Using Human Memory Structures to Model Knowledge within Algebra Virtual Laboratory

Mehdi Najjar and André Mayers
Department of Computer Science, University of Sherbrooke
2500 Bld. de l'Université, Sherbrooke, Qc J1K 2R1 Canada
{mohamed.mehdi.najjar, andre.mayers}@usherbrooke.ca

Abstract

In this paper we describe our knowledge representation model embodied within an algebra virtual laboratory dedicated to boolean reduction problem solving. The model is inspired by psychology theories that explain the human cognitive activity in terms of memory subsystems and their processes. During the problem solving, complex knowledge entities, built from primitive units of knowledge, are dynamically combined to represent the learner behavior. This has the advantage to offer a closely fine prediction of the mastery degree and the acquisition level of each element of the taught knowledge.

1. Introduction

Although the concept of applying computer software for educational purposes dates back some decades, using virtual learning environments for teaching and training is a field of increasing interest. During the last years, various attempts to create highly interactive virtual laboratories (VLs) have been made [7,9,12,16], engendering a large amount of enthusiasm in the educational community. Exploiting the multimedia features and the web advantages, VLs permit the learner to experience, through exploration, the nature of a wide variety of domains. Nevertheless, understanding how humans learn and how knowledge is structured and handled during the learning process is important if we are to develop VLs that include tutorial strategies capable of dealing with complex domains and learners with various degrees of knowledge acquisition [1]. Undoubtedly, modelling this knowledge, acquired in a learning context, is a realisation whose takes up a real challenge. Some representational knowledge models [3,11,15], which attempt to accomplish this task, showed that their elaborate structures, inspired from psychological based approaches, can offer more realistic and efficient representations. This leads us to deduce that it is certainly very beneficial to integrate the knowledge psychological research has accumulated on understanding the cognitive mechanisms of human learning and all of the positive results obtained by computational modelling theories in artificial intelligence.

This paper describes our approach for representing the knowledge taught and the one acquired by learners through VL interactions. We propose representation structures inspired by cognitive theories that explain the human cognitive activity in terms of memory subsystems and their processes. The remainder of the article is organised as follows : First, we present our algebra VL for the problem solving of boolean reduction. Next, the representation structures of the knowledge incorporated within the VL are described. This is followed by some original aspects of our approach. Finally, we announce future work.

2. The virtual lab environment

The VL is realised according to an object-oriented design, implemented in Java. It demonstrates a problem solving organisation that attempts to model the learner cognitive activity stored during the task accomplishment. Our subject-matter domain is the algebraic boolean expressions and their simplification by means of reduction rules, which are generally taught to undergraduate students on first cycle of higher education. In our VL, preliminary notions, definitions and explanations constitute a necessary knowledge background (available through sections exploration via clicking buttons) to approach the boolean reduction problem.

In the preliminary notions section, different boolean simplification rules are stated. By choosing particular rule in a combo-box menu, for example, the De-Morgan rule applied to a conjunction of two proposals (~(p & q) = (~p | ~q)), the latter is posted with a brief formal definition. In the explanation section, hints and thorough explanations on the boolean reduction rules suitable for usage are provided. In the examples section, examples are given. Those are generated randomly with variable degree of difficulty (from 1 to 5) chosen by the learner. For example Figure 1 shows a complexity (fixed by the learner) level 5 example provided by the tutor.

0-7803-8625-6/04/$20.00 ©2004 IEEE

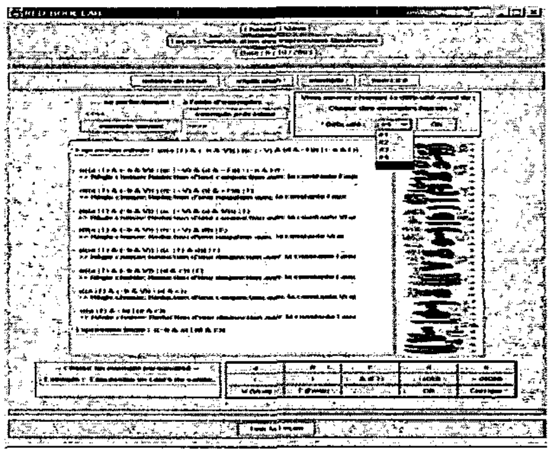

Figure 1. A level 5 Boolean expression and the corresponding problem solving steps.

