Prepared for *The USMA Network Science Workshop*, 15-17 October 2008

High-level Behavior Representation Languages and Moderators of Behavior

Frank E. Ritter
College of IST
Penn State
University Park, PA 16802

***Abstract*—Network science has often analyzed networks as if they were made of homogenous, non-time varying, and unintelligent nodes. Network science should be expanded to include heterogeneous, time varying, and more intelligent nodes. This paper presents some results that can be used to make intelligent, heterogeneous nodes. Because these nodes are sensitive to behavioral factors, their performance varies naturalistically, resulting in time variations and more realistic behavior. In conjunction with this work, I explore how various factors might interact to create nodes that may either be willing or unwilling to participate within a given network.**

***Index Terms*—Agent-based modeling, Cognitive Modeling, Cognitive architecture, High-level Behavior Representation, Network Science, and Participation**

I. Introduction

Network science has often analyzed networks as if they were exclusively composed of homogenous, non-time varying, and unintelligent nodes. This approach has proven problematic when modeling human social networks and where intelligent agents are nodes. In this paper, I offer an agent-based approach for modeling heterogeneous, sensitive, and intelligent nodes as a step towards developing better predictive models of social networks. I present some results that illustrate this approach using two cognitive modeling languages and explore factors that influence behavior, specifically the willingness of nodes to participate within a network.

Real networks consist of different members. Fig. 1 shows, as just one example, that individuals performing a simple task such as subtraction vary both in how many calculations they can perform, and how accurately they perform them [1]. For more complex tasks, individuals will vary even further.

Additionally, the nodes in real networks possess varying degrees of knowledge. This knowledge not only drives behavior but also requires nodes to communicate to transfer that knowledge. In the past, it was difficult to model individual differences, much less the knowledge possessed by nodes (let alone any other systems).

Cognitive models make this possible. Drawing upon a unified theory of cognition, cognitive architectures now exist that enable researchers to model both differences and knowledge [2]. A cognitive architecture is an attempt to provide, in the form of a computer program, a realization of what mechanisms are fixed across tasks, such as working memory, perception, motor processes, and central cognition [3]. Researchers add knowledge about the task to the architecture to create cognitive models.

Currently, however, cognitive models are often (and fairly) seen as too difficult to create [4-6]. Thus, high-level languages for representing intelligent heterogeneous nodes are necessary.

This paper briefly introduces two languages (Herbal and CoJACK) for creating intelligent nodes quickly. CoJack, in particular, is designed to moderate its behavior based upon time and other analyst defined parameters. Both languages reflect a general architecture-based approach for developing agents in environments where network effects are important and visible, and where adversarial problem solving can be studied.

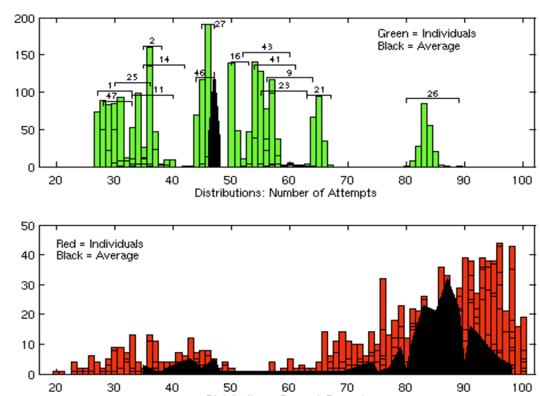

Fig. 1. How users vary when performing subtraction (light gray or green) and how a model (black) predicts that that individuals will vary in that they will have a more peaked distribution of performance than the whole subject distribution.

Manuscript submitted 25 October 2008. This work was supported in part by ONR Contracts N00014-03-1-0248, N00014-08-1-0481, and N00014-06-1-0164, as well as by the UK MOD's Directorate of Analysis, Experimentation, and Simulation corporate research program (Project No: RT/COM/3/006).
Frank E. Ritter is with Penn State's College of Information Science and Technology, University Park, PA 16802 USA (p: +1 (814) 321-3173, frank.ritter@psu.edu).

This approach offers a way to create nodes that more accurately reflect performance in the world by modeling factors such as stress, caffeine usage, and cognitive load. It also suggests that models of participation based upon interactional processes are needed for nodes to be more accurate. In other words, nodes performing more realistic tasks may choose to participate in some activities on some days and refuse on others.

