Enhanced Military Decision Modeling Using a MultiAgent System Approach

John A. Sokolowski, Ph.D.
Virginia Modeling, Analysis & Simulation Center
Old Dominion University
Norfolk, VA 23529
757-686-6215
jsokolow@odu.edu

Keywords:
Recognition-Primed Decision Making, MultiAgent System Simulation, Decision Modeling

ABSTRACT: *The U.S. military uses modeling and simulation as a tool to help meet its warfighting needs. A key element within military simulations is the ability to accurately represent human behavior. This is especially true in a simulation's ability to emulate realistic military decisions. However, current decision models fail to provide the variability and flexibility that human decision makers exhibit. Further, most decision models are focused on tactical decisions and ignore the decision process of senior military commanders at the operational level of warfare. In an effort to develop a better decision model that would mimic the decision process of a senior military commander, this research sought to identify an underlying cognitive process and computational techniques that could adequately implement it. Recognition-Primed Decision Making (RPD) was identified as one such model that characterized this process. Multiagent system simulation was identified as a computational system that could mimic the cognitive process identified by RPD. The result was a model of RPD called RPDAgent. Using an operational military decision scenario to test model validity, decisions produced by RPDAgent were compared against decisions made by military officers. It was found that RPDAgent produced decisions that were equivalent to its human counterparts. RPDAgent's decisions were not optimum decisions, but decisions that reflected the variability inherent in those made by humans in an operational military environment.*

1. Introduction

The U.S. Military uses modeling and simulation as a tool to help meet its warfighting needs. A key element within military simulations is their ability to accurately represent human behavior including emulating realistic military decisions. However, existing military decision models fail to provide the variability and flexibility that human decision makers exhibit [1]. This is especially true when trying to mimic the decision process of a senior military commander at the operational level of warfare[1]. As a result, the military training and analysis efforts have suffered.

In an effort to improve upon this modeling deficiency, a project was undertaken to develop a model that could replicate the human decision process based on an applicable cognitive decision theory. What resulted is a simulation called *RPDAgent*. The remainder of this article describes the design and implementation of RPDAgent and the process used to validate its behavior. It concludes with a comparison of decision results between humans and the model.

2. RPDAgent Design

The goal of the RPDAgent research was to produce a model that mimicked the decision process of an experienced decision maker in the military domain. The first step was to identify a cognitive theory that adequately explained this process. Classical decision theory was examined. This theory states that all decisions are rational choices that are made by optimizing the "payoff" of all possible outcomes. The theory was founded on economic and mathematical principles set forth by von Neumann [2]. However, decision makers do not always act in a rational manner [3]. Personal biases often affect a person's choice. Also, for classical decision theory to produce optimum decisions, all possible choices and the probabilities associated with the payoff of these choices must be known beforehand. This is often not the

[1] The operational level of warfare is concerned with planning, conducting, and sustaining campaigns and major operations as opposed to the tactical level, which concerns itself with fighting single battles.

case especially in a complex decision domain such as a military operation. However, decisions are made all the time with incomplete and conflicting information, a fact noted by Herb Simon [4] in his description of bounded rationality. So other psychological processes must be occurring to arrive at these decisions.

As an alternative to classical decision theory, *Naturalistic Decision Making (NDM)* was examined. NDM is defined as how people use their experience to make decisions in a field setting [5]. As a way to formally describe NDM, Klein developed the Recognition-Primed Decision (RPD) model to explain how people use their experience as the basis for decisions [6]. He observed that experienced decision makers handled approximately 50 to 80 percent of all decisions in this manner [7]. Pascual and Henderson saw strong evidence of RPD in the decisions made by operational military commanders [8]. Kaempf et al. [9] noted that 95 percent of all decisions made by naval officers on an AEGIS cruiser followed from RPD. With this evidence in hand, it was concluded that RPD would adequately describe a senior military commander's decision process. It was chosen as the basis for RPDAgent.

