In F. Detje, D. Doerner, & H. Schaub (Eds.), Proceedings of the Fifth International Conference on Cognitive Modeling. 2003. 63-68. Bamberg, Germany: Universitats-Verlag Bamberg.

Explaining Soar: Analysis of Existing Tools and User Information Requirements

Isaac G. Councill (igc2@psu.edu)
Steven R. Haynes (shaynes@ist.psu.edu)
Frank E. Ritter (ritter@ist.psu.edu)
School of Information Sciences and Technology
The Pennsylvania State University
University Park, PA 16802

Abstract

Providing models with explanation facilities to make the rationale for model behavior (both "external", with respect to the context of a simulation domain, and "internal", the structure and cognitive processes of models) would make them more accessible. In an effort to understand the potential explanation information requirements of an example population, data collected from a usability study of the TacAir-Soar Situation Awareness Panel was analyzed to determine where breakdowns occurred in users' attempts to understand this complex cognitive model. The analysis suggests that plan view displays of models' situation awareness are useful but users need the displays to be augmented with information about working memory changes over time, with actions both taken and not taken by models. In addition, explanation facilities of how and why behavior occurs appear to be necessary for most architectures.

Introduction

Cognitive models are increasingly used outside of the research lab. They are often used in synthetic environments to simulate the effects of expert human information processing (Pew & Mavor, 1998; Ritter, Shadbolt, Elliman, Young, Gobet, & Baxter, in press; also see numerous applications in the Computer Generated Forces Conferences), and in various training domains (e.g., Zachary, Jones, & Taylor, 2002; Anderson, Corbett, Koedinger, & Pelletier, 1995).

However, the impact of cognitive models in applied settings is currently limited by their usability. As others and we have noted before (Gluck & Pew, 2001; Pew & Mavor, 1998, p. 282, 292; Ritter et al., in press) cognitive models suffer from a range of non-trivial usability problems, such that the difficulties inherent in simply creating and testing models of behavior preclude us from spending more time identifying and realizing opportunities to implement models in applied settings, and then analyzing the efficacy of these models as implemented in practice. This problem is increasingly recognized in the modeling community.

Modelers interact with a given model many times and in many ways during the processes of model development, testing, and implementation. Complex cognitive models must be easy to create, test, refine, and reuse. As part of the model creation and validation process, models must be debugged on the syntactic level (will it run?), on the knowledge level (can it perform the task?), and on a behavioral level (does it perform the task like a human?). While we can point to some recent advances in usability (Anderson & Lebiere, 1998; Cooper & Fox, 1998; Jones, 1999; Kalus & Hirst, 1999; Ritter, Jones, & Baxter, 1998; Zachary, Ryder, Ross, & Weiland, 1992), further work is required to achieve what we perceive to be necessary efficiencies.

One of the reasons cognitive models have usability problems is because they represent both a theory of mind and an architecture-dependent representation of domain knowledge. In applied domains, perhaps more so than in pure research, models must be understood by people who interact with them as team members, domain and task experts, as advisors in training systems, and as users who are not experts in the model technologies. Difficulty in understanding models' behavior is partially due to their complexity, but this complexity is compounded by model development tools that do not provide a structured approach to model creation, and that fail to provide features for exposing and supporting exploration of a model's state by non-programmers. This situation has obviously arisen not intentionally, but as a result of researchers' focus on other aspects of cognitive models.

One method proposed to enhance the usability of systems operating at the knowledge level is to embed in them explanation facilities that draw on the structure and content of the systems themselves as the knowledge base used for explanation. Efforts to develop explanation facilities emerged from the early Mycin experiments with expert systems (Buchanan & Shortliffe, 1984) and their contribution to system usability is suggested by subsequent studies (Stylianou, Madey, & Smith, 1992). These studies suggest that explanation facilities may be especially important for expert users in complex domains (Ye & Johnson, 1995), though there are indications that even novices desire more information about complex domains than is often presumed (Forsythe, 1995). However, the apparent context-dependence of user information requirements introduces significant challenges to the task of identifying what information users need, and when and how to deliver it. Compounding the problem are pragmatic issues related to the additional project overhead introduced when explanation-generation software development and explanation knowledge engineering (elicitation and representation) are appended as discreet (and difficult) tasks to the model-building process.