II. Modeling Real World Behavior Using Herbal and CoJACK

A. High-level Behavior Representation Language (Herbal)

Herbal is a high-level behavior representation language [7]. The Herbal system has three objectives [8]. It is designed to make cognitive models easier to create, use, reuse, extend, and comprehend. It is designed to provide a user-centric cognitive modeling environment. Finally, it is designed to implement and test a theory of how to create high-level agents that, in effect, explain themselves by making design decisions explicit.

Herbal is based on HCI theory, Software Engineering, and empirical studies of modelers. Herbal implements the problem space computational model as an ontology which users can edit in Eclipse. This ontology is compiled using an XSLT script into Soar and Jess models concurrently. The code expansion from the XML representation used in Herbal to represent the PSCM is approximately 8. The Eclipse plugin includes descriptions of the components, as well as design information that provides a set of explanations that users typically request [8].

The Herbal system has been tested several times. Studies show that it is about 77% faster to create a model using Herbal versus using plain Soar. In addition, Herbal is easier to use for a wider range of users [9, 10].

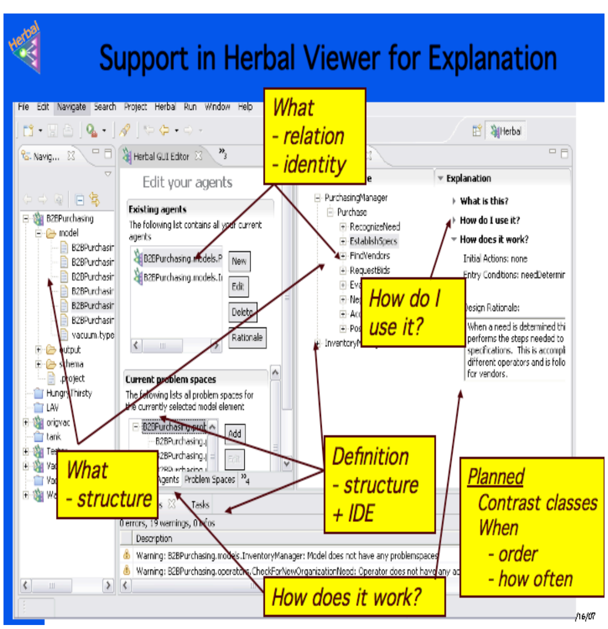

Fig. 2. Herbal interface, noting several of the explanation types provided.

Besides its accessibility and convenience, Herbal is a flexible language, capable of tackling a wide range of problems. It has been used to create user assistants (directive and non- directive) for the Rampart [11] anti-terrorism force protection planning environment [12]. It has been used to rapidly model multiple strategies for solving simple problems in a multi-problem problem solving task. The resulting models in this study matched the responses of most subjects where their problem solving strategies were known [13]. It has also been used to model adversarial reasoning, matching in this case, how likely users switch strategies when facing different pitchers in a simple baseball game [14]. We are currently using Herbal to develop agents in an IED production network.

Herbal offers a way to create Soar and Jess models more quickly and perhaps more accurately than in the past. Furthermore, Herbal agents offer a new ability. Herbal agents can explain themselves to their users and developers. Herbal does this by more explicitly representing the high-level structure in the model, and making it and the design rational more explicit. Thus, Herbal offers a way of making intelligent nodes without the fussiness and opaqueness associated with older cognitive modeling languages.

B. CoJack

CoJACK, our second language, is a Cognitive BDI Architecture. CoJACK offers another approach for creating intelligent nodes quickly [15-17]. It is based on the JACK agent architecture. Agent architectures are designed by programmers to create intelligent systems quickly.

CoJACK has been modified to include several aspects of Soar and ACT-R in JACK [18]. This makes the JACK architecture run more slowly, allowing agents to perform tasks at more realistic time intervals. CoJack also enables agents to make errors in declarative and procedural memory, provides a more detailed trace of an agent's cognitive performance, and includes a situation awareness component. Fig. 3. shows a schematic of CoJACK, specifically the modules used to model individual differences within the architecture. These modules include multiple parameters that users can modify statically or somewhat dynamically to represent effects such as caffeine or stress.

