Chen, W. et al. (Eds.) (2016). Workshop Proceedings of the 24th International Conference on Computers in Education. India: Asia-Pacific Society for Computers in Education

Can Students Build Cognitive Models That Reflect Their Own Cognitive Information Processing? Results of Preliminary Class Practice

Kazuhisa MIWA[a*] and Hitoshi TERAI[b]
[a]*Graduate school of information science, Nagoya University, Japan*
[b]*Faculty of Human-oriented science, Kindai University, Japan*
*miwa@is.nagoya-u.ac.jp

Abstract: We have developed a learning environment to enable students to build rule-based cognitive models. Participants were required to build cognitive models that solve a cryptarithmetic task. To solve this task, multiple types of procedural knowledge were used during the solution processes; participants were required to externalize their knowledge when constructing a model. Twenty-two participants successfully constructed sophisticated models that were able to solve the given task in approximately 21 steps. We compared the participants' and their models' problem-solving behaviors. As a result, only 9 of the 22 (41%) models trance the participants' problem-solving paths. This implies that it was relatively difficult for the participants to construct a model that reflects their own cognitive information processing.

Keywords: Cognitive models; Procedural knowledge, Production system

1. Research question

Students encounter challenges while posing problems. Preceding studies have reported that when students are required to pose problems, they merely replicate a familiar example problem, not employing effective styles of problem posing (Christou, et al., 2005; Kojima, Miwa, & Matsui, 2010; 2015). To enhance students' problem-posing activities, we must investigate the processes that underlie problem-posing activities.

When we pose an arithmetic problem, we must possess the procedural knowledge required to solve that problem. Our previous studies have confirmed that learning activities of creating computational cognitive models are effective for understanding such procedural knowledge (Miwa, et al., 2015). Especially, in such activities, it seems important to externalize their knowledge by constructing cognitive models that reflect their own cognitive information processing.

In cooperation with the experimental approach, the model-based approach is a primary methodology in cognitive science (Fum, Missier, &Stocco, 2007). Cognitive scientists have used computational models as research tools to understand the human mind. The authors have examined the functions of cognitive modeling as a learning tool and proposed the "learning by creating cognitive model" paradigm (Miwa, et al., 2009; Miwa, et al., 2014a). Previous studies have confirmed that creating cognitive models improves theory-based thinking (Saito, et al., 2013; Miwa, et al., 2014b) and active construction of mental models (Miwa, et al., 2015).

In our previous class practices, participants were required to construct cognitive models that solve a cryptarithmetic task. The results showed that nearly two-thirds of the students successfully constructed sophisticated models that reached the solution within a considerably small number of steps (Miwa, et al., 2016). This implies that the participants successfully externalized procedural knowledge to perform relatively complex arithmetic information processing. However, it is unclear whether such models actually reflect their own cognitive information processing.

Our research question in this study is whether students can construct cognitive models that reflect their own cognitive information processing. We conducted a cognitive science class wherein participants were required to build computational models that behave similarly as the students themselves behave. We report a preliminary analysis in this paper.

6. Task

The task used in our study is a cryptarithmetic task (Newell & Simon, 1972; Miwa, et al., 2009). The following is an example problem used in our class practice. The problem is to assign digits (0, 1, 2, ..., and 9) to the letters (A, B, C, D, E, F, G, H, I, and J), so that when the letters are replaced by their corresponding digits, the sum is satisfied. Here information F = 6 is given in the initial statement of the problem. The reason why this problem was used is because to solve this problem, various types of procedural knowledge that will be described below in detail are expected to be used.

```
  IGEAF F = 6
+DBJAD
CIHEGH
```

There are two solutions in this task. Specifically, I and E are undecided (2 or 4).

This problem is simple; however, the cognitive information processing for its solution is relatively complex. In fact, multiple types of procedural knowledge are used during the solution processes. The following describes some examples.

Numeral processing: If a column is $x + y = z$, and both x and y are known, then we can infer z by summing x and y. For example, in the rightmost column, when we know F = 6 and D = 9, 5 is assigned to letter H by applying this procedure.

Specific numeral processing: If a column is $x + y = x$, then we can infer that y = 0 or 9. For example, in the fifth column, we can obtain D = 0 or 9 independently without any other information. In this case, we can determine that D = 9 by the following process because C should be 1, meaning that a carry is sent to the left-side column.

Parity processing: If a column is $x + x = y$ and we have a carry from the right column, then we can infer that y is an odd number. For example, in the second column, we obtained a carry by the inference in the first (i.e., rightmost) column; therefore, we conclude that G is an odd number.

Inequality processing: If a column is $x + x = y$, and no carry is sent to the left column, then we can infer that x is less than 5. For example, in the second column, when we know there is no carry to the left column; thus, A is less than 5.