Next, a computational method had to be identified to implement RPD. Several methods were reviewed such as finite state machines, rule-based methods, case based reasoning, and neural networks. While these methods could have been used to implement RPD, they had some drawbacks. Finite state machines and rule-based methods are difficult to scale to large knowledge bases. Neural networks do not allow for tracking the model's reasoning process. Case based reasoning could not capture all aspects of the RPD process. Instead, a multiagent system (MAS) simulation approach was chosen because it embodied many characteristics of RPD [10]. Building on a *composite agent* developed by Hiles et al. at Naval Postgraduate School [11], a combination of symbolic and reactive agents were developed to capture the concepts of Klein's RPD model. This agent structure is depicted in the UML diagram of Figure 1.

RPDAgent was designed using a formal agent engineering methodology developed by DeLoach [12]. This methodology includes identifying system goals and use cases, matching agent roles to the goals, and creating specific agent tasks that help define the methods required for proper functionality. What follows is a brief description of each class within RPDAgent. All classes were implemented using the JAVA programming language.

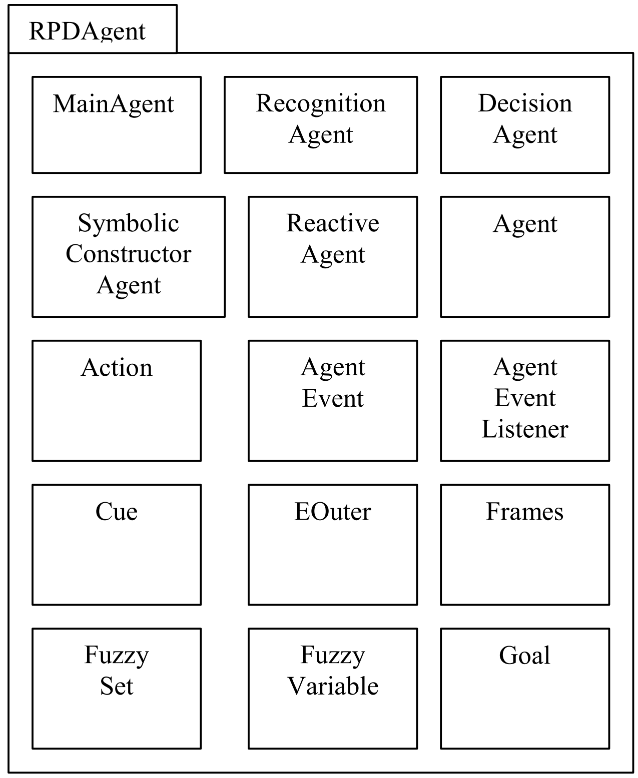

Figure 1. RPDAgent UML Diagram

The *MainAgent* class performs system management and user interface roles. However, its most crucial role is establishing and populating RPDAgent's experience database. The method employed to represent experience within the model is a key factor that accounts for its ability to replicate RPD. See the companion to this paper [13] for a treatment of experience representation in RPDAgent. Once the experience database is populated, RPDAgent waits for a decision request. It has two use cases for which it accounts. The first is a request for a decision not previously presented to it. The second is a reconsideration of a decision that is already active based on new or changing information.

Once a decision request is received, *RecognitionAgent* tries to match it with an existing experience via a table lookup method. If no matching experience is found, RPDAgent does not possess the experience to render a decision. If one is found, its associated data is retrieved and made available for use by the other agents. It then informs *SymbolicConstructorAgent (SCA)* of a pending decision request.

The SCA is responsible for creating an internal representation of the decision environment from the external information available to RPDAgent. This process represents a human's interpretation of his or her environment based on the stimuli provided and his or her past experience. That experience directly governs this interpretation.

