AN EXPERIMENTAL APPLICATION OF A TRAIT-BASED PERSONALITY MODEL TO THE SIMULATION OF MILITARY DECISION-MAKING

Frederic (Rick) D. McKENZIE, Mikel D. PETTY
and Jean CATANZARO

Introduction

Due to the uncertainties inherent in military operations and the variations in human personalities, military command and control behavior rarely conforms strictly to doctrine. (Of course, the degree of conformity to doctrine may vary.) In a military simulation with automated commanders, models of command behavior and decision-making that follow doctrine precisely and exhibit no variations are therefore unrealistic. Automated commanders that exhibit doctrinal behavior are essential for training, especially at the introductory levels, but are not sufficient for the full range of purposes the simulation may be applied to, such as advanced training and mission rehearsal. Simulation users may seek an automated commander that realistically models the effects of the fog of war and the difficulty of making doctrinal decisions under stressful conditions. Such realism in simulation could better prepare trainees for expected encounters on the battlefield.

To achieve this end, the realistic modeling of human behavior has become a pervasive topic in the modeling and simulation community. How does one go about representing realistic human behavior? Some psychologists have looked to personality traits to characterize behavior. Different humans behave differently in the same situation, depending on their personalities. Commander personality has a significant effect on command decision-making, so modeling personality and its effects on command could improve simulation realism. For some applications, realistic command simulation may require a representation of personality.

We have implemented and tested a simulation that incorporates a trait-based model of human personality into the decision-making of a simulated commander. The model was implemented and used in an experiment intended to test its utility in producing more realistic human decision-making in a way that could be validated by personality and performance measurements of real human commanders. The simulation models a decision-making scenario where a commander must dispatch and route trucks, assumed to be loaded with food and medicine, to a refugee camp with urgent requirements for the supplies. Land mines have been placed, in numbers that vary by road segment, in the road network between the supply depot and the refugee camp. The mines will disable a portion of the trucks that attempt to traverse each segment, based on mine density. The simulated commander knows the roads are mined but does not know the number of mines on the various road segments in the network and so must make dispatch and routing decisions in the absence of complete information and with the expectation of having trucks disabled. In this scenario, time does not permit the mines to be searched for or removed; the only method the commander has to learn of the relative danger of different road segments is to route trucks along them. The simulation forces the commander to make decisions under the stress of conflicting goals; the trucks must reach the refugees quickly, but the road network must be explored carefully to determine the least dangerous routes.

The following sections of this paper cover these topics. The personality traits that are the psychological basis of the model of human personality used in the automated commander are briefly reviewed. The experimental scenario and simulation of it is described. Details of the design of the automated commander, including the integration of the trait-based personality model, are given. The results of experimental testing of the model are reported. Finally, an alternative set of traits for the human personality model, suggested by the results, is provided.

Personality Traits and Decision-Making

In the context of military command it is critical to assign the right person to the right job and adequately train that person to competence. As military trainers have found, training and repetition can train out certain undesirable characteristics of a person's performance. For example, a person with sufficient training in the situation he/she faces may exhibit reduced fear and panic response, have better reaction time, and make fewer careless mistakes. Unfortunately, when in a stressful or unexpected situation, especially one, which a commander's training has not prepared him/her for, the commander's behavior and decision-making performance may revert to his/her innate psychological characteristics. In such circumstances a commander's individual personality is most visible in his/her behavior. For simulations purposes a commander model that accounts for personality would be useful for producing realistic decision-making behavior from psychological profiles of human decision-makers.

Models of personality have been considered in the past, but progress has been dependent on the existence of appropriate tools for evaluating personality traits in the context of military decision-making. The trait-based personality model used in this research is based on an extensive investigation of the battlefield behavior of 20th-century infantry[1] and has been previously suggested for applying a trait-based model in simulation.[2] The model asserts eight distinct personality traits that impact decision-making. Those personality traits are listed and defined as follows:

1. *Stability.* This is a generic trait that expresses a person's overall emotional stability, rather than a particular emotion. It serves as the "governor" of emotional expression, particularly extreme emotions such as panic.
2. *Anxiety.* This trait expresses a person's inherent fearfulness.
3. *Anger.* Broadly expressing the emotion of anger, this trait also accounts for a person's inherent aggressiveness and resentment.
4. *Humor.* Representing more than a simple sense of humor, this trait also expresses a person's capacity for emotional "bounce-back" and the ability to recover from sudden shocks, losses, and other negative impacts on morale.
5. *Acquiescence.* This trait represents a person's willingness to follow commands, orders, and other leaders.
6. *Independence.* This trait expresses the ability of a person to make decisions independently, without leadership.
7. *Charisma.* A composite trait that collectively expresses aspects of personality that others tend to find attractive.
8. *Knowledge.* This trait replaces the ambiguous term "intelligence" which has a particular meaning in military terms. It refers to military knowledge, ranging from weapons and equipment to tactics.

