Method of Development of Interactive Agents Grounding the Cognitive Model to the Virtual World

Junya Morita (j-morita@inf.shizuoka.ac.jp)
Kazuma Nagashima (cs16065@s.inf.shizuoka.ac.jp)
Yugo Takeuchi (takeuchi@inf.shizuoka.ac.jp)
Faculty of Informatics, Shizuoka University,
3-5-1 Johoku, Naka-ku, Hamamatsu City, JAPAN

Abstract

Toward the realization of cognitive agents that interact with humans, this research attempts to integrate the cognitive architecture ACT-R and a 3D game engine. We built a hierarchical architecture in which ACT-R and the game engine were connected through a blackboard server, and we constructed a cognitive model for searching the 3D environment. The constructed model reproduced behavioral differences by following parameters of the cognitive model. We also made interesting errors related to the brain-body connection. From these results, it is suggested that the method of cognitive modeling is useful for constructing agents that imitate human behaviors in 3D space.

Keywords: ACT-R; virtual agent; game engine

Introduction

There are several approaches to the ultimate goal of building human or animal-like artificial agents. In the field of human-agent interaction (HAI), researchers have attempted to achieve this goal by focusing on interactions between artificial agents and human users. By considering intelligence as emergent properties of interactions, researchers have developed physical robots and virtual agents that can interact with humans, and they have conducted psychological experiments to examine human reactions to the implemented agents. Throughout these efforts, researchers have tended to emphasize visual appearance (Minato, Shimada, Ishiguro, & Itakura, 2004) or social relationships (Reeves & Nass, 1996) rather than the internal representation and internal processing of agents.

Meanwhile, the method of implementing human nature into internal representations and processing them into artificial agents has been traditionally studied in the community of cognitive modeling, which is a traditional research approach that combines artificial intelligence researches and psychological studies in the field of cognitive science. In this community, cognitive models are assumed to be hypotheses of a human's internal processing, which are represented as a computational system. Unlike other artificial intelligence researches, the study of cognitive modeling focuses on reproducing human errors, biases, and bounded rationality (Simon, 1996) found in psychological studies, which are evaluated by simulation studies reproducing the results of psychological experiments.

Despite dealing with similar topics, not much knowledge has been exchanged between the two communities. For HAI researchers dealing with human response to agents as the main data, deep internal processing might not be of interest. However, in the future, when HAI handles complicated and long-term interaction series more often, the development of agents that include internal processing, as dealt with in cognitive modeling, will be required.

From the above recognition, the authors explored the development method of an interactive agent that involved a cognitive modeling approach. In particular, this paper aims to discuss approaches toward this goal and research topics derived from the developed approach. In the following sections, we first discuss the approach of integrating HAI and cognitive modeling along with previous findings in the related fields. Based on this approach, we then present our system and a preliminarily experiment to discuss its usefulness in HAI studies.

Integrating HAI and Cognitive Modeling

Cognitive Architecture

In the cognitive modeling community, the role of cognitive architectures has become increasingly important. Cognitive architectures are the basis for integrating methods developed in individual studies of cognitive modeling. By accumulating the findings obtained from individual model development, it is thought that the structure of a universal cognitive system can be approached (Newell, 1990). Several cognitive architectures have been developed so far. In the current research, we focused on ACT-R (Anderson, 2007). ACT-R has been developed in the community, where many researchers participate. In addition, psychological and physiological studies have been conducted to associate the modules and parameters of the architecture with the brain structure (Anderson, 2007) and physiological functions (Dancy, Ritter, Berry, & Klein, 2015). Although the original ACT-R is described in Lisp, there are also implementations in multiple programming languages, including Java (Harrison, 2002) and Python (Stewart & West, 2005), making it possible for it to be developed flexibly depending on each individual developer's environment.

Connect to the Virtual World

ACT-R has several modules that are not only related to internal processing, including goal, declarative, and imaginal, but also used for interaction with the external environment, including perception and motors. However, these interactive modules do not include sensors that acquire physical signals or actuators that interact directly with the physical world. In other words, to construct an interactive agent using ACT-R, it is necessary to prepare a separate body to be connected with ACT-R. Regarding this problem, Trafton et al. (2012) implemented ACT-R on a humanoid robot that was able to interact with humans in the real world although its interactions are limited because of hardware limitations.