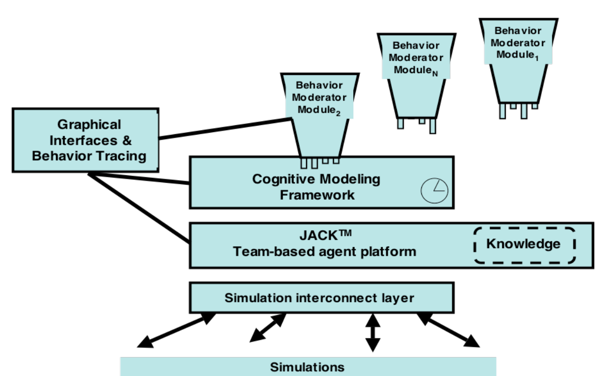

Fig. 3. CoJack schematic showing the underlying BDI agent architecture, JACK, and how the cognitive architecture can be modified with a set of parameter settings.

CoJACK has already been used to model networked agents. Evertsz, Ritter, Russell, and Shepherdson (2007) modeled the effects that Rules of Engagement (ROEs) have on performance using CoJack [19]. Evertsz, Busetta, Pedrotti, Ritter, and Bittner (2008) used CoJack to explore how moderators lead to differences in performance [20].

Both these studies implemented their agents within the dTank simulation tool, a simple system for rapid prototyping. We estimate that it is 100x faster to set up than OneSAF. It is not designed to be completely accurate (it assumes dead-reckoning and constant velocity on a flat surface) but to be somewhat accurate and to be easy to use, and to help with demonstrations of principle. dTank has been used for teaching undergraduates at several universities, to study the effects of communication in teams [21], and the effects of situational awareness (SA) [22]. An example dTank map is shown in Fig. 4.

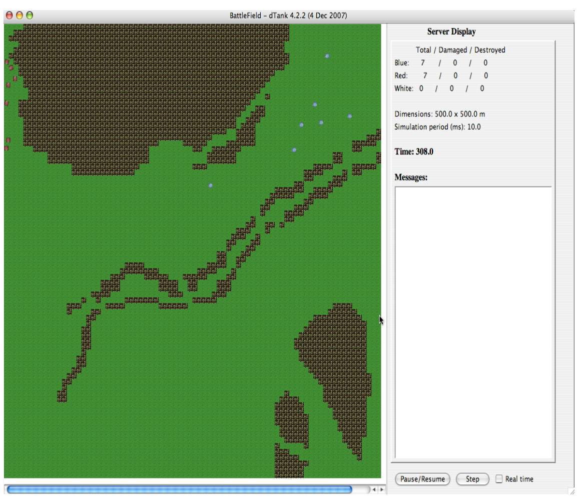

Fig. 4. Example dTank map showing the map of the Battle of El-Alamein.

In the 2008 study, the researchers pitted teams of CoJack and simple Java tanks against each other. Each team consisted of four tanks fighting on a plain map. Fig. 5. shows how they performed, as a measure of how many tanks out of the 4 on the other team were destroyed, where as Fig. 6. depicts individual differences in default action time and memory latency.

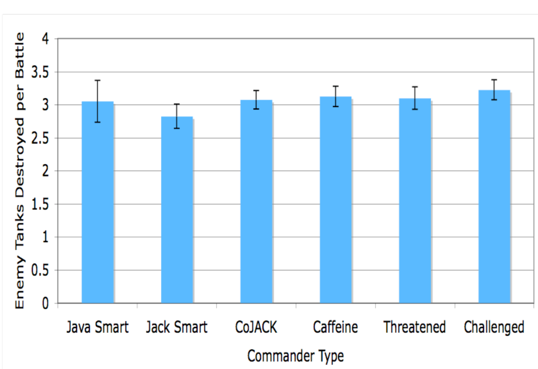

Fig. 5. Comparison of agents in dTank, including SEM bars based on 60 runs.

