Cognitive Models of Decision Making Processes for Human-Robot Interaction

Christian Lebiere[1], Florian Jentsch[2], and Scott Ososky[2]

[1] Psychology Department, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh PA 15208
`cl@cmu.edu`
[2] Institute for Simulation & Training, University of Central Florida
4000 Central Florida Blvd, Orlando, FL 32816
`Florian.Jentsch@ucf.edu, sososky@ist.ucf.edu`

Abstract. A fundamental aspect of human-robot interaction is the ability to generate expectations for the decisions of one's teammate(s) in order to coordinate plans of actions. Cognitive models provide a promising approach by allowing both a robot to model a human teammate's decision process as well as by modeling the process through which a human develops expectations regarding its robot partner's actions. We describe a general cognitive model developed using the ACT-R cognitive architecture that can apply to any situation that could be formalized using decision trees expressed in the form of instructions for the model to execute. The model is composed of three general components: instructions on how to perform the task, situational knowledge, and past decision instances. The model is trained using decision instances from a human expert, and its performance is compared to that of the expert.

Keywords: Human-robot interaction, shared mental models, cognitive modeling, cognitive architectures, ACT-R, decision trees.

1 Introduction

A fundamental aspect of human-robot interaction is the ability to generate expectations for the decisions of one's teammate(s) in order to coordinate plans of actions. Shared mental models of the knowledge and decision procedures of human and robotic teammates enable them to act as a team rather than a collection of individuals that have to be explicitly controlled [1]. Mental models include a representation of the current situation, the various entities (humans, robots, even animals) involved, their capacities and limitations, and some decisions of their past decisions and actions. Current mental models are often non-computational descriptions of that information that only provide a qualitative understanding and limited predictive power.

Computational cognitive models provide a promising approach to that problem. Cognitive models can provide a computational link from the shared mental model literature to the domain of robotic control and intelligence. By computationally representing the cognitive processes and representations underlying shared mental

models, they enable them to be leveraged in a number of ways to improve human-robot interaction. First, they provide a quantitative, predictive understanding of human shared mental models and their impact on team performance. Conversely, they also provide a cognitively based computational basis for the implementation of mental models in robots that are similar to those of human teammates. Thus, they foster better teamwork by allowing human and robotic teammates to work in similar ways by representing and simulating the internal processes of the others. Finally, the support the improved design of human-robot interaction tools and protocols by enabling quantitative predictions of their effectiveness.

Mental models can be computationally represented in a number of ways. They can consist of an ontology of concepts and decisions that represent the factors involved in shared mental models. They can take the form of symbolic frameworks such as decisions trees and semantic networks that capture the decision procedures followed by teammates. They can also be represented using statistical frameworks such as Bayesian networks or Markov models that capture the statistical regularities of the environment and decisions made.

Cognitive models provide a direct computational instantiation of the representations and mechanisms involved in shared mental models. They provide an account of situation awareness by mapping it to the representation of the current situation and its evolution in the various memory stores (working, short-term, episodic, long-term semantic) available. They include an account of perceptual-motor limitations such as attentional processes. Finally, the basic cognitive processes themselves combine capabilities such as associative pattern-matching and adaptivity with capacity limitations such as limited working memory, memory decay, and stochasticity.

Cognitive models have been used for a number of purposes related to shared mental models. A methodology called instance-based learning has been developed to learn to control complex systems by observing the actions of another controller and making decisions that generalize from those observations [2]. Perspective-taking in spatial domains such as hide-and-seek or collaborative work in space has been used to infer the knowledge of the current situation on the part of another human or robotic entity [3]. Predicting decisions of other entities has been implemented using a number of cognitive processes including theory of mind recursion [4], imagery-based simulation [5] and memory retrieval [6]. Cognitive models have been used to implement the execution of joint plans of actions between synthetic teammates in virtual and robotic simulations [7].

In this paper, we introduce a joint pursuit task involving shared mental models between human and robotic teammates. We describe a general cognitive model developed using the ACT-R cognitive architecture that can apply to any situation that could be formalized using decision trees expressed in the form of instructions for the model to execute. The model is composed of three general components: instructions on how to perform the task, situational knowledge, and past decision instances. The model is trained using decision instances from a human expert, and its performance is compared to that of the expert. Finally we discuss open issues and further work planned in the development of our cognitive representation of shared mental models.