This study is a first step in a project to design and implement software that will support the inclusion by model developers of explanation facilities into cognitive model applications developed within a variety of cognitive modeling architectures.

The TacAir-Soar SAP Usability Study

Uncovering which aspects of a model users and developers are most interested in having explained provides a starting point for the design of tools to meet these information requirements. However, explaining one's own behavior can be a difficult task (Ericsson & Simon, 1993). We approach this problem through an examination of the questions expert users ask of a relatively complex cognitive model (TacAir-Soar) while they perform a range of tasks in their domain.

TacAir-Soar was developed to provide virtual partners in military training simulations, reducing the need for expert human role-players during training exercises. TacAir-Soar agents operate simulated aircraft autonomously during the execution of dynamic missions developed as part of the Joint Semi-Autonomous Forces (JSAF) simulation.

The TacAir-Soar Situation Awareness Panel (SAP) is a graphical utility designed to allow users of TacAir-Soar to access and gain an enhanced understanding of the internal state of their agent partners. The SAP also supports a limited view of an agent's operating history in terms of goals and milestone events.

In an effort to understand and improve the SAP, a previous study had participants perform a set of basic tasks supported by the SAP during a preprogrammed scenario, and provide comments regarding how well these tasks are supported by the SAP and its components (Avraamides & Ritter, 2002). A list of suggested changes was passed back to the SAP developers. Participants included four expert SAP users and eight experts from a variety of domains including cognitive psychology, geographical information systems, HCI, software development, and the military.

During the study two versions of the SAP were available: the version that was at the time deployed in actual training exercises, and a newer software version (see Figure 1) that, due to the timing of the release, was shown to only the four expert SAP users in the study. The deployed version and the enhanced version shared many common characteristics, including a goal stack display, a milestone display, and a viewport that represents an agent's awareness of nearby objects in the simulation as a plan view display.

The newer SAP included improvements, both in the interface design and in the content provided to users. Included in the new SAP was an increased level of detail in the information displayed within the viewport, and a feature known as the *amplified goal display*, described below.

It is important to note that most of the tools implemented on the SAP are not, strictly speaking, explanation facilities. The milestone, viewport, and goal stack displays indicate "what" is in an agent's awareness and roughly "what" behavior an agent is engaging in. All explanations for agent behavior ("why" the agent believes something to be true or "why" the agent is behaving in some way) must be generated ad hoc by users of the SAP based upon the model's observable behavior of agents as well as users' beliefs about the implementation of TacAir-Soar.

The one exception is the amplified goal display. In response to mouse clicks over specific goals in the goal stack, the amplified goal display presents the reason that TacAir-Soar selected the goal, alternative goals that would have been chosen given changes to the agent's state, and hyperlinks to documentation relating to each goal.

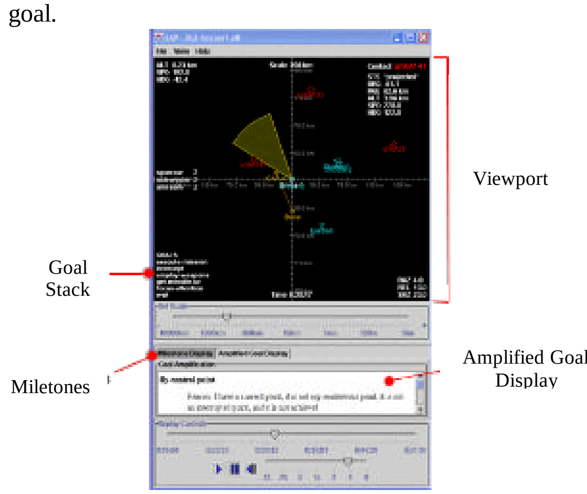

Figure 1. The later of the two versions of the SAP.

Study Method

Approximately seven hours of video recordings from the SAP usability study were transcribed and then analyzed for patterns in user information requirements. Due to significant variation between data collected from participants who were domain experts and other participants, only the four domain experts' transcripts are included in the present analysis.

The transcripts were reviewed to identify utterances relevant to explanation of agent behavior. Relevant material was defined as any utterance related to existing or imagined tools that assist users to access the knowledge embedded within TacAir-Soar agents, expressing evidence of a need for such tools, or expressing any evidence regarding the usefulness of agent behavior explanations.