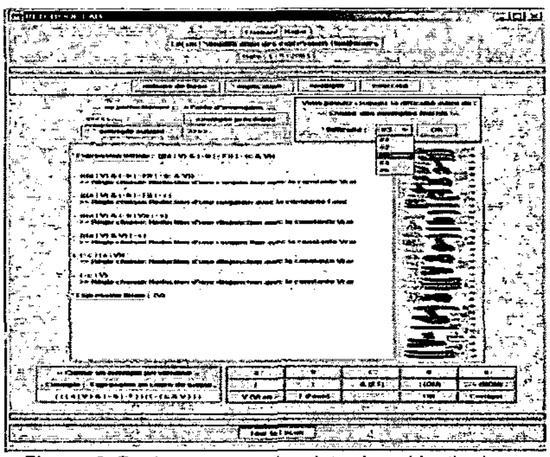

Figure 2. Boolean expression introduced by the learner and the problem solving steps given by the system.

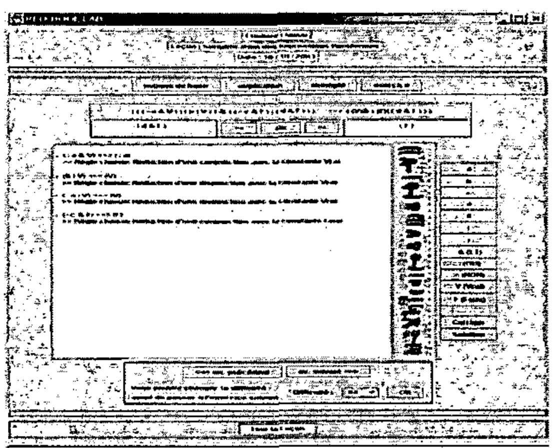

Figure 3. A level 4 expression under reduction process, in the exercise section.

Students can also enter, by means of a visual keyboard, any boolean expression they want and ask the system to solve it. Figure 2 illustrates a complexity level 3 example proposed by the learner. The problem solving steps and the applied rules are shown on a blackboard.

Generated and/or personalised examples show optimal solutions to simplify boolean expressions and are provided to guide learner during the problem solving, which begins by clicking on the exercise button. Exercises, also with variable complexity levels, give opportunities to learner to practice tasks. Via the visual keyboard, students reduce an initial boolean expression (generated randomly) by choosing suitable simplification rules to apply in the order they want. For example, Figure 3 shows a difficulty level 4 expression under simplification process by a learner. The initial expression "(((~a&V) | (b|V)) & ((~c&F) | (d&F)))" was reduced to "((V) & ((F) | (d&F)))". Other reduction rules must be applied from the last step to obtain a final reduced expression. Here, a student (Steve) was using the conjunction rule of a proposal with the "F" truth constant ($F \& p \Rightarrow F$, where p is a proposal) to transform sub-expression "(d&F)" into "(F)". Diverse tutorial strategies are explored. Actually, in the case of erroneous rule choice (or application) on any of the sub-expressions, whose simplification leads from the initial given expression to the final simplified one, the system notifies the learner and shows her/him correct response.

As we believe that students should not only learn theoretical contents and concepts of the domain but also how to handle their knowledge and the related skills in a practical world, our design idea enables students to explore the environment in dynamic interaction with the context as if they were really doing it in the classroom with a teacher monitoring. Because the VL is designed with the knowledge linked through buttons, students are able to determine their own learning paths through the materials which can be reviewed in any desired sequence. This provides learners a flexible learning environment for a non-guided exploration and assisted problem solving. Indeed, students can access all components – the domain knowledge, and problem solving activities – in non-linear ways. They can seek information in different manners. For example, orienting themselves to the case study by retrieving detailed case information, re-visiting previously explored components for additional ideas or reviewing prior knowledge and explanations.

3. The knowledge representation structures

If we have the ambition to endow an artificial system with competence in education and teaching, it is not possible to be unaware of all that concerns the human training, cognition and memory. Rather than being a simple hardware device of data storage (as in the computer's case), the principal characteristic of the human memory is carrying out categorisation, generalisation and abstraction processes [5].

Basically, It has been argued that knowledge is encoded in various memory subsystems not according to their contents but according to the way in which these contents are handled and used, making the memory a large set of complex processes and modules in continual interactions [4]. Several cognitive psychology approaches divide these subsystems in three main sections. Each one of them presents particular type of knowledge such as, semantic [10], procedural [2] and episodic knowledge [14]. To structure and represent the knowledge, handled and used in our VL, we have been inspired by these cognitive theories, which attempt to model the human process of knowledge acquisition.