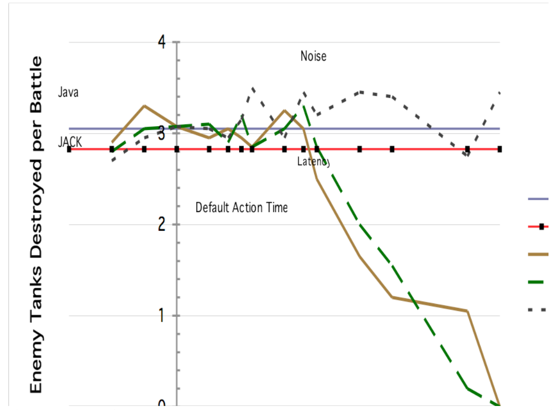

Fig. 6. How performance varies as procedural knowledge noise, default action time, and memory latency are varied from the default, from 0.5 to 20.

III. Modeling Participation

After reviewing David Grossman's work [23], among others, it was apparent that more accurately modeling participation in synthetic environments was possible in dTank. It is possible to explore one of Grossman's chief claims, namely: *soldiers do not always want to participate*.

After an initial implementation, it is possible to generalize this claim to some extent to include agents operating in insurgent networks. An agent-based approach augmented by a model of participation would also compliment and enhance current models of asymmetrical warfare by making the agents' decision making processes more explicit and more context dependent. Additionally, such an approach would benefit the Army's current simulation based training by making agent behavior and outcomes more realistic [24].

Grossman suggests that several factors influence one's willingness to participate, such as distance to the enemy. To test his approach, we implemented a theory of participation in a light weight simulation (dTank) that complements current agent-based approaches to modeling squad/team behavior [6, 23, 25]. Fig. 7 shows a trace of the average participation score of agents in a simple battle.

The participation score is based on distance from the team's leader, enemy, and fellow team members; attractiveness (defined as the ratio between friendly and opposing forces); a randomly generated predisposition score; and a default score for training. The simulation was of two infantry teams, each consisting of four combatants. In future simulations, we hope to better model the influence of training, as well as capture the impact of crew-served weapons and suppressive fire.

The figure shows that agents that used this score to choose to participate would behave differently than agents that always participated (shot at an enemy). We believe but have not yet conclusively proven that this would lead to more realistic agent behavior overall. It would, at least, stop agents from shooting at very large groups when alone, and would make agents in large groups more aggressive.

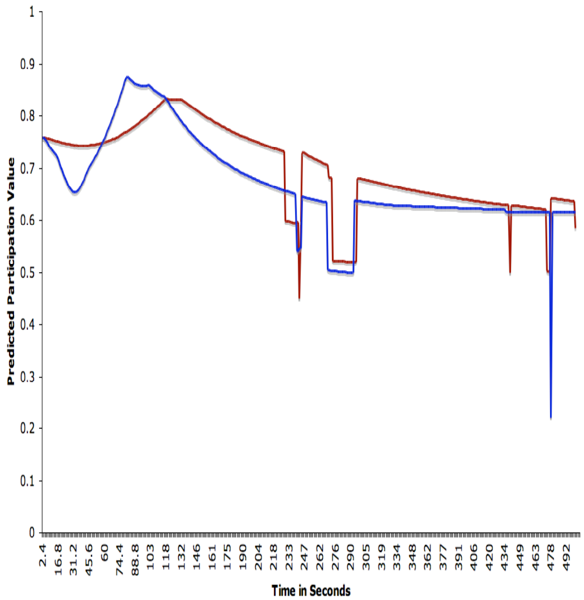

Fig. 7. Trace of predicted participation for a simple scenario in dTank.

While not finished, we have gathered several insights. The function is not as smooth as one might expect. After participating, the predicted participation value may drop or rise. It may rise if the opponent is killed (because that will move the ratio of forces in the combatant's favor), and it could decrease if the agents suddenly become aware of their proximity to a new opponent (a sudden and unprepared for decrease in distance). Finally, while this value was computed within the simulation, it is clear that more meaningful values must be based on what the agent knows, has learned, and can remember about the world. Doing this will make participation values reflect not only the influence of the interactional factors that Grossman identifies but also cognitive effects more generally.

IV. Conclusion

This paper has briefly reviewed an agent-based approach for capturing knowledge and individual differences in a network simulation. We discussed two languages (Herbal and CoJack) as two methods of realizing this approach, and discussed dTank as a light-weight environment well suited for implementing it. Finally, we identified a potential extension and application of this work, the modeling of agent participation.