Whereas personality *traits* are relatively stable characteristics of a person, his/her decision-making can also be affected by the more transient condition of psychological *state*. In contrast to traits, states are dependent on the situation and relatively temporary. For example, a person may have a consistent predisposition towards anger (a trait), but may have that angry disposition overlaid or temporarily displaced by tranquility (a state) resulting from an event such as a mission success. In other words, a person's trait-based tendencies can be temporarily counteracted by event-driven states.

The personality model synthesizes the basic psychological notions of personality traits and states into composite factors that influence military command decision-making; these factors include situational stress (e.g., the friend-to-foe ratio) and morale (based on a combination of personality traits, stress, and support).

Experimental Scenario and Simulation

An experimental scenario was designed to exercise command decision-making. A simulation was implemented to specifically support that scenario.

Experimental Scenario

A hypothetical United Nations (UN) peacekeeping and humanitarian assistance force has received an extremely urgent request to deliver medical supplies and food to a refugee camp in the Balkans. The supplies are needed within the next 12 hours to avoid many refugee deaths. Extremely bad weather prevents air transport of the supplies. The UN force has assembled a group of trucks at the closest supply depot and loaded them with the needed supplies. The trucks must travel to the refugee camp as quickly as possible.

Unfortunately, what would otherwise be a simple route-planning problem is complicated by the fact that hostile militia forces have placed land mines throughout the road network between the supply depot and the refugee camp. The terrain is rugged enough to restrict truck travel to the roads. The exact locations and density of the mines are unknown to the UN commander, and there is not sufficient time to perform mine search and removal. The mines used by the militia are of a type that if hit by a truck will disable the truck but will not kill the UN drivers. The trucks are all equipped with radios and global positioning system receivers. The UN commander decides to dispatch and route the trucks individually to the refugee camp, controlling their movements centrally by radio from the command post, and to adjust later truck's routes based on knowledge of the mine locations learned from the preceding trucks.

Experimental Simulation

In the simulation of this scenario, the terrain is represented as an undirected graph, with vertices corresponding to road intersections and edges to the roads connecting the intersections. Trucks are located at vertices. Trucks move from vertex to vertex along edges. The supply depot and refugee camp are both vertices. Figure 1 is an example of a terrain graph.

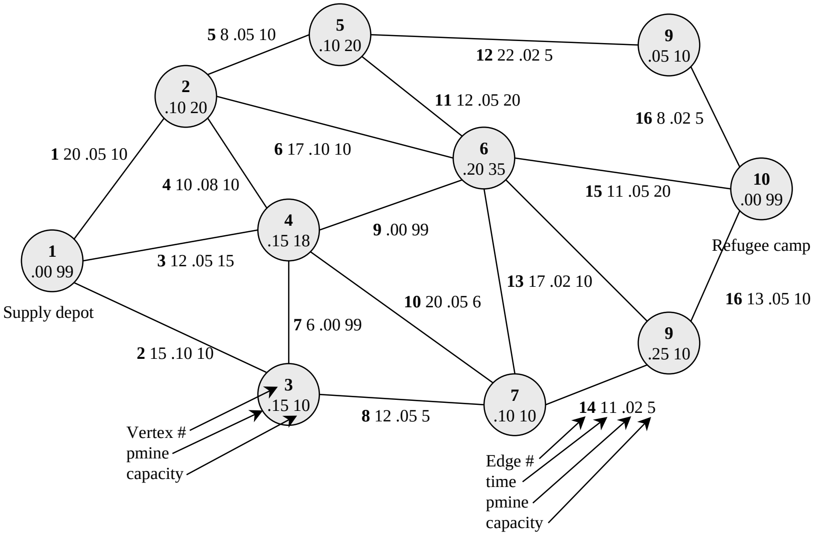


Figure 1: Example Terrain Graph.

Each edge *e* in the terrain graph has these attributes:

1. *e.time* Time required by a truck to traverse edge *e*. These values are determined by the geographical distance between the connected vertices, the average speed of truck movement, and a random increase to reflect specific road conditions.
2. *e.pmine* Probability of a truck being disabled by a mine when traversing edge *e*.
3. *e.capacity* Maximum number of trucks that may simultaneously be traversing edge *e*. This capacity limit includes trucks that may become disabled on *e*.

Each vertex *v* has these attributes:

1. *v.pmine* Probability of a truck being disabled when moving to vertex *v*.
2. *v.capacity* Maximum number of trucks that may be located at vertex *v*. This capacity limit includes trucks that may become disabled at *v*.