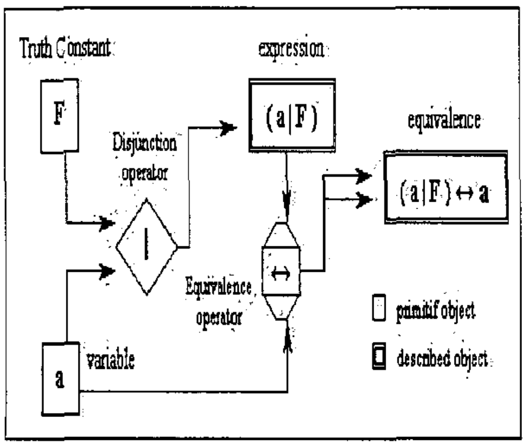

Figure 4. Semantic components of an expressions equivalence.

3.1. Semantic knowledge structuring

Our model regards semantic knowledge as concepts taken in a broad sense. Thus, they can be any category of objects, relations or functions. Moreover, we subdivide concepts in two categories : primitive concepts and described concepts. The first is defined as a syntactically non-split representation; i.e., primitive concept representation can not be divided into parts. For example, in Figure 4, which shows a boolean reduction example (with a difficulty level fixed to 1), the symbol "a" is a non-split representation of the corresponding proposal. On the other hand, we define described concept as a syntactically decomposable representation. For example, the expression "a|F" is a decomposable representation of proposal "a". It represents a disjunction between proposition "a" and the truth constant "F" (False), two primitive concepts. The symbol "|" represents the disjunction logic operator (OR), and is a primitive concept. In this way, the semantic of a described concept is given by the semantics of its components and their relations or functions (which take those components as arguments to create the described concept). For example, as shown in the diagram illustrated by Figure 4, in the expression "(a|F) ↔ a", symbols "a" and "F" are associated to primitive objects (proposal and truth constant), the symbol "|" is associated to primitive function (the disjunction) and the symbol "↔" is associated to primitive relation (the equivalence). Finally, "(a|F) ↔ a" is a described object having three components: "a|F ", "↔" and "a" and it represents an equivalence between two expressions.

3.2. Procedural knowledge structuring

In opposition to semantic knowledge, which can be expressed explicitly, procedural knowledge is inferred by a succession of actions achieved automatically – following internal and/or external stimuli perception – to reach desirable states [2]. A procedure can be seen as a mean of achieving a goal to satisfy a need, without using the attention resources. For example, procedural knowledge allow us to add automatically "25" and "13" (if our goal is to find the corresponding sum), without being obliged to recall the algorithm explicitly. i.e., making the sum of the units, the one of the tens and twinning the two preceding sums. During the boolean reduction process, substituting automatically "~V" by "F", making abstraction to the explicit call of the truth constant negation rule (~V ⇔ F, where "V" ≡ "TRUE"), can be seen as procedural knowledge which was acquired by the repetitive doing.

In our approach, we subdivide procedures in two main categories : primitive procedures and complex procedures. Executions of the first are seen as atomic actions. Those of the last can be done by sequences of actions, which satisfy scripts of goals. Each one of those actions results from a primitive procedure execution; and each one of those goals is perceived as an intention of the cognitive system.

3.3. Episodic knowledge structuring

Episodic memory retains details about our experiences and preserves temporal relations allowing the reconstruction of previously experienced events as well as the time and context in which they took place. In our approach, the episode representation is based on particular generic knowledge (goals) instantiation, retrieved from semantic memory. Episodic knowledge is organised according to goals. Each episode specifies a goal that translates the student interest and gives a sense to the underlying events. These events are sub-episodes that correspond to subgoals realisation. The latter are specified by the procedure used to achieve the main goal. Thus, executions of procedures are encoded in episodic memory and each goal realisation is encoded in an episode. In this way, the learner episodic memory stores all facts during the training activities. Note that episode, seen as specific form of knowledge, has been extensively used in various approaches in a wide variety of domains; such as, modelling cognitive mechanisms of analogy-making [8], artificial intelligence planning [6], knowledge modelling and learner diagnosis within intelligent tutoring systems [15] and neuro-computing [13].