This paper examined how to capture variability, whether in the form of stress, interactional factors, or knowledge retention rates, in intelligent models. These variations across agents (nodes) can help create networks that more accurately reflect the differences that actually exist between members of real networks, as well as differences between different networks. This work even provides a basis for modeling how a network collectively will respond to time and stressors.

Our work suggests that military and paramilitary networks, in particular, are sensitive to variation in factors such as team and leader proximity, social and physical distance, training, and attractiveness. These factors, among other cognitive effects, appear to influence daily performance, and must be captured in future simulations to advance the state of the art.

This work provides several opportunities. The Herbal language and the CoJACK architecture can be extended. While both have been used, their use in network simulations can be deepened to support network related cognition more directly and fully.

The use of the participation scores is in development. This, too, could be extended to more accurately reflect performance across a wider range of scenarios, and then more directly incorporated into future agent models. Finally, these effects could be combined with stress, knowledge, and cognition to more accurately model both how agent participation influences larger networks and how those networks in turn respond to such stressors. Using these results might prove helpful in exploring network interaction and integrity.

Acknowledgment

Support was provided by ONR Contract N00014-03-1-0248, N00014-06-1-0164, and N00014-08-10481 and by the UK MOD's Directorate of Analysis, Experimentation and Simulation corporate research programme (Project No: RT/COM/3/006). Discussions with my cited co-authors, as well as Paul Bello, Susan Chipman, Harold Hawkins, John Horgan, Kevin Murphy, Colin Sheppard, Lora Weiss, and Elizabeth Whitaker, have contributed to these ideas. Jeremiah Hiam and Geoffrey Morgan created and modified the participation simulation, while Jonathan Morgan provided theoretical input and some editorial assistance. Discussions with the workshop participants, particularly Kathleen Carley and John James, have informed my thinking.

References

Papers by Ritter are available at acs.ist.psu.edu/papers.

[1] S. E. Kase, F. E. Ritter, and M. Scholles, "From modeler-free individual data fitting to 3-D parametric prediction landscapes: A research expedition," In: Proceedings of the 30th Annual Conference of the Cognitive Science Society, 1398-1403. Cognitive Science Society: Austin, TX 2008.

[2] G. Jones and F. E. Ritter, "Production systems and rule-based inference," in Encyclopedia of cognitive science, vol. 3, L. Nadel, Ed. London: Nature Publishing Group, 2003, pp. 741-747.

[3] A. Newell, Unified Theories of Cognition. Cambridge, MA: Harvard University Press, 1990.

[4] F. E. Ritter, N. R. Shadbolt, D. Elliman, R. M. Young, F. Gobet, and G. D. Baxter, Techniques for modeling human performance in synthetic environments: A supplementary review. Wright-Patterson Air Force Base, OH: Human Systems Information Analysis Center (HSIAC), 2003.

[5] R. W. Pew and A. S. Mavor, "Modeling human and organizational behavior: Application to military simulations." Washington, DC: National Academy Press. books.nap.edu/catalog/6173.html, 1998.

[6] R. W. Pew and A. S. Mavor, "Human-system integration in the system development process: A new look." Washington, DC: National Academy Press. books.nap.edu/catalog.php?record_id=11893, 2007.

[7] M. A. Cohen, F. E. Ritter, and S. R. Haynes, "Herbal: A high-level language and development environment for developing cognitive models in Soar," In: Proceedings of the 14th Conference on Behavior Representation in Modeling and Simulation, 05-BRIMS-043. Orlando, FL, 2005.

[8] S. R. Haynes, M. A. Cohen, and F. E. Ritter, "A design for explaining intelligent agents," International Journal of Human-Computer Studies, in press.

[9] M. A. Cohen, "Herbal: Improved modeling through using a High-level behavior representation language to support Reuse and Maintenance," unpublished PhD thesis, College of Information Sciences and Technology, The Pennsylvania State University, 2008.

[10] G. P. Morgan, M. A. Cohen, S. R. Haynes, and F. E. Ritter, "Increasing efficiency of the development of user models," In Proceedings of the IEEE System Information and Engineering Design Symposium, Ellen J. Bass, (ed). IEEE and Department of Systems and Information Engineering, University of Virginia: Charlottesville, VA 2005.