The simulation implementation employs the discrete-event simulation paradigm. The movement of a truck from one vertex to another is a simulation event, as is the disabling of a truck on an edge or a vertex. When a truck located at vertex v_i at current time *t* moves from vertex v_i to v_j, along edge e_k, it arrives at v_j at time $t + e_k.time$. At time $t + (e_k.time / 2)$, $e_k.pmine$ is used to determine if the truck is disabled while moving along the edge. If it is disabled, the UN commander is notified that the truck was disabled. Otherwise, the truck arrives at vertex v_j at time $t + e_k.time$. At that time a determination is made using $v_j.pmine$ if the truck is disabled at the vertex and the UN commander is notified of the truck's status, either arrived safely or disabled at the vertex. Because the effects of a large number of mines are being modeled probabilistically, mines are not removed when a truck is disabled, i.e., the *e.pmine* and *v.pmine* values are not changed at such events. No more trucks may be simultaneously located at a vertex v_i than its capacity $v_i.capacity$; similarity, no more trucks may be traversing an edge e_k than its capacity $e_k.capacity$.

The automated UN commander decides when to dispatch each truck from the supply depot. It also decides when each truck reaches a vertex, which connected vertex the truck will next move to. At the start of a trial, the UN commander has no knowledge of the mine distribution (i.e., of the *v.pmine* and *e.pmine* values). Over time, the UN commander accumulates an estimate of the *pmine* values based on the experiences of the trucks as they move through the graph. The dispatch and routing decisions are made using a decision model, described in the next section that may consider the estimated *pmine* values.

In addition to the simulation time taken by the trucks' movements, the automated UN commander's decisions require time. The amount of time per decision is a parameter of the commander model. A trial ends when all trucks have either reached the refugee camp vertex or been disabled.

Design of the Automated Commander

The implementation of automated commander's decision model essentially consists of two parts. The first part of the decision model is a set of graph search algorithms that find routes in the terrain graph; they generate alternatives for the commander's basic routing decisions. The routing algorithms differ in terms of whether they seek to minimize time, minimize risk of disablement, or minimize some combination of those. The second part of the decision model is the trait-based personality model. It influences the routing decisions in that it is used to select among the alternatives generated by the routing algorithms and may also cause a delayed or degraded decision.

Search Algorithms

The cost of a route in the terrain graph is a function of the time to traverse it and the probability of being disabled on the route. The set of graph search routing algorithms used in the automated commanders consider one, or the other, or both of those aspects of cost. The routing algorithms are:

1. Minimum Time Cost (MTC)
2. Least Damage (LD)
3. Least Percent Damage (LPD)
4. Minimum Time Cost and Least Damage (MTCLD)
5. Minimum Time Cost and Least Percent Damage (MTCLPD)

An A* heuristic search algorithm[3,4] is used to find the minimum time route through the road network. The minimum time route may vary over simulation time because edges in the network may become unusable when trucks are disabled and block further truck traversal on particular edges. The algorithm is executed repeatedly to update the minimum time route. This route is used as a standard to measure the performance of the search procedures used. The true risk of a route may be calculated using the true probabilities of disablement (the *e.pmine* and *v.pmine* values), rather than the estimates of those values derived from experience as a percentage of trucks disabled on the edge or vertex. The true risk of a route may also serve as a performance standard.

The other four routing algorithms consider not only minimum time but also heuristics dealing with the estimated probability of disablement on a route, based on the quantity or percentage of trucks that have been disabled at each edge or vertex on the route. These values will change as the scenario is executed and more trucks are disabled.

Some details of the routing algorithms are now given. They use these parameters:

$\#D$, $\%D$, pD = number, percentage, and probability of trucks disabled, in total

$\#D_e$, $\%D_e$, pD_e = number, percentage, and probability of trucks disabled, on edge *e*

$\#D_v$, $\%D_v$, pD_v = number, percentage, and probability of trucks disabled, at vertex *v*

Minimum Time Cost (*MTC*). As mentioned earlier, the MTC algorithm uses an A* graph search procedure to find the path of least cost (time), which is approximated by an evaluation function $e(v)$ that is calculated for each vertex v along the path. The evaluation function sums the actual time $c(v)$ required to reach v and the estimated cost $h(v)$ of getting from v to the goal vertex. The MTC algorithm uses the time required to traverse each edge *e.time* throughout the network to calculate these costs. The evaluation and cost functions for MTC are defined as follows:

$$e(v)_{MTC} = c(v)_{MTC} + h(v)_{MTC}$$

$$c(v)_{MTC} = \sum_{i \in R} e_i.time$$

$$h(v)_{MTC} = d(v) / K$$

where R is the route the truck has taken so far to vertex v, K is the average truck speed, and $d(v)$ = Euclidean distance $\sqrt{a^2 + b^2}$ assuming a and b are the horizontal and vertical distances from v to the goal vertex.