Considering such implementation difficulties, the current research adopts a virtual agent in a three-dimentional (3D) world instead of physical robot. To build a 3D virtual world, we used a game engine. Many game engines developed in recent years include sophisticated physical engines and body models, and they can build worlds with high reality. Recently, several studies linking these 3D environments and ACT-R have appeared. One study has developed a virtual humanoid robot that determines simple actions, such as walking and rotation, according to its perception of the 3D environment (Somers, 2016), and another study has developed a virtual robot that searches a maze environment in the virtual world while constructing a map of its environment (Smart, Scutt, Sycara, & Shadbolt, 2016). Based on the findings of the previous studies, the current research extends the scope of application while developing a novel architecture that links an ACT-R model with the virtual world.

Integrating Cognitive Architecture and the Virtual World
When connecting ACT-R to the virtual world, we need to solve a problem derived from different time scales of the two systems. In the virtual world, multiple independent events usually proceed in real time. By contrast, the process occurring within ACT-R is sequential. Therefore, for the integration of ACT-R and the virtual world, a framework such as the Subsumption Architecture (Brooks, 1986), which organizes sub-behaviors into hierarchical layers, is required to run processes of different layers in parallel. In other words, the control of body movement in the virtual world occurs in the lower layer, and decision making based on knowledge representation by ACT-R occurs in the upper layer. Both of these layers operate in parallel while communicating at regular intervals. The upper layer decides upon an action based on inference with a knowledge base while inputting the perceptual information acquired in the lower layer. The lower layer receives the decision of the upper layer as a command and transforms it to perform low-level body movement (walking, changing posture, turning around, etc.).

System

Architecture

We implemented a prototype hierarchical system that connects ACT-R (Python ACT-R) and a game engine (Unreal Engine 4) via a blackboard server (Figure 1). The server was implemented in C language, and had slots for storing action commands from agents and slots for storing visual information obtained from the environment. The value of each slot was updated via periodic socket communication from the game engine or ACT-R. With reference to past research (Somers, 2016; Smart et al., 2016), the data format used for communication was unified to JavaScript Object Notation (JSON).

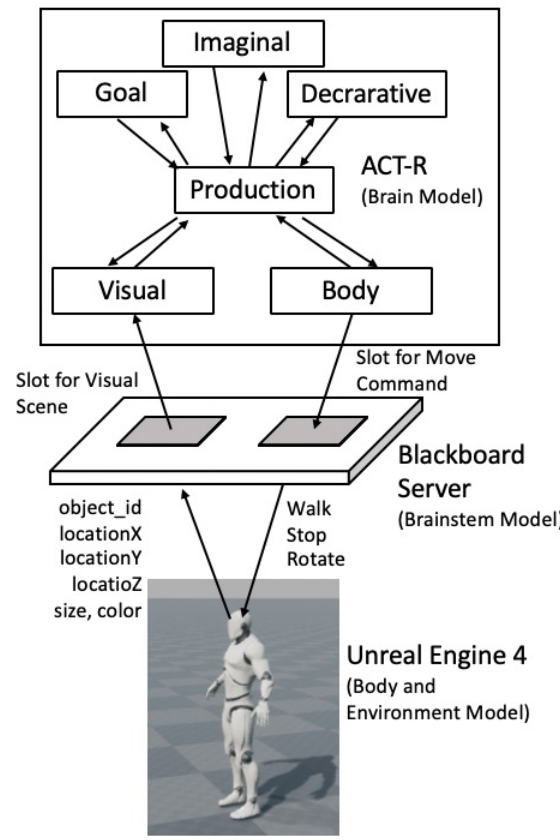

Figure 1: Architecture connecting ACT-R and the virtual world.

According to Anderson (2007), the ACT-R modules correspond to brain regions: the production module to the basal ganglia, the visual module to the visual cortex, the motor module to the motor cortex, the imaginal module to the parietal lob, the goal module to the anterior cingulate cortex, and the declarative module to the prefrontal cortex. Therefore, in this architecture, we assumed that the server corresponds to the brainstem connected to the brain model (ACT-R) with the virtual body, which have several movement patterns. The ACT-R architecture communicates with the server to monitor the state of the body, and to send a command for the next movement pattern, and to interrupt the current movement when necessary.