University students easily understand such procedural knowledge sets if they are given; however, they may face challenges finding the knowledge by themselves and externalizing it while solving the problem.

7. Learning System

We have developed a learning environment to enable students to construct rule-based cognitive models. The system consists of two modules, i.e., a knowledge editor and a problem-solving simulator.

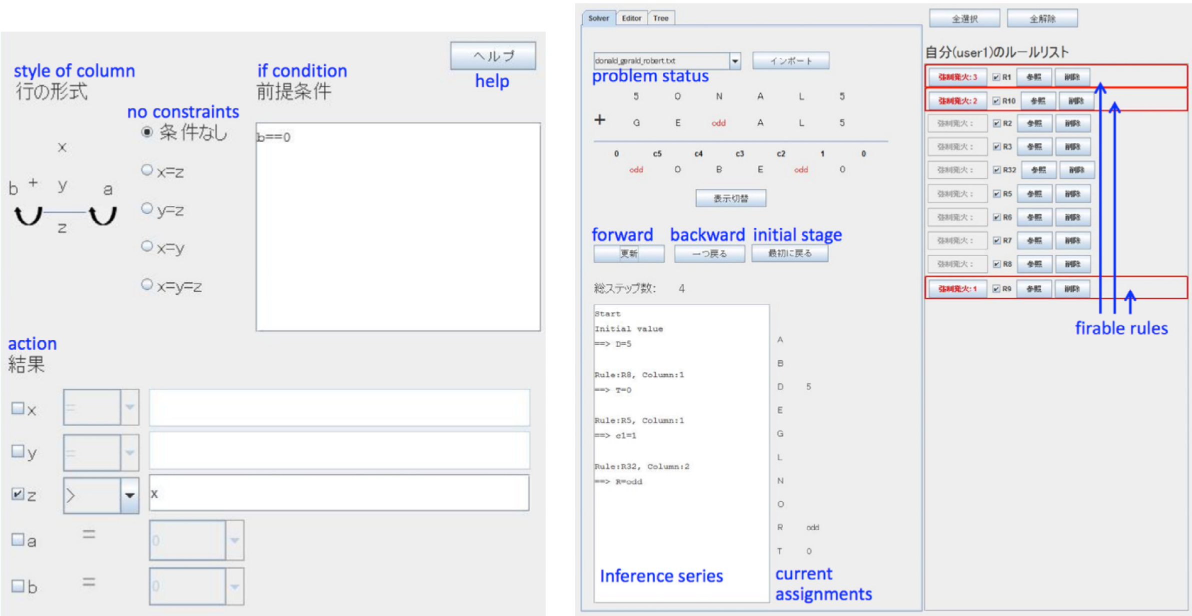
7.1 Knowledge editor

First, students externalize a set of procedural knowledge, i.e., describing rules, to solve cryptarithmetic tasks using the knowledge editor.

Figure 1 (a) shows an example screenshot of the knowledge editor wherein the rule of inequality processing is described, i.e., if a column is $x + y = z$ and no carry is sent to the left column (b == 0 in the figure), then we can infer that z is greater than x.

7.2 Problem-solving simulator

The problem-solving simulator is mounted on the learning system. The problem solver that simulates behavior has the potential to perform an exhaustive search for the assignment of digits to letters. Specifically, it selects one of the letters that have not been determined and systematically assigns each digit to a letter. If a contradiction is found in the inference process, another assignment is tested. If the problem solver has no procedural knowledge, it is impossible to derive the solution because the problem space spreads exhaustively. Students must give the problem solver adequate procedural knowledge using the knowledge editor.



2. Knowledge Editor (b) Problem-solving simulator

Figure 1 Example screenshots of the learning system

Figure 1 (b) shows an example screenshot of the problem-solving simulator, which presents a problem status (the assignment status of digits to letters) and an inference status (a step-by-step series for information processing). A list of rules installed for the problem solver is presented on the right-hand side of the window. Rules that can fire at a specific problem-solving step are marked by bold red lines. In this case, three rules are available. The conflict resolution mechanism is simple, and the most specific rule that provides the most specific inference result has priority for firing. Students can test any rule by forcibly firing it and confirming the resulting inferences. Moreover, students can modify the model very easily. For example, if we uncheck items in the list, students can simulate the behavior of the problem solver with that knowledge excluded.

The system also presents the problem solver's behavior, represented as a search tree of problem-solving processes. Students can confirm inference steps one by one by forwarding the inference by clicking the inference button. At any point in the problem-solving process, students can install, delete, or revise knowledge using the editor and restart the inference from the given problem-solving point.

8. Class Practice

The class practice was performed as part of a cognitive science class in the first author’s university. Participants included 25 undergraduates from Nagoya University. In the initial week, the participants spent one hour learning how to manage the knowledge editor and operate the problem-solving simulator. Specifically, participants were given an example problem: MEST + BADE = MASER. They then installed seven pieces of procedural knowledge to solve the given problem with a tutor's guidance, and they simulated behavior at each stage of the construction process.