Least Damage (LD). The LD algorithm is focused on reducing risk, not time, on its routes; the LD cost function considers only the number of trucks previously disabled along a possible route segment (edge and terminating vertex). Movement is directed toward the segment with the least number of previously disabled trucks.

$$c(v)_{LD} = \#D_E + \#D_N$$

Least Percent Damage (LPD). Similar to the LD algorithm in its focus on risk, the LPD algorithm's cost function considers the percentage, rather than the number of trucks that have previously been disabled when attempting to traverse a route segment.

$$c(v)_{LPD} = \%D_E + \%D_N$$

Minimum Time Cost and Least Damage (*MTCLD*). In its cost function, the MTCLD algorithm considers both the time and number of trucks disabled for a particular route segment.

$$c(v)_{MTCLD} = \frac{\#D_E + \#D_N}{\sum_{i=0}^{AP} \left(\#D_E(i) + \#D_N(i)\right)} + \frac{e.time}{\sum_{i=0}^{AP} e.time}$$

$$h(v)_{MTCLD} = \frac{d(v)}{\sum_{i=0}^{AP} d(i)}$$

where AP denotes all paths.

Minimum Time Cost and Least Percentage Damage (*MTCLPD*). In its cost function, the MTCLPD algorithm considers both the time and percentage of trucks disabled for a particular route segment.

$$c(v)_{MTCLPD} = \frac{\%D_E + \%D_N}{\sum_{i=0}^{AP} \left(\%D_E(i) + \%D_N(i)\right)} + \frac{e.time}{\sum_{i=0}^{AP} e.time}$$

$$h(v)_{MTCLD} = \frac{d(v)}{\sum_{i=0}^{AP} d(i)}$$

Trait-Based Personality Model

Within the framework of the five routing algorithms the challenge is to define what constitutes normal and sub-optimal decision behavior and establish a link between a commander’s personality and the decisions he/she makes. The automated commander’s decision model is based on the assumption that a human commander would make routing decisions that closely approximate (perhaps in sub-optimal form) one of the routing algorithms previously described. Which algorithm would the commander use, and whether or not the decision made would be sub-optimal, depends on the commander’s personality traits and the current state of the simulation.

As previously described, the commander’s personality is specified with a set of eight personality traits. In general, personality traits determine the predisposition of people to exhibit a particular behavior under varying situational conditions. Such trait and state effects on decisions are modeled in this research as decision delay and decision optimality. For example, stress is a situational condition that may affect the decision-making performance of a military commander. The personality model causes commanders with certain personality traits to make sub-optimal decisions under high stress conditions. Sub-optimal decisions are obtained from evaluating and ranking the five search algorithms against a particular scenario. Ranking may also be chosen based on qualitative criteria. Delayed decisions are obtained by randomly increasing the decision time according to parameters that are part of the commander's personality profile (the term personality profile refers to the collection of the eight trait values for a particular commander).

The decision model design uses the commander's personality traits and current simulation state to calculate the commander's stress and morale and ultimately his/her accuracy and effectiveness. Based on those results, one of the available decision actions calculated by the five decision algorithms is selected. The decision selection also includes the possibility of a delayed decision (long decision time).

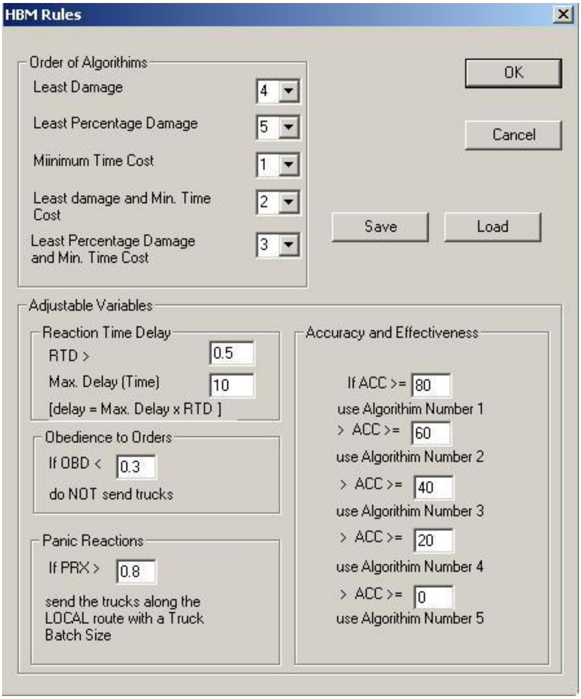
A user interface in the simulation, shown in Figure 2, is used to enter the parameters that connect the commander's personality to the process of selecting the decision of one of the routing algorithms. Via this interface the user enters the effectiveness ordering of the routing algorithms, the commander's reaction time, the commander's obedience and panic parameters, and the accuracy and effectiveness levels associated with the routing algorithms. The commander's personality traits are used to compute his/her accuracy and effectiveness in a given situation; then that value is used, based on the parameters entered in the last portion of this interface, to select one of the routing algorithms' decisions. Leaders with personalities that make them more effective in the current situation will select the decisions of the better algorithms.