Relevant passages were analyzed in order to identify emergent patterns. Codes to identify utterances were developed and refined during iterative reviews of the transcripts. These codes were applied to specific utterances yielding a hierarchical organization of the utterances with discreet categories.

The Resulting Code Tree

The Model/View/Controller (MVC) framework for user interface development (Krasner & Pope, 1988) was used as a conceptual base for classifying utterances. Briefly stated, MVC separates program components into three distinct classes: the *model* is all of the information contained within the application itself; *views* are the onscreen displays of selected portions of that information to the user(s); and *controllers* implement the reactivity of views to user input.

In order to enhance the clarity of our presentation and restrict the MVC concept to the explanation domain, we have renamed the first two components of the MVC. The model component we refer to as the "explanatory content" of programs and we refer to views as "delivery mechanisms." Though we feel that the controller component of MVC will be of great importance to the design of effective explanation facilities, we will ignore controllers for the time being. Controllers represent a level of granularity in design that is beyond the scope of the present study.

The portions of the transcripts deemed relevant were classified into two main categories: utterances related to explanatory content within the TacAir-Soar and its simulated world and the delivery mechanisms (data views) implemented in the SAP. Here the term "explanatory content" is defined broadly, but generally includes the triggers for an agent's behavior (i.e., those conditions that caused a specific goal to be put on Soar's goal stack) along with the "intentions" of an agent (e.g., an agent is traveling south at so-and-so speed because the agent is chasing a particular bogey), as well as the causes for events that befall an agent (e.g., an agent was shot down because that agent was not "aware" of the MiG closing in from behind).

Table 1 provides a description of all of the codes within the code tree, along with the relationships among codes. Included with each code is the number of utterances classified with the code and the percentage of the transcripts that the coded utterances represent (calculated by line count).

Discussion

The results of this study (summarized in Table 1 below) indicate a need for improved tools and techniques for the visualization of agent cognition. Though participants unanimously found the SAP to be useful in explaining agent behavior, all participants made suggestions for improvements or indicated some level of dissatisfaction with the SAP.

Utterances in the category regarding the need for explanations of agent behavior validate our basic premise that TacAir Soar users require additional information to help them understand its behavior. Participants in the study were unanimous in claiming that a major use of the SAP is in fact to derive explanations for agent actions. A critical theme that emerged from participants' comments was that questionable behaviors undertaken by a TacAir-Soar agent lead to a need by users to determine whether that behavior is a result of intended agent behavior or faulty programming on the part of the developers.

At the time of this study, SAP users derived explanations from the contents of the SAP – an interpretive process lacking explicit support from the interface. The analysis summarized in Table 1 suggests that the most used element in the users' process of explanation-derivation is the viewport, arguably the most "intuitive" component of the SAP. This is an indication of the success of the viewport as a tool for agent display. However, the viewport only represents the model's declarative knowledge.

The only component of the original SAP that explicitly represents information related to the process of agent cognition, the goal stack display, was declared to be useless by half of the participants and of only marginal utility with need for improvements by the other half. All participants indicated that they did not fully understand the contents of the goal stack. The reasons for the unilateral dissatisfaction with the goal stack display are unclear and the opinions of our participants varied significantly, representing the views that the goal stack contains too much information, too little information, or the wrong information. We may conclude, however, that the goal stack display and the processes associated with it are not clear to the participants in this study. This is alarming in that all knowledge of the process of and reasons behind agent cognition must be inferred by an observer relying on assumptions about the implementation details of TacAir Soar agents. This is likely to be a concern for other architectures that rely on a goal stack to convey process information.

The implementation in the newer SAP of the amplified goal display was generally regarded as a major improvement to the previous goal information (indicated by four out of six utterances regarding the feature). Participants remarked on both the increased