Task and Model

To test the above architecture, we implemented an agent that performs a simple environment search with the constructed architecture. Figure 2 shows the 3D environment in which the agent is located. A bird's-eye view is shown in the upper left, and a visual perspective of the agent is shown in the lower

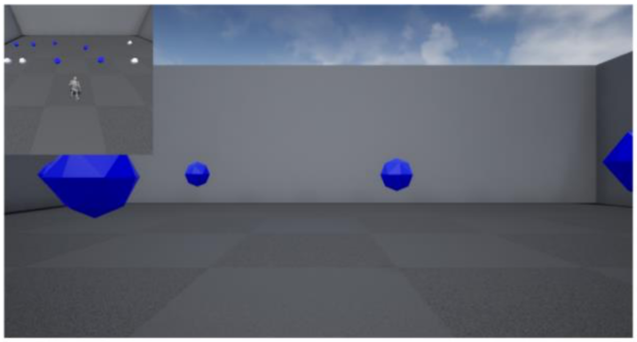

Figure 2: Task environment. The upper left window shows a view from a bird's-eye view camera, and the bottom window shows a view from the agent. The blue-colored objects are in the agent' s field of view. The white-colored objects are out of sight.

right. The task of the agent in this environment is to collect all the blue objects in as short a time as possible. However, with this agent, we did not aim to search for the shortest path connecting the positions of the objects. At each time point, the agent repeated a forward chaining search toward the nearest object.

Figure 3 is a flow chart showing the operation of the agent. Before collecting each object, the agent rotates its body and searches for objects in the environment. When the agent pays attention to one of the objects, it perceives the distance from it. When there are multiple objects in the field of view, one of the objects is selected according to the saliency values set for the object (Stewart & West, 2005). In the current agent, the saliency values were determined by the size of the object projected in the field of view, which corresponds to the distance from the agent. Based on the distance of the object to which its attention is directed, the agent updates the "nearest distance object" in the goal buffer.

At the blue triangle in Figure 3, the rule for searching for objects in the environment (*the searching rule* represented in the right-directed arrow from the triangle) conflicts with the rule for finishing the search (*the finishing rule* represented in the downward arrow from the triangle). Depending on the result of this choice, two types of errors might occur: incorrectly going to the non-nearest objects or continuing the search even after all objects were checked. In ACT-R, the frequency of these errors is controlled by conflict resolution. When the utility (priority) of the searching rule is higher than the utility of the finishing rule, the agent carefully checks the nearest object. Otherwise, the possibility of the other type of error (heading to the non-nearest object) is increased.

Experiment

We considered that one of the benefits of incorporating a cognitive modeling approach to HAI research is representing the individual difference between agents at a behavioral level.

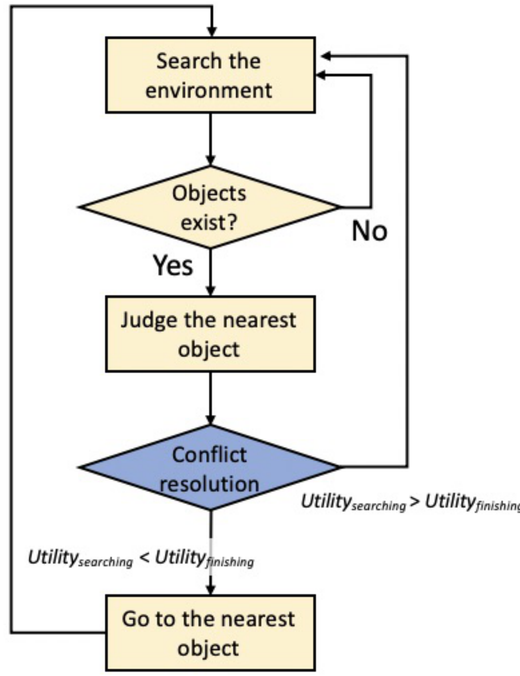

Figure 3: Flowchart of the environment search model.

Recently, in the cognitive modeling community, the exploration of model parameters that represent personal traits is a major topic (Rehling, Lovett, Lebiere, & an B. Demiral, 2004; Anderson, Bothell, Fincham, & Moon, 2016). Using parameters implemented in ACT-R, some researchers have also constructed models of atypical personal traits, such as depression (van Vugt & van der Velde, 2018) and autism (Morita et al., 2017). Utilizing these studies, it is possible to create various types of agent manipulating parameters that can be implemented in the model and architecture. In the case of our model, the agent that has a high utility value for the finishing rule can be regarded as *the reckless agent*, while the agent that has a high utility value for the searching rule can be regarded as *the careful agent*.