Simulation Experiments

A series of simulation experiments were conducted to test the integration of the trait-based personality model into the automated commander and its effectiveness at producing realistic decision-making behavior.

Simulation Environment and Scenario Generation

The simulation's user interface allows the user to create and edit terrain graphs. Based on user inputs, vertices and edges in a terrain graph may be randomly generated and/or manually edited. Similarly, edge and vertex attributes, such as *e.time* and *v.pmine*, can be generated by the simulation and/or manually edited. Other scenario information, such as number of trucks, is also input via the user interface. Once generated, terrain graphs and scenario data can be saved.



Order of Algorithims

Least Damage 4

Least Percentage Damage 5

Miinimum Time Cost 1

Least damage and Min. Time Cost 2

Least Percentage Damage and Min. Time Cost 3

Adjustable Variables

Reaction Time Delay

RTD > 0.5

Max. Delay (Time) 10

[delay = Max. Delay x RTD]

Obedience to Orders

If OBD < 0.3

do NOT send trucks

Panic Reactions

If PRX > 0.8

send the trucks along the LOCAL route with a Truck Batch Size

Accuracy and Effectiveness

If ACC >= 80

use Algorithim Number 1

> ACC >= 60

use Algorithim Number 2

> ACC >= 40

use Algorithim Number 3

> ACC >= 20

use Algorithim Number 4

> ACC >= 0

use Algorithim Number 5

Figure 2: User Interface for Connecting Commander Personality to Decision Algorithms.

Figure 3 shows an example scenario; the example in the figure is smaller than the road networks used for the actual experiments. In the figure the circles represent the road intersections (vertices) interconnected with lines that represent the roads (edges). The color green (G) and the light lines denote a road or intersection that has been traversed without incident, red (R) indicates that at least one truck has been disabled on that road or intersection, and blue (B) means that the road or intersection has not yet been traversed by any trucks. In the figure the leftmost intersection (a white (W) node) is the supply depot and the rightmost intersection (a green (G) circle) is the refugee camp. The numbers labeling each edge and vertex indicate the number of trucks traversed and disabled.

In addition to the automated commander, the simulation has interactive capability whereby a human operator can make the trucks' routing decisions. This capability provides a mechanism to compare the automated commander's performance with that of human commanders.

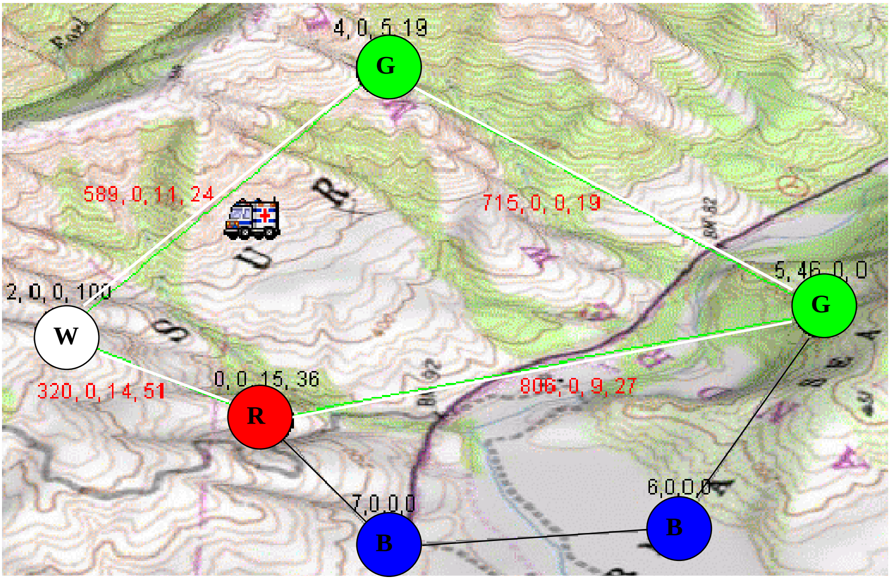


Figure 3: Example Road Network and Scenario.

Performance Evaluation Metric

The long-term goal of this research is to contribute to generating realistic decision-making behavior by automated commanders in simulations. To assess progress towards that goal, the effectiveness and realism of the decisions made by the automated commander must be quantitatively measured. The objective is not to obtain some mathematically optimum performance level for an automated commander, but rather to generate similar decision-making outcomes as would be found in human commanders.

Under the performance metric defined for the scenario, the commander seeks to maximize number of trucks arriving at the refugee camp within a given time limit and minimization of both the number and the lateness of trucks arriving after the time limit. The performance metric is defined as follows:

$$\max\left\{\sum_{i=1}^{N}\left[c_i d_i + c_i(1-d_i)(s/a_i)\right]/N\right\}$$

where:

N = The number of trucks.
s = A constant; the time limit given for trucks to arrive at the refugee camp.
d_i = 1 if truck i arrives within the critical time limit, 0 if it is late.
c_i = 1 if truck i arrives at the refugee camp vertex, 0 if it is disabled.
a_i = The arrival time of truck i at the refugee camp vertex.