Code	Number	Percent	Description
Explanatory Content	24	14	Indicates that a participant would benefit from the inclusion of explanation features into the SAP and/or the inclusion of additional explanatory information in the SAP
Need for Explanation	14	9	Explicitly indicates that a participant would benefit from access to the reasons why an agent is or is not engaging in a behavior
Request for Additional Content	10	5	Indicates that a participant would benefit from the inclusion of additional explanatory information in the SAP
✍ Beyond Model Scope	5	3	Request for information that would not likely (or would not) be in a human pilot's awareness
✍ Within Model Scope	5	2	Request for the inclusion of material into the SAP that would likely be information within a human pilot's situation awareness
Delivery Mechanisms	86	57	Indicates a participant's opinion on the degree to which existing agent display tools implemented in the SAP are useful for deriving explanations of agent behavior
Milestones	6	2	Comment regarding the usefulness of the milestone feature
✍ Clarity	2	1	Opinion regarding the clarity with which the milestone feature presents information
✍ Content	4	2	Opinion regarding the usefulness of the information contained in the milestone feature
Goal Stack	12	8	Comment regarding the usefulness of the goal stack feature
✍ Clarity	2	2	Opinion regarding clarity with which the goal stack tool presents information
✍ Content	10	6	Opinion regarding the usefulness of the information contained in the goal stack feature
Amplified Goal Display	6	3	Comment regarding the usefulness of the amplified goal display
Proposal for New Delivery Mechanism	6	8	Participant proposed a new tool for incorporation into the SAP
Viewport	56	36	Comment regarding the usefulness of the viewport feature
✍ Scale and Orientation	8	8	Comment related to the display of scale (in miles or km) and spatial orientation within the viewport
? User Ergonomics	6	4	Comment having to do with the usability of the viewport
? Scale as Awareness	2	4	Comment regarding the scale of the viewport as an aspect of agent awareness
✍ Supplemental Information	22	11	Comment regarding the textual information displayed around the perimeter of the viewport, such as aircraft speed, heading, altitude, contact information, etc.
? Usefulness of Existing Information	16	7	Opinion regarding the usefulness of the information presented
? Request for Additional Information	6	4	Request or suggestion to include additional information alongside the information already presented
✍ Symbols	26	18	Comment regarding the icons within the viewport (i.e. the graphical representations of planes, route features, etc.)
? Usefulness of Existing Symbols	12	7	Opinion regarding the usefulness of symbols presented within the viewport
? Request for Additional Symbols	14	11	Request for the addition of symbols representing agent knowledge into the viewport

Table 1. The code tree developed in the study, with the number of occurrences of the codes in the transcripts, percentage of the transcripts (calculated via line count), and descriptions of the codes.

clarity of the information provided within the amplified goal display and the explanation of competing goal selection as factors in their approval. Two comments indicated that participants found no use for the feature. However, the feature was new to all participants and little time was available to explore the display during this study. It remains to be seen how the feature is used in practice and to what extent it assists users to understand the processes of the model.

The frequency of participant comments related to the viewport, as well as the numerous requests that were made for additions to this feature raise some interesting questions about enhancing the functionality and scope of the SAP. Future directions in the development of the SAP may expose a tension between the goals of psychological realism of agent simulations versus their utility as software tools.

The SAP makes plain the fact that there may be more active elements in a TacAir-Soar agent's working memory than would be realistic to expect of a human in similar circumstances (see Figure 1). Even so, each participant would have preferred more information regarding all aspects of the agent's true situation in addition to that displayed on the SAP, which represents the situation as "perceived" by the agent. Out of the 56 utterances regarding the viewport and all of the information that it encapsulates, no comments indicated that participants would prefer less information than that given and 20 of those utterances indicated that participants want even more information within the viewport. In addition, all three participant-initiated proposals for new delivery mechanisms to assist explanation derivation were presented as extensions to the viewport. Given these observations, we believe the SAP would be less useful if it indicated a more realistic working memory environment in the agents whose awareness it displays.

This situation may be exemplified by an account of the scan patterns that pilots are trained to undertake. A scan pattern is the continual shifting of a pilot's attention over various instruments that display information necessary to the pilot's effective performance, and in some cases, their survival. Scan patterns were developed to address the fact that pilots are required to attend to more information than can be stored in a human's working memory at any one time. Scan patterns help pilots ensure that even though they cannot consider all the information relevant to their situation at one time, no potentially important information goes unconsidered for more time than it takes to perform a scan cycle.