2   Pursuit Scenario

A foot pursuit scenario was developed that highlights the need for close cooperation and an understanding of the capabilities and vulnerabilities of the various human and robotic teammates involved. A decision structure for the pursuit scenario was drafted based on publicly available information on police foot pursuit procedures, online video, and incident reports. These were then altered to the appropriateness of a robot attempting to make a similar action selection about whether or not to pursue a suspect.

Fig. 1. Representation of the foot pursuit scenario

Eight scenarios were developed representing different situations and resulting decisions. Data related to the scenarios were encoded and organized according to shared mental models held by expert teams, including item categories such as equipment, task, team interaction and team characteristics (Fig. 1). A decision tree was then built using information from police foot pursuit procedures leading to four possible initial decisions: soldier-only pursuit, robot-only pursuit, joint soldier-robot team pursuit, and holding position without pursuit and reporting the incident (Fig. 2).
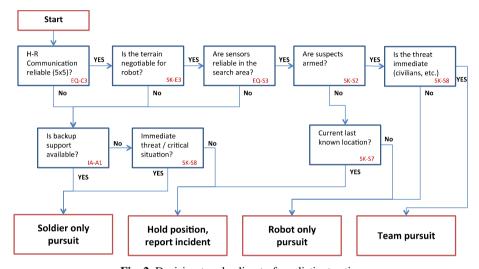
Fig. 2. Decision tree leading to four distinct actions

For each decision, the most critical item was listed, although other factors were usually also considered in the decision. The scenario data provided information about both the items involved in each decision and the correct decision according to a human expert. Finally, the human expert rank-ordered the four possible actions according their suitability for each of the eight scenarios.

3   Cognitive Model

The cognitive model was developed using the ACT-R cognitive architecture [8], [9]. Cognitive architectures are computational representation of invariant cognitive mechanisms specified by unified theories of cognition. ACT-R is a modular architecture, reflecting neural constraints, composed of asynchronous modules coordinated through a central procedural system (Fig. 3). The procedural system is in charge of behavior selection and more generally the synchronization of the flow of information between the other modules. It is implemented as a production system where competing production rules are selected based on their utilities, learning through a reinforcement mechanism from the rewards and costs associated with their actions. The production system conditions are matched against limited-capacity buffers that control the interaction with the other modules by enabling a single command (e.g.,

retrieval of information, focus of visual attention) to be given at a time to a given module, and a single result to be returned (e.g., chunk retrieved from memory, visual item encoded). A declarative memory module holds both short-term information, such as the details of the current situation, as well as long-term knowledge, such as the procedural rules to follow. Access to memory is controlled by an activation calculus that determines the availability of chunks of information according to their history of use such as recency, frequency, and degree of semantic match. Learning mechanisms control both the automatic acquisition of symbolic structures such as production rules and declarative chunks, and the tuning of their subsymbolic parameters (utility and activation) to the structure of the environment. The perceptual-motor modules reflect human factor limitations such as attentional bottlenecks. Individual differences can be represented both in terms of differences in procedural skills and declarative knowledge, as well as in terms of architectural parameters controlling basic cognitive processes such as spreading of activation.

Fig. 3. ACT-R Cognitive Architecture

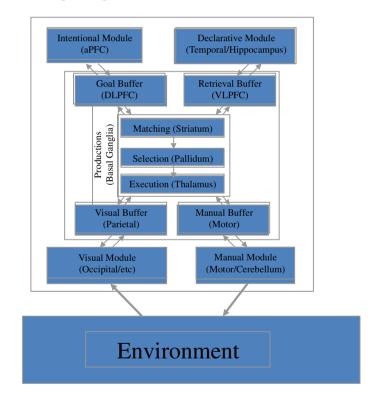As is standard with cognitive modeling practice, we initially considered developing a cognitive model of this specific domain and decision procedure. Instead, we aimed to develop a general cognitive model of shared mental models independent of any domain or decision procedure. Therefore, we generalized our initial cognitive model

to apply beyond the current scenario to any situation that could be formalized using a decision tree procedure. This general ACT-R model takes mental models formalized as decision trees and expresses them in the form of instructions for the model to execute. The declarative knowledge of the model is composed of three general components.

The first part is the general declarative knowledge, encoded as instructions, of how to carry out decisions in a type of situation. Each decision is represented as a set of chunks that represent the situation variables relevant to the decision as well as how to chain multiple decisions together in an overall decision tree. General knowledge on how to carry out the decision, such as heuristic rules, could also be represented but is not currently. This first part can be seen as the core mental model relevant to that situation. At this point we do not differentiate between mental models for different individuals, including teammates and/or opponents, assuming that those models are generally shared between individuals. However, elaborating the representation to allow for mental models of individual decision-making would be quite possible.