Note that for each truck the quantity c_id_i will be 0 or 1, the quantity $c_i(1 - d_i)(s / a_i)$ will be in the range 0 to 1, and only one will be non-zero. The performance of a commander will be the sum of N such quantities, divided by N, which will therefore be in the range 0 to 1 (inclusive). This normalized measure of performance allows the commanders' performance to be compared for different numbers of trucks and different terrains.

Experimental Results

A series of experiment trials were performed using typical road network topology generated over a given terrain. In preparation for the experiments the five search algorithms were executed on the experimental networks in order to determine their effectiveness ranking on those networks.

Figures 4, 5, and 6 compare the performance of the five routing algorithms without personality influence. In the figures, the horizontal axis shows time limit and the vertical axis shows performance metric values. All three figures illustrate that the more time a commander has the better he/she will perform. Figure 4 shows the performance of the five algorithms over seven trials with a common road network and increasing time limits. For these trials the road network had an equal probability of being disabled by mines at every intersection (vertex) and road (edge) in the graph. In such a road network, where no route segment is lower risk than any other, the MTC algorithm will outperform the other algorithms; Figure 4 confirms that result. On the other hand, if the probabilities of being disabled vary across the intersections and roads, the relative rankings of the five algorithms may be different. Figure 5 shows a series of ten trials, again with a common road network and increasing time limits. In the road network used for these trials the MTC had the worst performance and the LD algorithm was the best in terms of the performance metric. Figure 5 illustrates that the time delay associated with taking alternate routes can be justified if a sufficient reduction in the number of disabled trucks results from the detours.

Figure 6 illustrates that when the shortest path is only slightly riskier than a longer path there is a balance between taking the shortest path (minimizing time) and a longer patch (minimizing risk). If the time limit is large (toward the right side of the figure) detouring off the shortest path will yield better results, but when the time limit is small (toward the left side of figure) the MTC algorithm performs best.

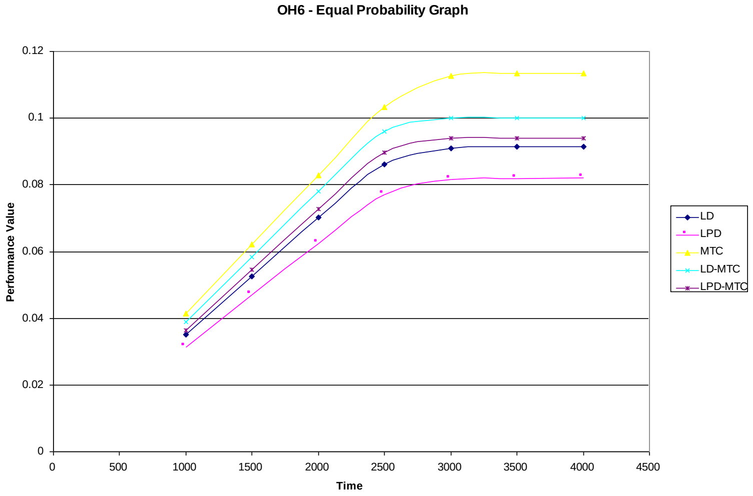


Figure 4: Results for a Road Network with Equal Probabilities of Disablement.

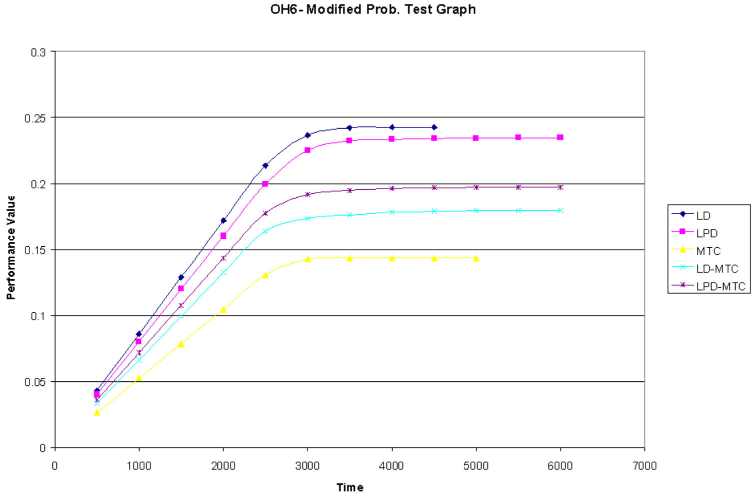


Figure 5: Results for a Road Network with Widely Varying Probabilities of Disablement.