With internalization complete, SCA instantiates a *DecisionAgent* to manage the remaining decision process. One DecisionAgent is provided for every active decision being considered by RPDAgent. DecisionAgent looks at the internal representation of the situation and the model's encoded experience and chooses the potential decision that appears most favorable. This is analogous to the concept in RPD of a decision maker knowing "intuitively" what action appears to work satisfactorily. DecisionAgent then instantiates one *ReactiveAgent* for every goal that RPDAgent is trying to achieve for the current decision situation.

As explained in the companion paper [13], a ReactiveAgent is responsible for evaluating how well a potential decision satisfies its assigned goal. Since it is autonomous, it does not care about how the other ReactiveAgents are viewing their goals and does not try to influence them. It is only concerned with evaluating how well the proposed decision satisfied its goal. Once its goal evaluation is complete, it informs its parent DecisionAgent of the evaluation. If all goals were satisfactorily met by the potential decision, that decision becomes the one that RPDAgent will render. If DecisionAgent sees that all goals were not met, DecisionAgent requests that the ReactiveAgents negotiate a compromise to achieve their goals. This concept is similar to humans weighing competing goals in their minds and deciding how much they are willing to compromise to achieve an end result. Humans usually have some threshold below which they will not negotiate. The same is true for the ReactiveAgents. If no compromise is reached, the next most favorable action is evaluated and the process is repeated. If no actions can be found to adequately satisfy the goals, RPDAgent renders a default decision appropriate to the situation. See the companion paper [13] for a more thorough explanation of the negotiation process.

There are several supporting classes listed in Figure 1. The *Frames, Cue, Goal, EOuter, and Action* classes all work together to form a major portion of experience representation. Their structure and methods are explained in Sokolowski [13]. The *Agent* class is a super class on which all other agents are based. It provides for agent data storage and for abstract methods to handle agent communication events. Agents communicate with a subset of the *Knowledge Query and Manipulation Language (KQML) [14]* using the event handling capabilities of JAVA and the *AgentEvent* and *AgentEventListener* classes. The *FuzzySet* and *FuzzyVariable* classes provide for the definition and functionality of fuzzy logic used in experience representation.

3. RPDAgent Validation Methodology

For a model to be accepted as an accurate representation of its real world system, it must be subjected to formal validation. RPDAgent's algorithms were validated against decisions produced by military officers playing the role of a Joint Task Force Commander in an operational decision scenario. The scenario consisted of a series of four decision points tied together with an overarching amphibious assault scenario. The scenario provided decisions that were typical of those faced by operational commanders. The decision points consisted of a decision on amphibious landing location, landing timing, a modification to the timing based on new information, and a decision on whether to continue the assault based on unpredicted enemy opposition and a high friendly casualty rate. This scenario was used as a test case for the model and as a proof of concept. It does not represent the complete experience base required by the model to make all plausible decisions in the military domain.

To gather the experience necessary to encode into RPDAgent, a *Cognitive Task Analysis (CTA)* of historical amphibious assaults was conducted. CTA provides a method to solicit knowledge and experience from subject matter experts [15]. A CTA of amphibious assaults from World War II through the Persian Gulf War was conducted [16,17,18]. This review provided the necessary information to identify the elements of experience associated with this type of military action. These elements were input into RPDAgent's experience database.

One can think of the data elements identified by the CTA as the training data for the model. These elements were identified through an independent survey of amphibious assaults and represent the experience elements that have historically influenced this type of scenario. Table 1 and Table 2 provide an example of the CTA data elements gathered for the location decision portion of the assault scenario. Table 1 shows the cues (one of the byproducts of recognition from Klein's model) associated with the location decision. It also shows the environmental variables and their value sets that influence the cues. Each landing sight under consideration would have specific values assigned to each environmental variable. These variables would then influence RPDAgent's assessment of the cues thus providing an evaluation of that landing site.

Table 2 lists the goals (another byproduct of recognition) and the cues that are linked to evaluating those goals. For each potential landing site, RPDAgent would have a set of cues that represents its evaluation of that site. The cue values are used to assess how well each landing site meets the goals set forth by RPDAgent. See Sokolowski [13] for a further discussion of the composition of RPDAgent's experience base and its methodology for decision-making.