In the second week, in a 70-minute training phase, the participants were given a training problem: DONALD + GERALD = ROBERT. By themselves, they were required to find a procedural knowledge set for the solution, install it in the problem solver with the knowledge editor, and then construct a model. In the third week, the participants were given the target problem: IGEAF+DBJAD = CIHEGH. They were required to construct a model for its solution. After model construction, they were required to solve the same problem by hand by writing their solution processes on an experimentation sheet. Both the model and participant solution processes were analyzed.

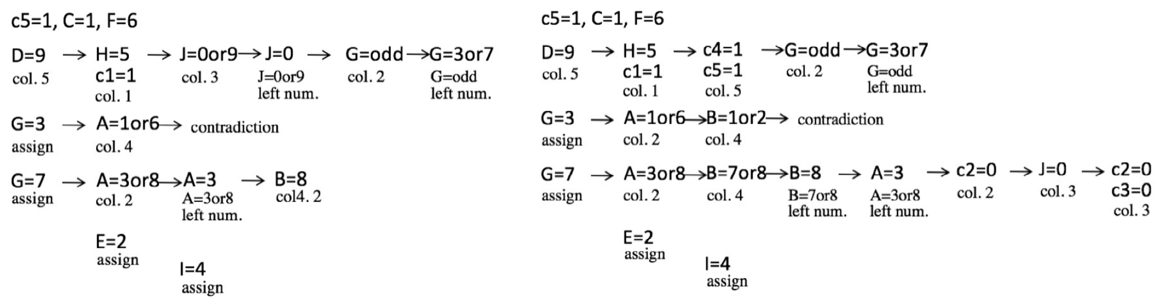
9. Results

One participant could not construct a complete model that reached the solution within 100 problem-solving steps. Two other participants' problem-solving paths were not clearly identified due to insufficient descriptions of the problem-solving paths written on the examination sheets. We excluded these three participants from our analysis.

The average number of problem-solving steps for the other 22 participants was 20.9 steps. We analyzed the participants' problem-solving paths. All participants initially processed the fifth column (I + D = I) and drew the decisive information D = 9. Then, the participants processed the other columns, coordinated multiple pieces of information obtained through the preceding problem-solving processes, and focused on a specific letter to which a possibility of assignments of numerals was limited for the following trial-and-error search. Specifically, for the example in Figure 2 (a), first, G = odd was determined by processing the second column where the same letters (A and A) were summed and a carry was received from the right-side column. Then, based on the information that G = odd, a limited possibility of assignments (i.e., G = 3 or 7) was obtained because other odd numbers (1, 5, and 9) had already been assigned to other letters (i.e., C, H, and D, respectively). Then, the participant began to examine G = 3 in a trial-and-error search.

Figure 2 (b) shows a problem-solving path of this participant's model. The path is similar to that of the participant shown in Figure 2 (a). Initially, the model drew G = 3 or 7, found that G = 3 was impossible, and reached the solution by examining another assignment, i.e., G = 7.

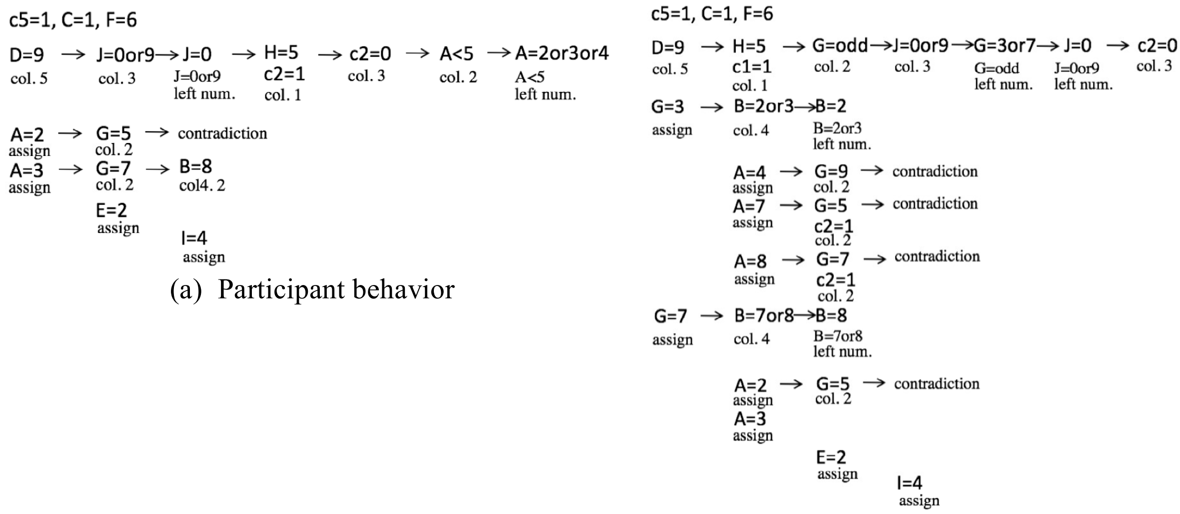
We focused on the overall patterns of the problem-solving paths determined by trial-and-error search driven by an examined letter, such as G in Figures 2 (a) and (b). In 9 of the 22 cases, the patterns of the participants' behaviors were similar to those of the models; however, the 13 other cases were not similar.