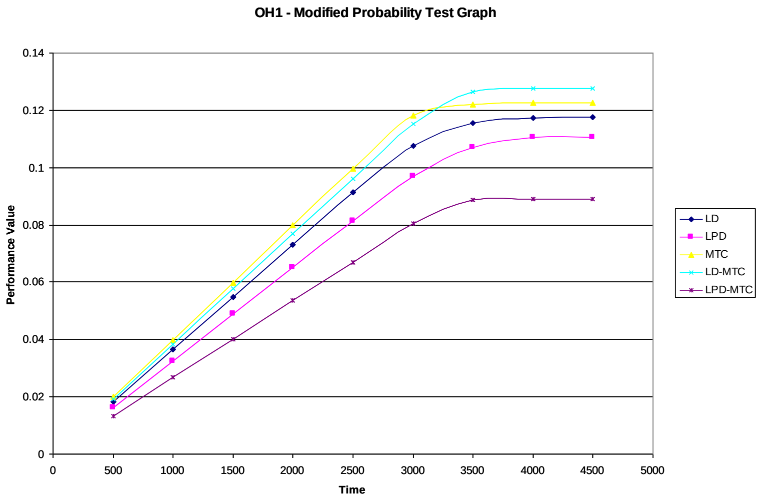
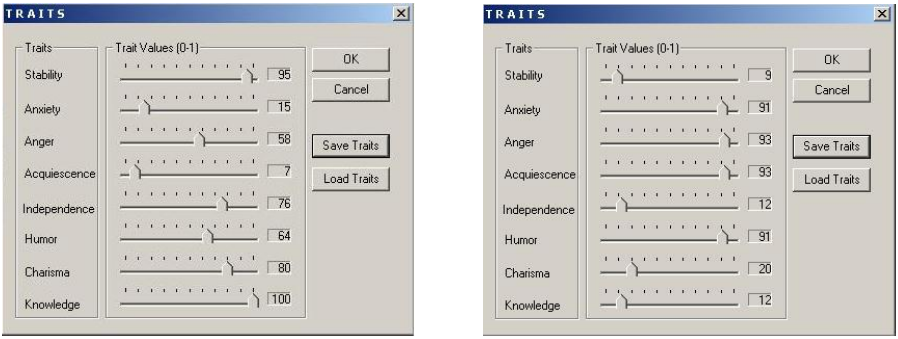


Figure 6: Results for a Road Network with Slight Varying Probabilities of Disablement.

After verifying the performance of the five algorithms against various road networks, a set of several widely varying personalities were encoded into the automated commander. These personality profiles did not correspond to specific persons; there were produced by the experimenters to evaluate the model's ability to increase realism and were intended to be representative of typical commander profiles. The performance results of the representative personality commanders were then analyzed for system sensitivity and realism.

Figure 7 illustrates the personality profiles used for two of the representative commanders (as well as the user interface in the simulation for entering commander personalities). Figure 7(a) shows a generally "good" commander, with personality traits typical of low anxiety and high knowledge. In contrast, figure 7(b) shows a generally "bad" commander, with high anxiety and low knowledge.

Over multiple trials the "good" commander's average performance value was 0.15 and the "bad" commander's average performance value was 0.08. Even though the "good" commander was simply choosing among decisions made by the five routing algorithms, that commander's average performance was better than any one of the five algorithms because his/her personality allowed him/her to choose the best decision for a situation. For the opposite reason the "bad" commander's average performance was worse than any one of the five algorithms. Figure 8 compares the good and bad commanders' performance values of 0.15 and 0.08 to the performance of the five routing algorithms without any personality influence. The "good" commander performs significantly better than any of the five algorithms, whereas the "bad" commander performs worse than any of the five algorithms for time limits ≥ 3000.



TRAITS

Traits	Trait Values (0-1)
Stability	95
Anxiety	15
Anger	58
Acquiescence	7
Independence	76
Humor	64
Charisma	80
Knowledge	100

(a) "Good" commander

TRAITS

Traits	Trait Values (0-1)
Stability	9
Anxiety	91
Anger	93
Acquiescence	93
Independence	12
Humor	91
Charisma	20
Knowledge	12

(b) "Bad" commander

Figure 7: Personality Profiles for Representative Good and Bad Commanders.

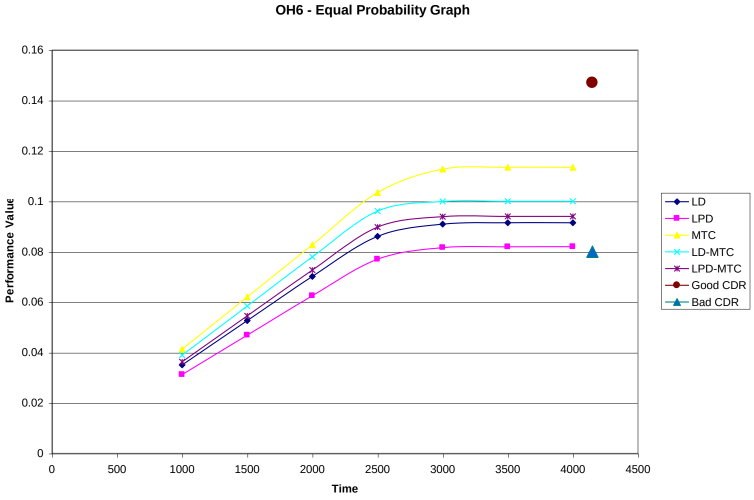


Figure 8: Average "Good" and "Bad" Commander Performance.