Next, the scenario was provided to thirty military officers in the grades of O-4 to O-6. Twenty-one were U.S. military officers from all Services. Nine were coalition officers from various countries. All had joint operational military experience. Their responses to the scenario decision points were recorded and became the decision set against which the model would be evaluated. In addition to their decisions, they were also asked to explain the rationale behind them. This information was used to verify that the historical CTA adequately captured the experience in this area. In only one decision case did the experience data have to be significantly adjusted because of an incomplete CTA. This adjustment was to the underlying input data that was considered the training data for the model. No adjustments were made to RPDAgent to try and match model output to role player decisions. Therefore, the model setup remained independent of the test data from the role players.

The final step in the validation process required RPDAgent to render a decision for each decision point in the scenario. Since RPDAgent is a stochastic model, two hundred replications of the scenario were run to determine the distribution of its responses. Each replication represented thirty decision sets that mimicked the role player decision sets (one decision set represents the four decisions in the scenario). The number of role players making a particular decision was compared against the mean number of times the model made the same decision. This comparison was analyzed via the statistical method of *equivalency testing* [19].

When dealing with human variability, testing whether two statistics are precisely equal may not be meaningful or may be too restrictive. Equivalency testing provides a method to test whether the difference between two statistics is insignificant as defined by some predetermined absolute maximum difference. For RPDAgent, this difference was set at twenty percent of the number of role players making a particular decision. Twenty percent was chosen because it was not too wide a band to be unreasonable, yet it was not too narrow to allow for human variability. For RPDAgent, if the mean number of times a specific decision was made fell statistically within this defined difference, RPDAgent's decision results would be considered equivalent to the role players' decisions. See Rogers et al. [19] for a further explanation of equivalency testing.

Table 1. Location cues and associated environmental variables

Cues	Environmental Variables	Variable Values
Beach Topography	Steepness	Shallow, moderate, steep
	Sand type	Coarse, fine
	Obstacles	None, walls, jungle, rocks
Beach hydrography	Reefs	None, partial, full
	Water depth	Shallow, moderate, deep
	Suitable anchorage	Yes, none
	Tides	Small, moderate, large
	Currents	Small, moderate, severe
Water obstructions	Mines	No, yes
	Barriers	No, yes
Staging area	Staging area	Adequate, marginal, none
Route to objective	Route to objective	Adequate, marginal, inadequate
Enemy defenses	Level	Company, battalion, brigade
	Equipment	Light, moderate, heavy
	Enemy experience	Novice, experienced, professional
	Experience change	Decreasing, constant, increasing
	Enemy CAS	None, yes
	Enemy naval support	None, yes
Enemy perception of location	Perception	Unimportant, important, vital
Quality of intelligence	Intel quality	Excellent, good, poor
Location of landing site	Location	Near objective, away from objective

Table 2. Location goals and associated cues

Goals	Associated Cues
Achieve mission	Beach topography, beach hydrography, water, staging area, obstructions route to objective, location of landing site
Minimize casualties	Enemy defenses, enemy perception of location, quality of intelligence

4. RPDAgent Decision Results

The role player and model results for each of the four scenario decisions are presented here. The first decision point required the role players to select an amphibious landing location from among four possible sites. These locations were: Alpha, Bravo, Charlie, and Delta. Each location had specific characteristics that differentiated it from the others. No other landing locations were available to the decision makers. Based on the information provided in the scenario, the role players' landing location decisions were distributed as shown in Table 3.

Table 3. Role player location decision results

Location choice	Number of role players who chose
Alpha	0
Bravo	21
Charlie	4
Delta	5

RPDAgent was presented with the same decision and supporting data. After two hundred replications, the model produced a distribution of decisions given in Table 4 with each replication representing one set of thirty decisions. The min value indicates the minimum number of times the model made this decision over all replications. The max number represents the maximum number of times the model made this decision over all replications. The mean is the average number of times the model made this decision over the two hundred replications.