(a) Participant behavior (b) Model behavior

Figure 2 Comparison of human and model problem-solving behaviors; example case of similar processes

Figures 3 (a) and (b) show an example case where the participant and model behaviors did not match. In Figure 3 (a), the participant inferred that A = 2, 3, or 4 by combining A < 5 that had been obtained by processing the second column with the information that no carry was sent to the left-side column and the information that 1 was already assigned to C. Based on this information, the participant examined each assignment to the letter A. Figure 3 (b) shows a problem-solving path through which the model that the participant constructed had run. The model initially inferred that G = 3 or 7, guiding a subsequent trial-and-error search that differed from the participant's path. The model did not infer information related to letter A and did not focus on letter A for the initial trial-and-error search, thereby changing the problem-solving path.



(a) Participant behavior

(b) Model behavior

Figure 3 Comparison of human and model problem solving behaviors; example case of different processes

10. Conclusions

We analyzed 22 participants who successfully constructed sophisticated models that can solve the given task in approximately 21 steps. However, only 9 of the 22 (41%) models trance the participants' problem-solving paths. This implies that it was relatively difficult for the participants to construct a model that reflects their own cognitive information processing. First, this problem comes from the participants' programming abilities. It appears that some participants could not implement appropriate rules even though they noticed their own procedural knowledge. This implies that our next step is to improve the learning environment developed in the current study. Another reason is that model construction that reflects each participant's cognitive processing was not emphasized in the current class practice. Some participants attempted to construct high-performance models that solve the task as quickly as possible or general models that can solve a variety of tasks. We believe this can be improved based on instructor suggestion.

Acknowledgement

This work was supported by JSPS KAKENHI Grant Number 15H02927.

References

Christou, C., Mousoulides, N., Pittalis, M., Pitta-Pantazi, D., Sriraman, B. (2005). An empirical taxonomy of problem posing processes. International Review on Mathematical Education. 37, 149–58.

Fum, D., Missier, F.D., Stocco, A. (2007). The cognitive modeling of human behavior: Why a model is (sometimes) better than 10,000 words. Cognitive Systems Research, 8, 135-142.

Kojima, K., Miwa, K., & Matsui, T. (2010). An experimental study on support for leaning of problem posing as a production task. Transactions of Japanese Society for Information and Systems in Education, 27, 302–15.
Kojima, K., Miwa, K., & Matsui, T. (2015). Experimental Study of Learning Support through Examples in Mathematical Problem Posing. Research and Practice in Technology Enhanced Learning. 10:1 doi:10.1007/s41039-015-0001-5
Miwa, K., Kanzaki, N., Terai, H., Kojima, K., Nakaike, R., Morita, J., Saito, H. (2015). Learning mental models on human cognitive processing by creating cognitive models. Proceedings of AIED 2015, LNCS, 9112, 287-296.
Miwa, K., Morita, J., Nakaike, R., Terai, H. (2014a). Learning through intermediate problems in creating cognitive models. Interactive Learning Environments 22, 326-350.
Miwa, K., Morita, J., Terai, H., Kanzaki, N., Kojima, K., Nakaike, R., Saito, H. (2014b). Use of a cognitive simulator to enhance students' mental simulation activities. Proceedings of ITS 2014, LNCS, 8474, 398-403.
Miwa, K., Nakaike, R., Morita, J., Terai, H. (2009). Development of production system for anywhere and class practice. Proceedings of the 14th International Conference of Artificial Intelligence in Education, 91-99.
Miwa, K. (2008). A cognitive simulator for learning the nature of human problem solving. Journal of Japanese Society for Artificial Intelligence, 23, 374-383.
Miwa, K., Terai, H., Shibayama, K. (2016). Understanding Procedural Knowledge for Solving Arithmetic Task by Externalization. Proceedings of ITS 2016, LNCS, 9684, 3-12
Newell, A., Simon, H. A. (1972). Human problem solving. Prentice-Hall, Englewood Cliffs, NJ.
Saito, H., Miwa, K., Kanzaki, N., Terai, H., Kojima, K., Nakaike, R., Morita, J. (2013). Educational practice for interpretation of experimental data based on a theory. Proceedings of 21th international conference on computers in education, 234-239