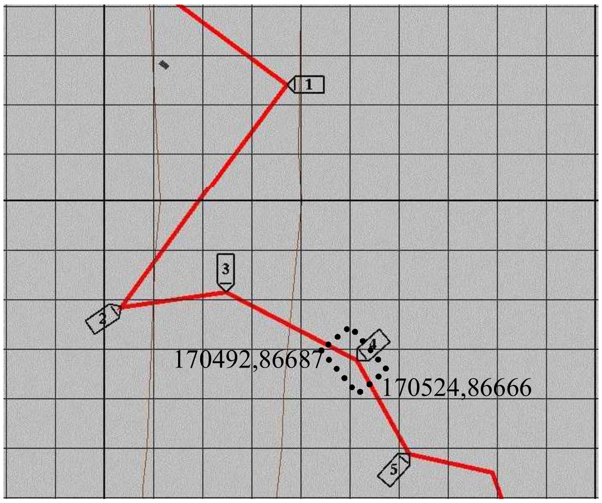


Figure 2. Route Used for Comparison

Figures 3 and 4 show the trajectory and speed data over the three runs made by the ModSAF entity and the SME, respectively. All data in all plots represent the vehicle state between the terrain database coordinates (170492,86687) and (170524,86666). The trajectory plots are presented relative to the route's center line, and both the trajectory and speed plots denote the point at which the way point change occurs[1].

General observations for the SAF entity's behavior include a drastic reduction in speed while approaching a turn and a drastic increase while departing the turn. Also, as evidenced in the trajectory plot of Figure 3, the SAF entity tends to take the turn through immoderate heading changes made over a relatively short period of time and shows little symmetry in its path relative to the way point. Alternatively, the SME generated human performance data shown in Figure 4 reveal a smoother, more continuous trajectory with greater symmetry about the way point. Further, in the corresponding speed plot, a general pattern of very little speed change is apparent.

[1] A way point change occurs in ModSAF when the entity's 3-dimensional euclidian distance is within 5 meters of the way point and a periodic update occurs. The way point change for SME data is derived by projecting two points equidistant from the way point along each road segment and then comparing the vehicle's position relative to each of these points. When the distance to the point projected on the departing side of the route is less than the distance to the point projected on the approaching side of the route, the way point is updated.

MODELING HUMAN PERFORMANCE DATA WITH NEURAL NETWORKS

The task of modeling human driving skills (e.g., as acceleration, steering, and vehicle following) with neural networks has been well-investigated (Pomerlau et al., 1995; Nechyba and Xu, 1997) in the autonomous learning and intelligent vehicle communities. A neural network is a collection of simple processors or nodes interconnected with each other that learn from examples and store the acquired knowledge in their interconnections, referred to as weights. Neural networks can solve a variety of problems related to non-linear regression and non-linear dynamic systems.

To model the SME's performance on the road march, a feed-forward architecture with back-propagation training was used to develop two networks. The first network estimates the change in the vehicle's speed and the second network estimates the change in the vehicle's heading. As indicated in Table 1, each of the networks consists of five inputs in the first layer, five nodes in the second layer, and a single dependent variable representing the response that is output by the last layer.

NN	Arch	Predictors	Resp
Speed	5-5-1	$Rwp_{t-1}, Rcl_{t-1}, Rs_{t-1}, HRab_{t-1}, HRbc_{t-1}$	ΔS_t
Heading	5-5-1	$Rwp_{t-1}, Rcl_{t-1}, Rs_{t-1}, HRab_{t-1}, HRbc_{t-1}$	$\Delta\theta_t$

Table 1. Architecture by Neural Network Type

The inputs shown in Table 1 were normalized according to equations (1) – (5) below. Fundamentally, the inputs for each of the networks were a function of the M1A2's state at the last simulation clock and how this state related to the road characteristics and March Order parameters.