Table 4. Model location decision results

Choice	Min	Max	Mean	Std. Dev.
Alpha	0	0	0	0
Bravo	13	27	21.745	2.4185
Charlie	0	12	3.955	1.8976
Delta	0	11	4.3	1.9257

As mentioned previously, equivalency testing was used to compare the model results to the role player results. Under this test, one compares the difference between the role player result and the model result to determine if that difference is statistically within twenty percent of the role player value. The test statistic used was the student-t test with a Type I error level of 0.05 giving a critical test statistic value of 1.645. Table 5 shows the results of the statistical analysis. To be statistically equivalent, the magnitude of the test statistic must be greater than the critical value.

Table 5. Location decision equivalency analysis

Choice	Role player	Model	Std. Error	20% delta	t
Alpha	0	0	0	0	NA
Bravo	21	21.745	0.1710	4.2	-20.2
Charlie	4	3.955	0.1341	0.8	5.630
Delta	5	4.3	0.1361	1.0	2.204

The second decision point required the role players to make a decision on when to launch the amphibious assault. This timing decision would also affect other operations in the overall campaign scenario and thus was limited to four choices. Each choice provided a time interval in which to be ready. The four choices were: 36 hours, 48 hours, 72 hours, and 96 hours. Again, each choice had different factors that affected their overall support of an assault. Table 6 shows the distribution of decisions by the role players.

Table 6. Role player timing decision results

Timing choice	Number of role players who chose
36 hours	2
48 hours	27
72 hours	1
96 hours	0

RPDAgent was again presented with the information for this decision. Two hundred replications were run with the model results shown in Table 7.

Table 7. Model timing decision results

Choice	Min	Max	Mean	Std. Dev.
36 hour	0	6	1.87	1.4981
48 hour	22	30	27.04	1.7532
72 hour	0	4	1.09	0.9033
96 hour	0	0	0	0

Equivalency testing was applied to compare the role player results with the model results. This test used the same Type I error level as the location decision comparison, which provided the same critical test statistic. Table 8 shows the results.

Table 8. Timing decision equivalency analysis

Choice	Role player	Model	Std. Error	20% delta	t
36 hr.	2	1.87	0.1059	0.4	2.549
48 hr.	27	27.04	0.1239	5.4	-43.2
72 hr.	1	1.09	0.0638	0.2	-1.72
96 hr.	0	0	0	0	NA

The third decision point involved a change to the timing decision. Because of unexpected enemy troop movements, it appeared that the assault might have had to be conducted sooner. The role players were not constrained on what decision to make. Three decisions emerged from the group of thirty role players. The decisions were: go earlier, go on time, and change location. Their decision distribution is shown in Table 9.

Table 9. Role player sooner decision results

Sooner choice	Number of players who chose
On time	7
Go early	21
Change location	2

The model results for this decision point are shown in Table 10.

Table 10. Model sooner decision results

Choice	Min	Max	Mean	Std. Dev.
On time	3	14	7.2	2.0529
Go early	16	26	20.755	2.1395
Change location	0	6	1.825	1.2777

Table 11 provides the equivalency testing results for this decision point.

Table 11. Sooner decision equivalency analysis

Choice	Role player	Model	Std. Error	20% delta	t
On time	7	7.2	0.1451	1.4	-6.7540
Go early	21	20.755	0.1512	4.2	26.1574
Change location	2	1.825	0.0903	0.4	2.4917

The fourth and final decision point involved a situation that the amphibious assault force faced once the landing was completed. Because of inaccurate intelligence, the landing force met with unexpectedly high enemy opposition. The friendly casualty rate was unexpectedly high. The role players had to decide whether to continue the assault or to withdraw to the safety of the landing support ships. Table 12 provides the distribution of their decisions.