The second part of the model is the declarative representation of specific situational knowledge, both the current one as well as past ones. This knowledge is composed of one chunk for each situational variable and its value, for each given situation. It also encodes instances of decisions made in the context of past situations, which are generalized to make decisions in future situations. This part of the model can be described as the situation awareness of the situation. Again, as for the mental model above, this representation of the situation is assumed shared between individuals but could be individualized as well.

The third part of the model is the representation of past instances of decisions. In keeping with the instance-based learning methodology, those decisions consist of a set of the type of decision in the decision tree, a set of values describing the relevant item in a past scenario, and the decision that was taken. Those instances are learned from the human expert annotation of the scenarios provided and represented his individual expertise. It is quite straightforward for models to acquire experience from different experts (or a combination of them) thus allowing for an individualized style of decisions tailored to specific teammates. There is no hardcoded decision logic: each decision depends on matching against past instances involving the subsymbolic (statistical) level of the architecture, with activation processes reflecting factors such recency, frequency and degree of partial matching as well as stochastic factors resulting in probabilistic decisions.

The final part of the model is the procedural logic that controls how the first part of the model, the shared mental model of procedures, is applied to the second, the situation awareness of the situation, using the third, the experience with similar situations, to generate a series of decisions in the current situation. While the other parts of the model can be quite complex and specific and require large numbers of chunks to be represented, this part of the model is quite general, being applicable to any mental model and situation that can be expressed in the current decision tree format, and compact, being represented using only 7 production rules. The production rules learn to retrieve and execute the instructions to interpret the mental model. Each decision is represented as sequence of chained steps to retrieve and encode in working memory

the various pieces of information from the current situation relevant to the decision. The model then makes a decision by retrieving and generalizing past decision instances for similar situations. Each decision leads to another according to the logic encoded in the decision tree procedures, until a decision regarding one of the four possible actions is taken.

4   Results

Fig. 4 illustrates how the model can learn to make decisions without being given explicit logic but instead individual decision instances. Instructions represented the factors involved in the decision (abstracted here as Factor1 and Factor2). Reflecting the probabilistic nature of the decisions, the model was run in Monte Carlo mode to produce a distribution of decision probabilities. The model has inferred that Factor1 (a binary yes/no item) results in an fairly constant probability increase of a "yes" decision while Factor2 (a range item with integer values between zero and five) resulted in a gradual increase in "yes" decisions of about 50% across its range of values.
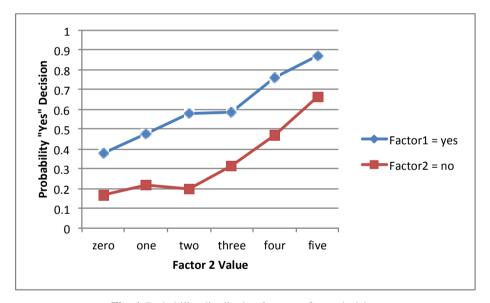
Fig. 4. Probability distribution for a two-factor decision

Since the decisions resulted from past instances rather than hardcoded rules, the model was trained on a subset of the scenarios. We cross-validated the model performance by selecting all subsets of 7 out of the 8 scenarios then tested its generalization to the remaining scenario. Model performance is summarized in Fig. 5, with decision probabilities plotted on a log scale. One can see that for each scenario, one of four possible actions usually (but not always, e.g. the third scenario) stands out as the correct one. Aggregating all decisions across scenarios according to the ranking (1 to 4) provided by the human expert, one can see that the cognitive model has

learned to reproduce the expert's judgment, overwhelmingly (~80%) favoring the #1-ranked decision. The decisions ranked #2 and #3 result in much closer probabilities of about 10% while the #4-ranked decision is almost never taken.

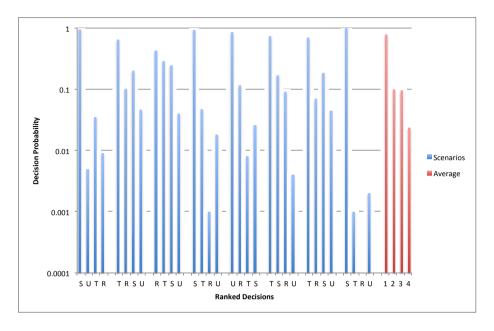**Fig. 5.** Model performance compared to subject judgments. S means soldier pursuit, R means robot pursuit, T means robot and soldier pursue together, and U means abandon pursuit.