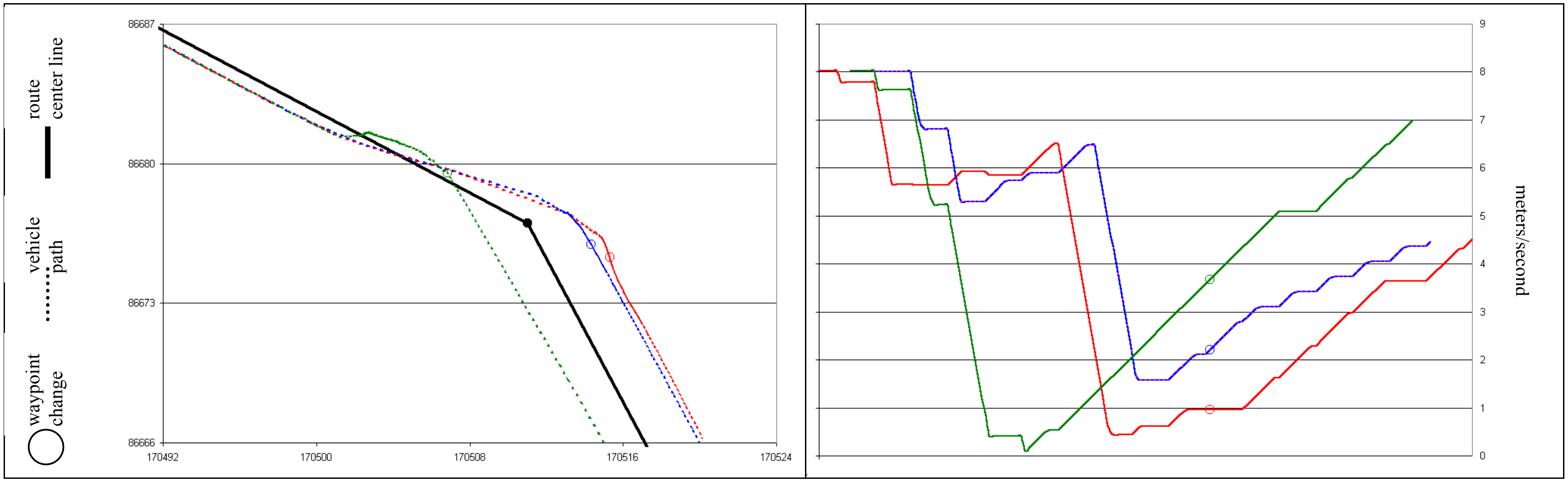


Figure 3. ModSAF's Trajectory (left) and Speed (right) Data for Runs 1, 2, and 3

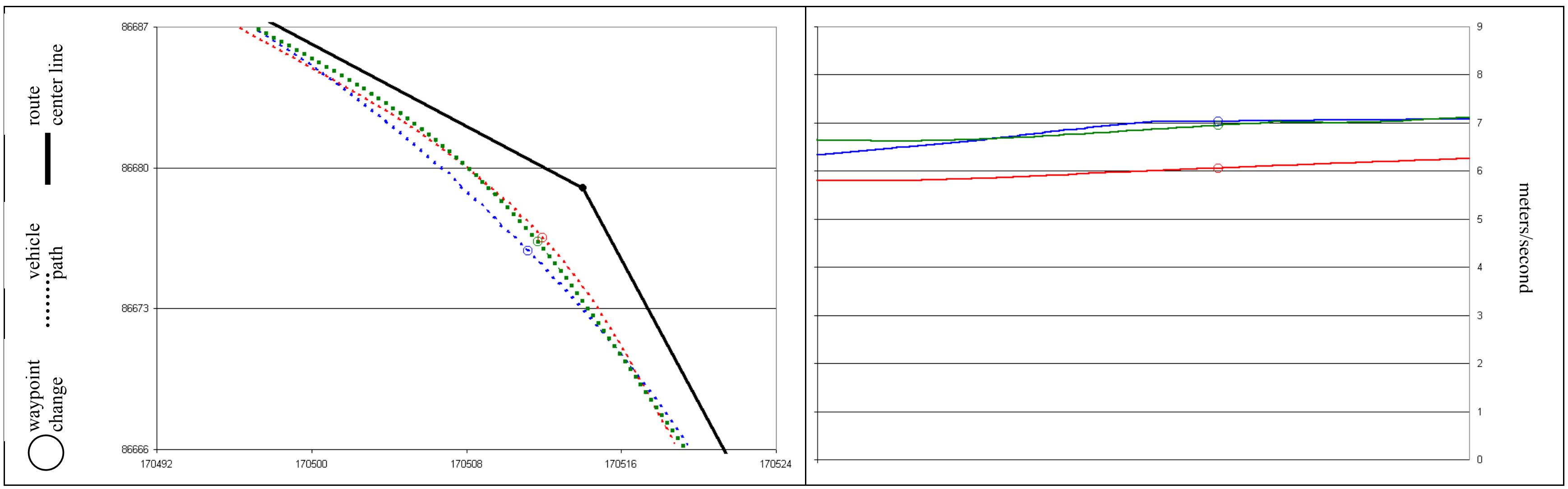


Figure 4. SME's Trajectory (left) and Speed (right) Data for Runs 1, 2, and 3

$$Rwp = D_{wp} / \max(D_{wp}) \quad (1)$$

$$Rcl = D_{cl} / M \quad (2)$$

$$Rs = S / M \quad (3)$$

$$HRab = Hab / Hxy \quad (4)$$

$$HRbc = Hbc / Hxy \quad (5)$$

where

$S = entity\ speed$

$Dwp = distance\ to\ waypoint\ at\ turn$

$M = march\ order\ speed$

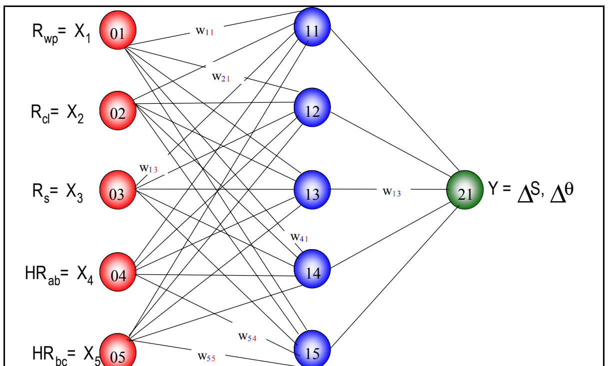
$Dcl = distance\ to\ road's\ centerline$

$Hab = direction\ of\ road\ segment\ ab$

$Hbc = direction\ of\ road\ segment\ bc$

$Hxy = entity\ heading$

Graphically, each of the networks may be visualized as shown in Figure 5. The goal of



where

- x_1, x_2, x_3, x_4, and x_5 are the predictors (k = 1 to 5)
- y is the response (i = 1)
- there are five hidden nodes (j = 1 to 5)
- there are three layers (L = 0,1,2)
- node "12" is node 2 in layer 1
- weight "13", w_{13}, connects node 3 of layer L to node 1 of layer L+1

Figure 5. 5-5-1 Feed-Forward Architecture

training these networks is to determine the set of weights that will minimize the error between the calculated output, y_l, that reflects the propagated effects of the inputs, and the training output, d_l, as shown in equation (6).