Table 12. Role player continue decision results

Continue choice	Number of role players who chose
Continue	21
Withdraw	9

Model results for this decision are shown in Table 13.

Table 13. Model continue decision results

Choice	Min	Max	Mean	Std. dev
Continue	12	28	22.09	2.3917
Withdraw	2	18	7.91	2.3917

Equivalency testing results for this decision point are provided in Table 14.

Table 14. Continue decision equivalency analysis

Choice	Role player	Model	Std. Error	20% delta	t
Continue	21	22.09	0.1691	4.2	-18.4
Withdraw	9	7.91	0.1691	1.8	4.199

In addition to the equivalency testing, a second validation approach was carried out. While equivalency testing provided an analysis of the individual decisions, it did not assess a possible pattern in the sequence of four decisions that the role players and the model were required to make. Did some pattern exist in this sequence that would allow a subject matter expert (SME) to distinguish the human decisions from those of the model? To answer this question, a test was devised to see if SMEs could make this distinction. This test was patterned after the Turing test first conceived by Alan Turning [20]. Five SMEs (three four star and two three star general officers) were each provided with twenty sets of the four decisions. These sets were selected at random from among the thirty role player decision sets and a group of thirty model decision sets. The SMEs were asked to determine which sets were from the human role players and which sets the model produced. Their selections are shown in Table 15. The first column represents each SME. The second column depicts the number of sets of data of which the SMEs could not tell whether it was human or computer generated. The third column lists the number of sets that the SMEs believe they could tell the difference. The fourth and fifth columns represent the number and percentage correct respectively out of the twenty sets.

Table 15. Turing test results

SME	Num. of can't tell	Number of human or model responses	Number of correct human or computer responses	Percent correct
Gen . A	20	0	0	0
Gen. B	6	14	8	40%
Gen. C	13	7	3	15%
Gen. D	18	2	2	10%
Gen. E	17	3	2	10%

When analyzing these results, one can compare the number of correct human or computer responses to Bernoulli trials of randomly guessing which is which. For this test, the expected number of correct responses is given by: S=np, where S is the expected number of correct responses, n is the number of trials, and p is the probability of success. Here, there are twenty trials corresponding to the twenty decision sets. Randomly guessing has a 50% probability of selecting human from computer. Therefore, the expected number of successes from guessing would be 10. From Table 15, one can see that the SMEs would have been better off in the long run by guessing which data was which. This result indicated that there was no hidden pattern in the data that could be discerned by a human that would readily identify computer-generated results.

5. Conclusion

Based on the equivalency statistics for each decision point and the Turing test results, it was concluded that the model produced decisions that were equivalent to the role player decisions. Therefore, the model was able to mimic the cognitive decision process used by these military officers playing the role of a Joint Task Force Commander. Of note, RPDAgent was able to replicate the variability of human decision-making and thus it can provide for a more realistic and human-like set of decisions. This capability would enhance the training and analysis value of existing and future simulation systems.

Further research is required to develop an experience database that would allow for the full domain of decisions facing an operational military commander. By identifying high-level cues that are representative of the type of information keyed on by these commanders along with the primary physical and mental parameters that are present in operational military situations, it is felt that this model could perform as a general decision-making entity. It could substitute for existing decision models in new and legacy simulation systems within the Department of Defense.

The author is grateful for funding support for this research provided by Joint Warfighting Center, U.S. Joint Forces Command and the Defense Modeling and Simulation Office.

6. References

[1] Pew, R.W. and Mavor, A.S. *Modeling Human and Organizational Behavior: Application to Military Simulations*. National Academy Press, Washington, D. C., 1998.

[2] von Neumann, J. and Morgenstern, O. *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, NJ, 1953.

[3] Tversky, A. and Kahneman, D. "Judgment Under Uncertainty: Heuristics and Biases," *Science*, Vol. 185, pp. 1124-1131, 1974.

[4] Simon, H.A. *The Sciences of the Artificial*. The MIT Press, Cambridge, MA, 1981.