To demonstrate the difference between the behaviors of such agents, we conducted a simple experiment in which the utility values of the two rules in Figure 3 were varied. We prepared five conditions of 1:5, 2:4, 3:3, 4:2, and 5:1. The numbers on the left and right indicate the utility values of the searching rule and the finishing rule, respectively. In the simulation, transient noise ($s = 0.5$) was added to each utility value. The agent, whose walking speed was 450 cm/s, searched the environment presented in Figure 4 ten times for each condition. Figure 5 shows the completion time of each condition in box plots. From this figure, we can observe differences between the behaviors of each agent. Compared to the careful agents (the box plots toward the right), the reckless agents (the box plots toward the left) indicated better performance. However, we are not intending to conclude on the superiority of reckless decisions. There is a possibility that this

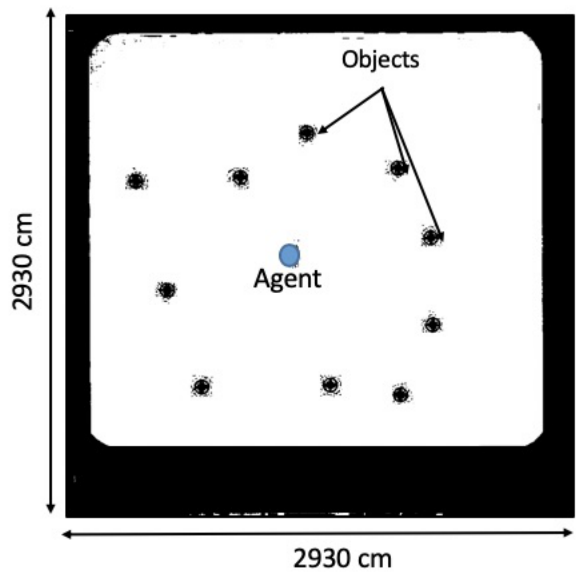

Figure 4: Arrangement of objects in the experiment.

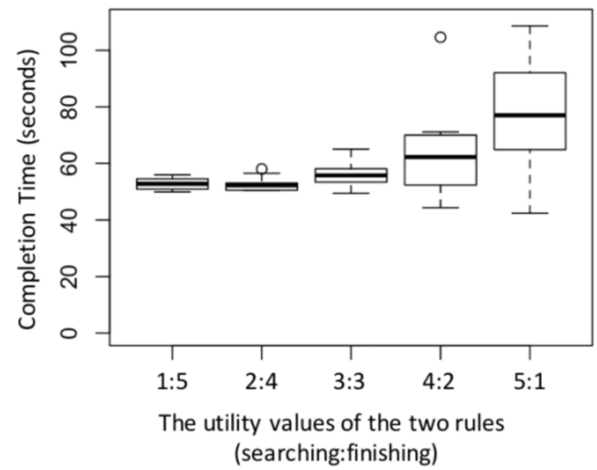

Figure 5: Results of the experiment.

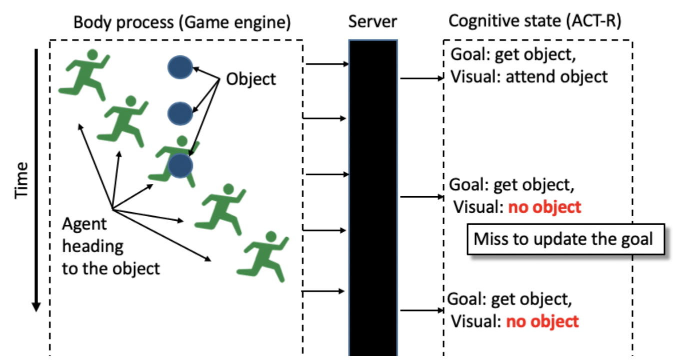

Figure 6: Schematic presentation of microslip error.

result may change depending on the simulation settings (the map or the parameters of the agent such as walking speed). The key point is that by manipulating the parameters of the ACT-R model, we can easily represent a variability of the behavior in the 3D environment.

In addition to the above quantitative result, we found that the qualitative result indicating the potential of our architecture to replicate human-like behavior. In our architecture, the game engine and ACT-R regularly communicate via a blackboard server (Figure 1). During this process, the ACT-R model sometimes overlooked the update from the blackboard server due to mismatches between the communication rate and movement speed of the virtual agent. When such a communication error occurred, the current agent typically failed to be aware of finishing its own behavior keeping searching for the object even though it has already gotten (Figure 6). From an engineering point of view, such an error is regarded as a bug that should be fixed. However, in cognitive modeling or when building a human-like agent, we should evaluate such agent behaviors based on their correspondence to human behaviors. With regard to this error, we can find similar errors in the literature, in ecological psychology, called microslips in which an erroneous action is initiated but aborted (Reed & Schoenherr, 1992). The similarity between the human error pointed out in this psychological study and our agent shows that there is a certain validity in the structure of this architecture.