$$E^p(w) = \frac{1}{2}\sum_{i=1}^{I=1}[d_i^{(2)}(p) - y_i^{(2)}(p)]^2 \quad (6)$$

where y_l is calculated by summing the products of the inputs and weights, w, for each node in the hidden layer (7), applying a non-linear transformation function to each the nodes in that layer (8), and then propagating these effects to the output layer (9). So, to compute the output of the middle layer, y_j, the total weighted input to the j^{th} node for pattern *p* is given by:

$$net_j^{(1)}(p) = \sum_{k=1}^{K=5} x_k^{(0)}(p) w_{jk}^{(1)} \quad (7)$$
$$\text{for } j = 1 \text{ to } 5$$

and a non-linear activation function is applied to give the output of the jth node, $y_j^{(1)}(p)$:

$$y_j^{(1)}(p) = g(net_j^{(1)}(p)) = \frac{1}{1+e^{-net_j(p)}} \quad (8)$$

where $g\,()$ is the frequently used sigmoid function.

These effects are then propagated to the output layer by applying similar sets of equations (9). The net input to the i^{th} node for pattern *p* is given by:

$$y_i^{(2)}(p) = net_i^{(2)}(p) = \sum_{j=1}^{J=5} y_j^{(1)}(p) w_{ij}^{(2)} \quad (9)$$
$$\text{for } i = 1$$

and, in this instance, the output of the i^{th} node for pattern *p*, $y_l(p)$, is not transformed, and thus is the same as the input.