Comments on the Results

Though the representative "good" and "bad" commanders showed good and bad performance as expected, some of the results obtained using the other representative personality profiles were more mixed. For example, it was possible to define a "panicky" commander that would outperform a "good" commander in some situations, an unexpected result. This could be due simply to randomness in the simulation, but it is also possible that the representative personality profiles were flawed or the method used to connect personality traits to decision-making behavior needs improvement.

The next step is to encode personality traits of human commanders obtained via personality tests and use those traits in the automated commander. Eventually, it is hoped that the performance of the automated commander and the real commander will be statistically equivalent. In order to achieve these results, two issues must be addressed. First, a more objective means of providing personality profiles is needed. Unfortunately, there are no personality tests that will provide values for the model's eight traits directly. A reliable means of determining the values of a commander's personality traits is needed. Second, the additive linear relationships used to describe a commander's reaction based on personality are imperfect at best. A learning algorithm using non-linear methods to determine likely patterns of behavior may be needed.

Conclusions

The experiments showed that a trait-based personality model could be integrated into an automated command and used to influence the decision-making of that commander. Different personality profiles were seen to produce different performance in the experimental scenario.

The experiments suggest that using a trait-based personality model of a commander could improve the decision-making realism of the commander. It also seems that, if reliable personality assessment tools can be developed, the personalities of human commanders can be used within an automated commander. The performance of an automated commander could then be compared to the human counterpart as a means of validation. Looking farther ahead, a personality model may also be applied to the task of predicting how a particular military commander might react in a situation and how to improve that commander's performance.

Acknowledgement

This work was sponsored by an internal research and development grant of the Virginia Modeling, Analysis and Simulation Center.

Notes:

[1] S. M. Silver, *Unpublished communication*.

[2] Sheila B. Banks and Martin R. Stytz, "An Approach to Enhance Human Behavior Modeling for Computer-Generated Actors," in *Proceedings of the Fourth International Simulation Technology and Training Conference (SimTecT '99)* (Melbourne, Australia, March 29-April 1, 1999), 199-204.

[3] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Transactions on Systems Science and Cybernetics* 4, 2 (July 1968): 100-107.

[4] Patrick H. Winston, *Artificial Intelligence*, Third Edition (Reading MA: Addison-Wesley, 1992).

FREDERIC (RICK) D. McKENZIE is an Assistant Professor of Electrical and Computer Engineering at Old Dominion University. Prior to joining ODU, he held a senior scientist position at Science Applications International Corporation (SAIC), serving as Principal Investigator for several projects. At SAIC he was a Team Lead on a simulation system that encompassed the training requirements of all military services and joint operations. He has several years of research and development experience in the software and artificial intelligence fields, including object oriented design in C++, LISP, and knowledge-based systems. Both his M.S. and Ph.D. work have been in artificial intelligence, focusing on knowledge representation and model-based diagnostic reasoning. *Address for correspondence:* Department of Electrical and Computer Engineering, Old Dominion University, 231C Kaufman Hall, Norfolk VA 23529; *E-mail:* fmckenzi@ece.odu.edu; *Tel.*: 757-683-5590.

MIKEL D. PETTY is Chief Scientist of the Virginia Modeling, Analysis & Simulation Center at Old Dominion University. He received a Ph.D. from the University of Central Florida in 1997, an M.S. in 1988, and a B.S. in 1980, all in Computer Science. Dr. Petty has worked in modeling and simulation since 1990 and has interests in simulation interoperability and composability, computer generated forces, and applications of theory to simulation. He was a member of a U. S. National Research Council committee on modeling and simulation and is currently an Associate Editor of the journal *SIMULATION: Transactions of the Society for Modeling and Simulation International*. *Address for correspondence:* Virginia Modeling, Analysis & Simulation Center, Old Dominion University, 7000 College Drive, Suffolk VA 23435; *E-mail:* mpetty@vmasc.odu.edu; *Telephone*: 757-686-6210.

JEAN CATANZARO is a Human Factors Engineer assisting in the design of virtual environments for military training. She has a M.S. in Industrial/Organizational Psychology and is completing her Ph.D. at Old Dominion University in Human Factors Psychology. *Address for correspondence:* Department of Psychology, Old Dominion University, Norfolk VA 23508; *E-mail:* jcatanza@odu.edu.