5   Discussion

While the existing model is quite general, a number of improvements are possible. Reflecting the general transition from declarative knowledge to procedural skills, the model could proceduralize the individual decision steps from declarative instructions to production rules to replicate the learning curve from novice performance to proficiency and expertise. ACT-R includes a production compilation mechanism that compiles retrievals from declarative memory into production rules to implement this learning process. A feature selection process using the utility learning mechanism could be added to encode and use only a subset of data items for each decision. Data items would only be encoded to the extent that they contributed to making the correct decision. Items that do not contribute to a correct decision would be dropped from the procedure, in a manner similar to that by which experts decide which aspects of a situation warrant their limited attention in high-pressure situations. Similarly, the model could learn shortcuts that combine multiple individual binary decisions into a single, multi-outcome decision, as when experts can recognize a situation and make a rapid, almost instantaneous decision instead of going through the same painstaking process that novices go through. This is possible in our model since the final

decisions involving the four possible actions can be represented in the same instance-based manner as the individual decisions leading to it. Finally, since Monte Carlo simulations are not a cognitively plausible process, it would be desirable to generate rankings of the various actions directly from the activations of the memory retrievals involved in the decision process.

Alternative implementations are also possible. Adopting a Bayesian network formalism (in keeping with the Bayesian underpinnings of the ACT-R activation calculus) would provide an alternative to decision trees in order to enhance generalization in multi-step decisions.

Improvements in the model validation are also possible. Model performance could be validated against human participants data along the entire learning curve, reflecting the learning processes currently existing in the model (strengthening of instructions, accumulation of decision instances) and those described above. As mentioned previously, the model could also be "seeded" with decision instances from different human experts, and its performance compared to that of the individual experts. Integrating the cognitive model in multi-agent simulations would permit to validate it in a dynamic decision-making setting in which a series of decisions is taken rather than a single one. Finally, and most fundamentally, integrating the cognitive model on a robotic platform would allow us to assess its ability to improve human-robot interaction through the computational implementation of shared mental models. This would involve adding to the model the procedural control to perform inferences about the situation awareness knowledge of other entities, and the procedural control to use mental models to plan joint actions involving both teammates and opponents.

References

1. Jentsch, F., Ososky, S., Schuster, D., Fiore, S., Shumaker, R., Lebiere, C., Kurup, U., Oh, J., Stentz, A.: The Importance of Shared Mental Models and Shared Situation Awareness for Transforming Robots from Tools to Teammates. In: Proceedings of the 2012 SPIE Conference, Baltimore, MD (2012)
2. Gonzalez, C., Lerch, F.J., Lebiere, C.: Instance-based learning in dynamic decision making. Cognitive Science 27(4), 591–635 (2003)
3. Trafton, J.G., Cassimatis, N.L., Bugajska, M.D., Brock, D.P., Mintz, F.E., Schultz, A.C.: Enabling effective human-robot interaction using perspective-taking in robots. IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans 35(4), 460–470 (2005)
4. Hiatt, L.M., Trafton, J.G.: A cognitive model of theory of mind. In: Salvucci, D.D., Gunzelmann, G. (eds.) Proceedings of the 10th International Conference on Cognitive Modeling, pp. 91–96. Drexel University, Philadelphia (2010)
5. Wintermute, S.: Using Imagery to Simplify Perceptual Abstraction in Reinforcement Learning Agents. In: Proceedings of the 2010 AAAI Conference on Artificial Intelligence (2010)

6. West, R.L., Lebiere, C.: Simple games as dynamic, coupled systems: Randomness and other emergent properties. Journal of Cognitive Systems Research 1(4), 221–239 (2001)
7. Best, B.J., Lebiere, C.: Cognitive agents interacting in real and virtual worlds. In: Sun, R. (ed.) Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation. Cambridge University Press, NY (2006)
8. Anderson, J.R., Lebiere, C.: The Atomic Components of Thought. Lawrence Erlbaum Associates, Mahwah (1998)
9. Anderson, J.R., Bothell, D., Byrne, M.D., Douglass, S., Lebiere, C., Qin, Y.: An integrated theory of the mind. Psychological Review 111(4), 1036–1060 (2004)