[5] Zsambok, C.E. "Naturalistic Decision Making: Where are We now?," In *Naturalistic Decision Making*, Zsambok, C.E. and Klein, G. Eds., pp. 3-16, Lawrence Erlbaum Associates, Publishers, Mahwah, New Jersey, 1997.

[6] Klein, G. *Sources of Power: How People Make Decisions*. The MIT Press, Cambridge, MA, 1998.

[7] Klein, G. "Strategies of Decision Making," *Military Review*, Vol. 69, No. 5, pp. 56-64, 1989.

[8] Pascual, R. and Henderson, S. "Evidence of Naturalistic Decision Making in Military Command and Control," In *Naturalistic Decision Making*, Zsambok, C.E. and Klein, G. Eds., pp. 217-226, Lawrence Erlbaum Associates, Publishers, Mahwah, NJ, 1997.

[9] Kaempf, G.L. et al. "Decision Making in Complex Naval Command-and-Control Environments," *Human Factors*, Vol. 38, No. 2, pp. 220-231, 1996.

[10] Sokolowski, J.A. "Can a Composite Agent be Used to Implement a Recognition-Primed Decision Model?," In *Proceedings of the Eleventh Conference on Computer Generated Forces and Behavioral Representation*, Orlando, FL., May 7-9 2002. pp. 473-478.

[11] Hiles, J. et al. "Innovations in Computer Generated Autonomy," Tech Report NPS-MV-02-002, Naval Postgraduate School, Monterey, CA, 2001.

[12] DeLoach, S. "Analysis and Design Using MaSE and agentTool," In *Proceedings of the 12th Midwest Artificial Intelligence and Cognitive Science*

Conference, Oxford OH, March 31 - April 1 2001. pp. 1-7.
[13] Sokolowski, J.A. "Representing Knowledge and Experience in RPDAgent," In *Proceedings of the 12th Conference on Behavior Representation in Modeling and Simulation (BRIMS)*, Scottsdale, AZ, May 12-15 2003, (accepted for publication).
[14] Wooldridge, M. *An Introduction to MultiAgent Systems*. John Wiley & Sons, Ltd., West Sussex, England, 2002.
[15] Gorgon, S.E. and Gill, R.T. "Cognitive Task Analysis," In *Naturalistic Decision Making*, Zsambok, C.E. and Klein, G. Eds., pp. 131-140, Lawrence Erlbaum Associates, Publishers, Mahwah, NJ, 1997.
[16] Fuquea, D.C. "Bougainville: The Amphibious Assault Enters Maturity," *Naval War College Review*, Vol. L, No. 1, 1997.
[17] Alexancer, J.H. *Storm Landings*. Naval Institute Press, Annapolis, MD, 1997.
[18] Alexancer, J.H. and Bartlett, M.L. *Sea Soldiers in the Cold War*. Naval Institute Press, Annapolis, MD, 1995.
[19] Rogers, J.L., Howard, K.I., and Vessey, J.T. "Using Significance Tests to Evaluate Equivalence Between Two Experimental Groups," *Psychological Bulletin*, Vol. 113, No. 3, pp. 553-565, 1993.
[20] Turing, A. "Computing Machinery and Intelligence," *Mind*, Vol. 59, No. 236, pp. 433-460, 1950.

Author Biography

JOHN A. SOKOLOWSKI is a senior research scientist at the Virginia Modeling, Analysis & Simulation Center. Prior to this, he spent twenty-seven years in the Navy as a submarine officer. His last assignment was Head, Modeling and Simulation Division, Joint Warfighting Center, U. S. Joint Forces Command. He holds a Bachelor of Science degree in Computer Science from Purdue University, a Master of Engineering Management from Old Dominion University (ODU), and a Ph.D. in Engineering with a Concentration in Modeling and Simulation from ODU. His research interests include human behavior and decision modeling and multiagent system simulation.