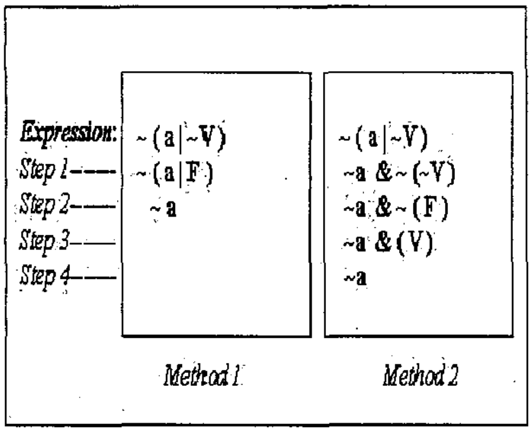

Figure 5. Two solving methods.

In our model, the learner cognitive activity prediction is not determined systematically, in a static way, starting from her/his main goal. Generally, the complex procedure "P", which achieves a given goal "G", determines number and order of "G" subgoals, whose each one can be achieved, in turn, by a procedure (called, in this case, a "P" sub-procedure). The choice of "P" depends of the learner practices and preferences when s/he achieves a task. This means that goal realisation can be made in various ways, by various scenarios of procedures execution sequences. Therefore, number and chronological order of "G" subgoals are not predefined. For example, when the learner goal is reducing "~(a | ~V)", satisfying this goal amounts to find the simplest expression which is equivalent to the initial one. Figure 5 shows two methods leading to the desirable final state. Here, steps correspond to transitions realisable by means of primitive procedures, whose each one is applied to satisfy a subgoal and handles primitive and/or described concepts (propositions and truth constants). In this case, procedural knowledge allowing to achieve the goal *"reduce ~(a | ~V)"*, is a complex procedure giving rise to two subgoals, according to method 1, and four subgoals, with method 2. If the expression is complex (for example, the difficulty level 3 expression "(((a|V) & (~b|~F)) | ~(c&V))"), there can be various ways to reduce it. Therefore, number and order of applied procedures – and the concepts they handle – depend on the selected complex main procedure. This last constraint implies that number and order of subgoals, translating the learner interests, are not determined statically in advance.

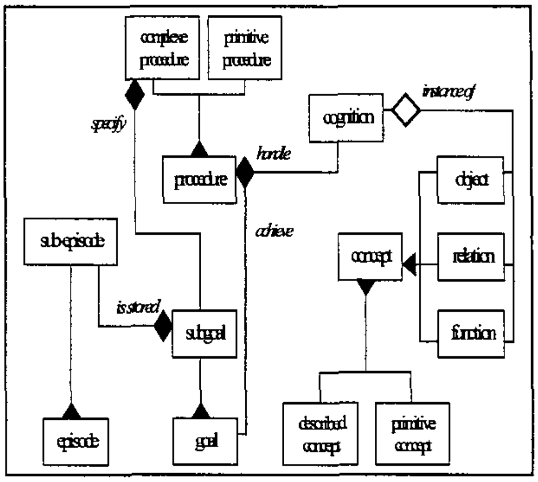

Figure 6. The knowledge structures relationships.

4. Original aspects

We think that it is practical to be inspired by a cognitive approach that attempts to model the human process of knowledge acquisition, to represent knowledge within our VL. To encode the knowledge to be taught, we use an architecture structured according to the human memory structures described earlier (semantic, procedural and episodic). We divide these structures into two categories : on one hand, semantic and procedural knowledge, common, potentially accessible and shared – with various mastery degrees – by all learners; and, on the other hand, episodic knowledge, specific for each learner and whose contents depend on the way with which the common knowledge (semantic and procedural) is perceived and handled. As shown in Figure 4, primitive units of semantic and procedural knowledge, chosen with a small level of granularity, are used to build complex knowledge entities which are dynamically combined in order to represent the learner acquired knowledge. Figure 6 shows a diagram that summarises our knowledge structures relationships. The dynamic aspect is seen in the non-predefined combinations between occurrences of concepts and the applied procedures handling them, which translate the learner interests (goals). Traces of the learner cognitive activity are structured as specific episodic knowledge. We use the episodic knowledge model – that the system formulates in its internal representation of the learner – to determine the mastery degree of procedural knowledge and the acquisition level of semantic knowledge. Retrieving the acquired knowledge has the advantage to permit to the tutor to impose, for example, a guided exercise and to force student to use and handle particular rules and/or concepts that s/he is often mistaken when applying them. The tutor can also can build suggestions and examples well adapted to each learner because they are built with specific and quite detailed cognitive elements that the learner has.