TacAir-Soar currently fails to account for the complexity of scan patterns and abstracts pilot behavior so that all information that would be available to a pilot through various instruments is fed directly into Soar's working memory, bypassing normal human attentional processes. We contend that a realistic model of human attention is required to improve the correspondence between cognitive models and actual human behavior. As progress towards this goal is achieved and the working memory of models becomes as dynamic as that of actual humans, devices such as the SAP that display only the contents of an agent's working memory at any instant will become confusing indeed, thus increasing users' need for access to the state of the environment in addition to the simulated sense data from which the working memory of an agent is derived.

In order to reconcile the potentially divergent objectives of achieving increasingly realistic models of human cognition and developing useful mechanisms for displaying the situation awareness of these models, we believe efforts should focus on development of a general mechanism that allows users to access the working memory and processes of an agent in a structured way while supporting access to objects within an agent's simulated environment that are outside of the agent's current awareness. As indicated by our participants' experience with the TacAir-Soar SAP, explanations of model behavior require not only access to agent knowledge that is currently active, but also to the antecedents and context-dependent factors that give rise to this behavior.

Acknowledgements

This project was supported by the US Office of Navy Research, award number N00014-02-1-0021.

References

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. *Journal of the Learning Sciences*, 4, 167-207.

Anderson, J. R. & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, NJ: LEA.

Avraamides, M. N. & Ritter, F. E. (2002). Using multidisciplinary expert evaluations to test and improve model interfaces. *Proceedings of the 11th Computer Generated Forces Conference*, 553-562. Orlando, FL: U. of Central Florida.

Buchanan, B. G. & Shortliffe, E. H. (Eds.). (1984). *Rule-based expert systems*. Reading, MA: Addison-Wesley.

Cooper, R., & Fox, J. (1998). COGENT: A visual design environment for cognitive modeling. *Behavior Research Methods, Instruments and Computers*, 30, 553-564.

Ericsson, K. A. & Simon, H. A. (1993). *Protocol analysis: Verbal reports as data*, 2nd ed. Cambridge, MA: MIT Press.

Forsythe, D. E. (1995). Using ethnography in the design of an explanation system. *Expert Systems with Applications*, 8:403-417.

Gluck, K. A., & Pew, R. W. (2001). Overview of the

Agent-based Modeling and Behavior Representation (AMBR) model comparison project. *Proceedings of the 10th Computer Generated Forces and Behavior Representation Conference*, Orlando, FL.

Jones, R. M. (1999). Graphic visualization of situation awareness and mental state for intelligent computer-generated forces. *Proceedings of the 8th Conference on Computer Generated Forces*, Orlando, FL.

Kalus, T. & Hirst, T. (1999). ViSoar. *Proceedings of the Soar Workshop 19*, Ann Arbor, MI.

Krasner, G. E. & Pope, S. T. (1988). A cookbook for using the Model-View-Controller user interface paradigm in Smalltalk-80. *Journal of Object-Oriented Programming* (August), 26-49.

Pew, R. W. & Mavor, A. S. (Eds.). (1998). *Modeling human and organizational behavior: Applications to military simulations*. Washington, DC: National Academy Press.

Ritter, F. E., Jones, R. M., & Baxter, G. D. (1998). Reusable models and graphical interfaces: Realising the potential of a unified theory of cognition. *Mind modeling – A cognitive science approach to reasoning, learning, and discovery*. Lengerich: Pabst Scientific Publishing, 83-109.

Ritter, F. E., Shadbolt, N. R., Elliman, D., Young, R., Gobet, F., Baxter, G. D. (in press). *Techniques for modeling human performance in synthetic environments: A supplementary review*. Wright-Patterson Air Force Base, OH.

Stylianou, A. C., Madey, G. R., & Smith, R. D. (1992). Selection criteria for expert system shells: A socio-technical framework. *Communications of the ACM*, 35, 30-48.

Ye, L. R., & Johnson, P. E. (1995). The impact of explanation facilities on user acceptance of expert systems advice. *MIS Quarterly*, *19*, 157-172.

Zachary, W., Ryder, J., Ross, L., & Weiland, M. Z. (1992). Intelligent computer-human interaction in real-time, multi-tasking process control and monitoring systems. *Proceedings of the Conference for Human Factors in Design for Manufacturability*, New York.

Zachary, W., Jones, R. M., & Taylor, G. (2002). How to communicate to users what is inside a cognitive model. *Proceedings of the 11th Computer Generated Forces Conference*, Orlando, FL, 375-382.