Conclusion

In this research, we constructed a mechanism to integrate cognitive modeling with the 3D virtual world. This was not the first time that it has been attempted to connect a game engine and ACT-R cognitive model. However, our architecture was different from the previous research in using a blackboard server (Figure 1) and not connecting the ACT-R and game engine with peer-to-peer. Due to this mechanism, novel agent behaviors, such as the microslip mentioned at the end of the previous section, emerged, and these were caused by ACT-R and the game engine operating in parallel.

Thus, the architecture constructed in this research may lead to the modeling of cognitive processes that have been overlooked in previous research. Many of the conventional cognitive models do not have a body and deal with the problems of a simple system closed in the brain. By giving a body in the virtual space to the cognitive model, there is a possibility of simulating important phenomena related to the interaction between the body and brain. In addition, the architecture of this research also has advantages in terms of being extended to a multi-agent environment. Considering this advantage, in the future, we plan to model interactions between groups with multiple embodied agents in the virtual world.

In addition, visualization of the virtual world using game engines has the advantage of making it possible to interact with agents operating using the ACT-R cognitive model and human users. The advantage of such interactive agent development with ACT-R is systematic diversion of research knowledge accumulated in cognitive modeling research. Furthermore, visualizing the behavior of interpersonal agents in the virtual world may also lead to a new methodology of validating hypotheses behind the implemented internal process in a cognitive model. In this way, the integrated approach that this research aimed for may lead to new HAI and cognitive modeling research methods.

References

Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* New York: Oxford University Press.

Anderson, J. R., Bothell, D., Fincham, J. M., & Moon, J. (2016). The sequential structure of brain activation predicts skill. *Neuropsychologia*, *81*, 94-106.

Brooks, R. A. (1986). *Planning is just a way of avoiding figuring out what to do next* (Tech. Rep.). MIT Artificial Intelligence Laboratory.

Dancy, C. L., Ritter, F. E., Berry, K. A., & Klein, L. C. (2015). Using a cognitive architecture with a physiological substrate to represent effects of a psychological stressor on cognition. *Computational and Mathematical Organization Theory*, *21*(1), 90–114.

Harrison, A. (2002). jACT-R: Java ACT-R. In *Proceedings of the 8th Annual ACT-R Workshop.*

Minato, T., Shimada, M., Ishiguro, H., & Itakura, S. (2004). Development of an android robot for studying human-robot interaction. In B. Orchard, C. Yang, & M. Ali (Eds.), *Lecture Notes in Computer Science* (Vol. 3029, p. 424-434). Springer.

Morita, J., Konno, T., Okuda, J., Samejima, K., Li, G., Fujiwara, M., & Hashimoto, T. (2017). Implicit memory processing in the formation of a shared communication system. In *Proceedings of the 15th International Conference on Cognitive Modeling (ICCM 2017)* (p. 19-24).

Newell, A. (1990). *Unified Theories of Cognition*. Harvard University Press.

Reed, E. S., & Schoenherr, D. (1992). *The neuropathology of everyday life: On the nature and significance of microslips in everyday artivities'.*

Reeves, B., & Nass, C. (1996). *The Media Equation: How People Treat Computers, Television, and New Media Like Real People and Places*. Cambridge University Press.

Rehling, J., Lovett, M., Lebiere, C., & an B. Demiral, L. M. R. (2004). Modeling complex tasks: An individual difference approach. In *In Proceedings of the 26th Annual Conference of the Cognitive Science Society* (p. 1137-1142).

Simon, H. (1996). *The Sciences of the Artificial* (Third edition ed.). Cambridge, MA: MIT Press.

Smart, P. R., Scutt, T., Sycara, K., & Shadbolt, M. R. (2016). Integrating ACT-R cognitive models with the unity game engine. In J. O. Turner, M. Nixon, U. Bernardet, & S. DiPaola (Eds.), *Integrating Cognitive Architectures into Virtual Character Design.* Hershey, Pennsylvania, USA: IGI Global.

Somers, S. (2016). ACT-R 3D: A 3d simulation environment for python act-r. In *Proceedings of the 14th International Conference on Cognitive Modeling (ICCM 2016)* (p. 107-112).

Stewart, T. C., & West, R. L. (2005). Python ACT-R: A new implementation and a new syntax. In *Proceedings of 12th Annual ACT-R Workshop.*

Trafton, J. G., Hiatt, L. M., Harrison, A. M., II, F. P. T., Khemlani, S. S., & Schultz, A. C. (2012). ACT-R/E: An embodied cognitive architecture for human-robot interaction. *Journal of Human-Robot Interaction*, *1*(1), 78-95.

van Vugt, M. K., & van der Velde, M. (2018). How does rumination impact cognition? a first mechanistic model. *Topics in Cognitive Science*, *10*(1), 175-191.