Once the error is computed (6), the weights are adjusted by computing the negative gradient of the error function and taking the partial derivatives of this function with respect to the weights (equations 10 and 11). This allows errors at the output layer to be propagated backward toward the input layer in proportion to the error contributions due to the weight changes at the previous layer.

$$\Delta w_{ij}^{(2)} = -\eta \left(\frac{\partial E}{\partial w_{ij}^{(2)}}\right) \quad (10)$$

$$\Delta w_{jk}^{(1)} = -\eta \left(\frac{\partial E}{\partial w_{jk}^{(1)}}\right) \quad (11)$$

where η is a user defined positive constant representing the rate of descent along the error surface.

By applying the chain rule of derivation (see Rummelhart et al, 1986 for complete derivation), these equations reduce to:

$$\Delta w_{ij}^{(2)} = \eta \delta_i^{(2)}(p) y_j^{(1)}(p)$$
$$\text{for } i=1,\ j=1 \text{ to } 5, \text{ where} \qquad (12)$$
$$\delta_i^{(2)}(p) = g'(net_i^{(2)}(p)) \cdot (d_i^{(2)}(p) - y_i^{(2)}(p))$$

$$\Delta w_{jk}^{(1)} = \eta \delta_j^{(1)}(p) x_k(p)$$
$$\text{for } j=1 \text{ to } 5,\ \ k=1 \text{ to } 5 \text{, where} \qquad (13)$$
$$\delta_j^{(1)}(p) = g'(net_j^{(1)}(p)) \sum_{i=1}^{I=1} w_{ij}^{(2)} \delta_i^{(2)}(p)$$

For the example in Figure 5 representing the 5-5-1 single output networks used in this investigation, these equations simplify to:

$$\Delta w_{1j}^{(2)} = \eta\, (d_1(p) - y_1(p)) \cdot (y_1^{(1)}(p)) \qquad (14)$$

$$\Delta w_{jk}^{(1)} = \eta\, (y_j(p)) \cdot (1 - y_j(p)) \cdot (d_1(p) - y_1(p)) w_{1j}^{(2)} x_k(p) \qquad (15)$$

Each of these weight adjustments directs the network towards a solution to the input/output mapping. That is, these weights are training the network to produce a certain output given a set of inputs. This is one of the fundamental benefits of the neural network approach. With the proper training and representation, the network will arrive at a mapping of how the responses are formed and there is no need to acquire and represent an expert's knowledge in terms of rule sets.

RESULTS

Using data represented in Figure 4, architecture represented in Table 1, and the equations presented in the previous section, two neural networks representing the change in entity speed and change in entity orientation were trained and evaluated. The application of the resulting model to the initial conditions in Run 1 generated the trajectory and speed data displayed in Figures 6a and 6b, respectively. It is clear, even from a simple visual comparison that the neural network trained with SME generated data more accurately represents true SME behavior than the ModSAF entity movement model does. However, the improvement in fidelity comes at a small cost, as presented in Henninger et al (1999) which reports that the neural network-based model executes in an average of $2.925\text{x}10^{-4}$ seconds as compared to the standard ModSAF near term movement model which executes in an average of $1.7650\text{x}10^{-4}$ seconds. That is, the neural network program required almost two-thirds more processing time than the standard ModSAF near-term movement model required. This estimate, however, is a conservative figure favoring ModSAF, as it includes all computations (i.e., data pre-processing and model execution) for the neural network based movement behavior but does not include a portion of the pre-processing operations performed for the execution of the ModSAF movement model.

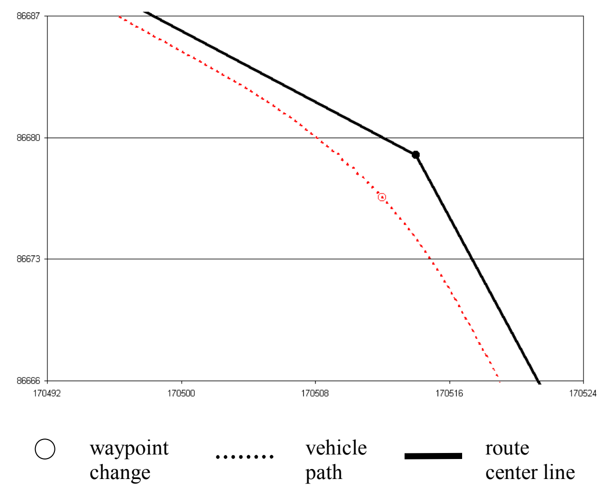


Figure 6a. Neural Network Based Movement Model's Trajectory

Figure 6b. Neural Network Based Movement Model's Speed Distribution (m/s)

CONCLUSIONS

The results of this initial study suggest that modeling human performance data with neural networks can improve the fidelity of SAF models.
However, since the domain of required SAF behavior is so broad and the scope of the

investigation was significantly limited, further work is required to evaluate the potential of this modeling paradigm to the complete domain of SAF behaviors.