Another original aspect of our model is the explicit introduction of goals into the knowledge representation. Although they are treated by means of procedures, we consider goals are a special case of semantic knowledge that represents intentions behind the cognitive system actions. i.e., a goal is seen as a semantic knowledge which describes a state to be reached. Because learner's reasoning depends crucially on her/his goals and how likely s/he thinks actions will be successful to achieve them [2], there exist a particular form of energy employed to acquire goals. That distinguishes them from any simple form of semantic knowledge. This distinction involves a different treatment for goals in the human cognitive architecture. We propose to treat goals explicitly to reify them as particular semantic knowledge which is totally distinct from those which represent objects, relations and functions.

5. Conclusion

We have presented an algebra virtual laboratory for the boolean reduction problem solving. We have described the representation structures of the knowledge incorporated within the lab. We have also underlined some original aspects of our present work. Further work will be focused on other interesting aspects. We are now investigating a new idea for integrating pedagogic and didactic knowledge in our knowledge representation model. We are also experimenting with reusable knowledge techniques to provide suitable and efficient knowledge use in similar contexts.

6. Acknowledgement

The authors thank Guillaume Marion, Steve Massicotte, Ian Bailey, Pierre-Alexendre Ladouceur and Dominic Paradis for their help on an earlier version of the boolean reduction laboratory source code.

7. References

[1] Anderson J. R. (1989). *Psychology and Intelligent Tutoring.* in Bierman, Breuker and Sandberg (Eds). Artificial Intelligence and Education. Springfield. pp1

[2] Anderson, J. R. (1993). Rules of the mind. Lawrence Erlbaum, Hillsdale, New Jersey.

[3] Arruarte, A., Fernandez-Castro, I., Ferrero, B. & Greer, J. (1997). The IRIS Shell: How to build ITSs from pedagogical and design requisites. *International Journal of AI in Education*, (8) 341-381.

[4] Baddeley, A. (1990). *Human Memory : theory and practice.* Hove (UK): Lawrence Erlbaum.

[5] Gagné, R.M, Briggs, L. & Wager, W. (1992). *Principles of Instructional Design.* (4th edition), New York : Holt, Rinehart & Winston.

[6] Garagnani, M., Shastri, L., & Wendelken, C. (2002). *A connectionist model of planning as back-chaining search.* Proceedings of the 24th Conference of the Cognitive Science Society, Fairfax (Virgina). pp 345-350.

[7] Heermann, D. & Fuhrmann, T. (2000). *Teaching physics in the virtual university: the Mechanics toolkit*, in Computer Physics Communications, 127, 11-15

[8] Kokinov, B. & Petrov, A. (2000). *Dynamic extension of episodic representation in analogy-making in AMBR.* In Proceedings of the 22nd Conference of the Cognitive Science Society, NJ. 274-279.

[9] Lintermann, B. & Deussen, O. (1999). *Interactive modeling of plants*, IEEE Computer Graphics and Applications, 19 (1).

[10] Neely, J. H. (1989). Experimental dissociations and the episodic/semantic memory distinction. *Experimental Psychology: Human Learning and Memory*, (6), 441-466.

[11] Nkambou, R., Gauthier, G. & Frasson, C. (1996). *CREAM-Tools: an authoring environment for curriculum and course building.* Proceedings of the 3rd Conference on computer aided learning and instruction. Berlin.

[12] Rzepa, H & Tonge, A. (1998). *VChemlab: A virtual chemistry laboratory.* Journal of Chemical Information and Computer Science, Vol 38, pp1048-1053.

[13] Shastri, L. (2001). A computational model of episodic memory formation in the hippocampal system. Neurocomputing, 38-40: 889-897.

[14] Tulving, E. (1983). *Elements of Episodic Memory.* Oxford University Press, New York.

[15] Weber, G. & Brusilovsky, P. (2001). ELM-ART: An adaptive versatile system for Web-based instruction. *In International Journal of AI in Education* 12 (4) : 351-384.

[16] Wells, K. and Jeffrey, T. (1996). *LabVIEW for Everyone: Graphical Programming Made Even Easier*, Prentice Hall.