[11] S. R. Haynes, T. G. Kannampallil, L. L. Larson, and N. Garg, "Optimizing anti-terrorism resource allocation," Journal of the American Society for Information Science and Technology, vol. 56, pp. 299-309, 2005.

[12] S. R. Haynes, T. G. Kannampallil, M. A. Cohen, and F. E. Ritter, "Rampart: A service and agent-based architecture for anti-terrorism planning and resource allocation," To be presented at EuroISI 2008, European Conference on Intelligence and Security Informatics, Lecture Notes in Computer Science series. Berlin, in press.

[13] M. B. Friedrich, "Implementing diagrammatic reasoning strategies in a high level language: Extending and testing the existing model results by gathering additional data and creating additional strategies," vol. Masters thesis: Faculty of Information Systems and Applied Computer Science, University of Bamberg, Germany, 2008.

[14] M. A. Cohen, F. E. Ritter, and S. R. Haynes, "Using reflective learning to master opponent strategy in a competitive environment," In Proceedings of the 8th International Conference on Cognitive Modeling, 157-162. Oxford, UK: Taylor & Francis/Psychology Press, 2007.

[15] F. E. Ritter and E. Norling, "Including human variability in a cognitive architecture to improve team simulation," in Cognition and multi-agent interaction: From cognitive modeling to social simulation, R. Sun, Ed. Cambridge, UK: Cambridge University Press, 2006, pp. 417-427.

[16] E. Norling and F. E. Ritter, "A parameter set to support psychologically plausible variability in agent-based human modelling," Proceedings o f The Third International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS04), 758-765. New York, NY: ACM. 2004.

[17] F. E. Ritter, R. Evertsz, P. Busetta, M. Pedrotti, and J. L. Bittner, "CoJACK: Achieving Principled Behaviour Variation in a Moderated Cognitive Architecture. Joint Systems and Analysis Group, Modeling and Simulation Technical Panel, Computer Generated Forces Workshop. The Technical Cooperation Program (TTCP)," 2008.

[18] E. Norling and F. E. Ritter, "COJACK Software Specification," Agent Oriented Software Limited 2004.

[19] R. Evertsz, F. E. Ritter, S. Russell, and D. Shepherdson, "Modeling rules of engagement in computer generated forces," Proceedings of the 16th Conference on Behavior Representation in Modeling and Simulation, 80-89. 08-BRIMS-025. Orlando, FL: U. of Central Florida. 2007.

[20] R. Evertsz, P. Busetta, M. Pedrotti, F. E. Ritter, and J. L. Bittner, "CoJACK—Achieving principled behaviour variation in a moderated cognitive architecture," Proceedings of the 17th Conference on Behavior Representation in Modeling and Simulation, Orlando, FL, 2008.

[21] S. Sun, I. G. Councill, X. Fan, F. E. Ritter, and J. Yen, "Comparing teamwork modeling in an empirical approach," Proceedings of the Sixth International Conference on Cognitive Modeling, 388-389. Mahwah, NJ: Erlbaum. 2004.

[22] F. E. Ritter, S. E. Kase, D. Bhandarkar, B. Lewis, and M. A. Cohen, "dTank updated: Exploring moderator-influenced behavior in a light-weight synthetic environment," Proceedings of the 16th Conference on Behavior Representation in Modeling and Simulation, Norfolk, VA, 2007. 51-60. Orlando, FL: U. of Central Florida. 07-BRIMS-014.

[23] D. Grossman, on killing: The psychological cost of learning to kill in war and society. New York, NY: Back Bay Books, Little Brown and Company, 1996.

[24] J. Horgan, The psychology of terrorism. London: Routledge, 2005.

[25] R. Collins, Violence: A micro-sociological theory. Princeton, NJ: Princeton University Press, 2008.

[26] F. E. Ritter, S. R. Haynes, M. Cohen, A. Howes, B. John, B. Best, C. Lebiere, R. M. Jones, J. Crossman, R. L. Lewis, R. St. Amant, S. P. McBride, L. Urbas, S. Leuchter, and A. Vera, "High-level behavior representation languages revisited," Proceedings of ICCM - 2006- Seventh International Conference on Cognitive Modeling, Trieste, Italy, 2006. 404-407. Trieste, Italy: Edizioni Goliardiche.