ACKNOWLEDGEMENTS

This work was sponsored by the U.S. Army Simulation, Training, and Instrumentation Command as part of the Inter-Vehicle Embedded Simulation and Technology (INVEST) Science and Technology Objective (STO) contract N61339-98-K-0001. The support of Hubert Bahr, Claude Abate, and others on the INVEST program is gratefully acknowledged.

BIBLIOGRAPHY

Boose, J. H., and Bradshaw, J. M. (1987). Expertise Transfer and Complex Problems: Using AQUINAS as a Knowledge Acquisition Workbench for Knowledge-Based Systems. International Journal of Man-Machine Studies, vol. 26, no. 1, pp. 3-28.

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

Deutsch, S. (1993). Notes Taken on the Quest for Modeling Skilled Human Behavior. In Proceedings of the Third Conference on Computer Generated Forces and Behavior Representation. Orlando, FL., March 17-19, pp. 359-365.

Henninger, A., and Gonzalez, A. (1999). Neural Network Based SAF Systems: Experimental Results. Proceedings of the 19th Interservice/Industry Training Systems Conference, Orlando, FL. Nov 30-Dec 1, 1999, pp 1411-19.

Hilgard, E.R., Atkinson, R.L., and Atkinson, R.C. (1979). Introduction to Psychology, Seventh Edition. Harcourt Brace Jovanovich, Inc. New York, NY.

Lugar, G.F., and Stubblefield, W. A. (1993). Artificial Intelligence Structures and Strategies for Complex Problem Solving. The Benjamin/Cummings Publishing Company, Inc. Redwood City, CA.

Nechyba, M.C. and Xu, Y. (1997). Learning and Transfer of Human Real-Time Control Strategies. Journal of Advanced Computational Intelligence, vol.1, pp. 137-154.

Ourston, D., Blanchard, D., Chandler, E., and Loh, E. (1995). From CIS to Software, In Proceedings of the Fifth Conference on Computer Generated Forces and Behavior Representation. Orlando, FL., May, 1995, pp. 275-285.

Pew, R.W., and Mavor, A.S. (1998). “Modeling Human and Organizational Behavior: Applications to Military Simulations”. National Academy Press, Washington D.C., 1998.

Pomerlau, D., Thorpe, C., Longer, D., Rosenblatt, J.K., and Sukthankar, R. (1994). AVCS Research at Carnegie Mellon University. Proceedings Of Intelligent Vehicle Highway Systems America 1994 Annual Meeting, p. 257-262.

Rummelhart, D., and McClelland, J. (1986). Parallel Distributed Processing. MIT Press, Cambridge, MA.

Shaw, M.L., and Gaines, B.R. (1987). KITTEN: Knowledge Initiation and Transfer Tools for Experts and Novices. International Journal of Man-Machine Studies, vol. 27, no. 3, pp. 251-280.

Smith, J. (1994). Near-term Movement Control in ModSAF. In Proceedings of the Fourth Conference on Computer Generated Forces and Behavior Representation. Orlando, FL., March, 1994, pp. 249-260.

Smith, S.H., and Petty, M.D. (1992). Controlling Autonomous Behavior in Real-Time Simulation. In Proceedings of the Second Conference on Computer Generated Forces and Behavior Representation. Orlando, FL., March, 1992, pp. 27-40.

Velt, C.T. (1993). Developing Validated Behavioral Representations for Computer Simualtion. In Proceedings of the Third Conference on Computer Generated Forces and Behavior Representation. Orlando, FL., March 17-19, pp. 417-428.

This document is an author-formatted work. The definitive version for citation appears as:

A. E. Henninger, A. J. Gonzalez, W. Gerber, M. Georgiopoulos, and R. F. DeMara, "On the Fidelity of SAFs: Performance Data" in *Proceedings of the 2000 Interservice/Industry Training, Simulation and Education Conference (I/ITSEC-2000)*, pp. 147 – 154, Orlando, Florida, U.S.A., November 27 – 30, 2000.