

Towards Rule-based System for the Assembly of 3D Bricks

Sanguk Noh

School of Computer Science and Information Engineering, The Catholic University of Korea, Republic of Korea

Copyright © 2015 by authors, all rights reserved. Authors agree that this article remains permanently open access under the terms of the Creative Commons Attribution License 4.0 International License

Abstract This paper concentrates on the design and preliminary implementation of an intelligent inference system for the assembly of 3D bricks. We model the 3D assemblage as the output of the assembly of 3D bricks. To intelligently guide the assembly of 3D bricks to the completion of a specific 3D assemblage, our system must be equipped with the component of identifying the necessary parts needed to complete the final product, and the component of checking whether the steps needed for completion are followed. Towards this end, we formulate logical rules represented as if-then rules into knowledge base. The knowledge base, thus, contains facts about the intermediate components of building bricks and their sequences given a specific domain. Our system then uses the logical rules in knowledge base and entails new facts from the knowledge base to provide a final assemblage with proper building components. Based upon the design of our system, we will develop and construct various knowledge bases for 3D assemblages, which are made for creative science education.

Keywords Intelligent Inference System, 3D Brick Assembly Process, Creative Science Education

facts about the intermediate components of building bricks and their sequences given a specific domain. Our system then uses the logical rules in knowledge base and entails new facts from the knowledge base to provide a final assemblage with proper building components. As a case study in this paper, we develop a rule-based system built by CLIPS [3] and JESS [4] to guide the assembly of a car using LEGO® bricks, and its graphic user interface to check whether or not a proper sequence of assembly has been performed. Based upon the design and preliminary implementation of our system, we hope that we are able to develop and construct various knowledge bases for 3D assemblages and apply them to the assemblages built using 3D bricks, which are made for creative science education.

The paper is organized as follows. In Section 2, we briefly compare our approach to related research. Section 3 briefly presents the simple domain of a 3D brick assembly process for creative scientific education, and is devoted to the design of our intelligent inference system. Section 4 provides a preliminary implementation of our system including both rule-based system and graphic user interface in a simple car assembly domain. In Section 5, we summarize our results and discuss further research issues.

1. Introduction

In this paper, we focus on the design and preliminary implementation of an intelligent inference system for the assembling process of bricks, which is made for creative science education. We model the 3D assemblage as the output of 3D brick assembly process. To intelligently guide the assembling process of 3D bricks given a specific 3D assemblage, our system should be equipped with the component of identifying the necessary parts of assemblage to be completed, and the component of checking the integrity of sequence for a complete assemblage. Towards this end, we formulate logical rules represented as if-then rules into knowledge base [1, 2]. The knowledge base, thus, contains

2. Related Work

To intelligently guide the assembling process of 3D bricks given a specific 3D assemblage, our system needs an inference engine, such as CLIPS and JESS. In this section, we briefly compare these two frameworks, and analyze their characteristics. We implemented these systems into the domain of 3D assemblages.

CLIPS (C Language Integrated Production System) [3] is a public domain building tool for expert systems. CLIPS deals with rules and facts in knowledge base, as depicted in Fig. 1. According to the user's responses to a series of yes/no or textual questions, the CLIPS inference engine can make a rule applicable and then be asserted into knowledge base as new facts. The Jess programming language [4] is also a rule-based system for the Java platform, which was

developed by E. Friedman-Hill at Sandia National Labs. The Jess rule engine utilizes the Rete algorithm. The pattern matcher of Jess inference engine applies a set of rules in rule base to a set of facts in fact base, as depicted in Fig. 2. We summarize the features of CLIPS and JESS in Table 1.

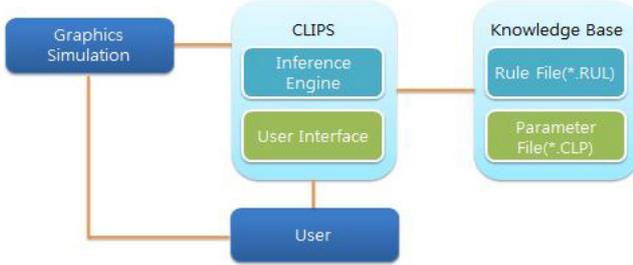


Figure 1. Architecture of CLIPS.

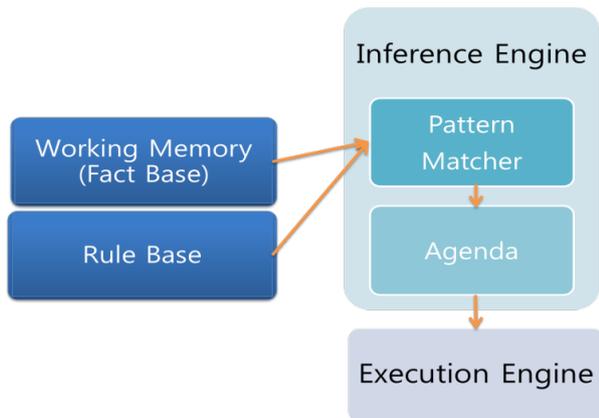


Figure 2. Architecture of JESS programming language.

Table 1. The features of CLIPS and JESS

	CLIPS	JESS
Common Features	<ul style="list-style-type: none"> Using Rete algorithm for production rule systems A declarative paradigm rather than a procedural paradigm 	
Platform	C or C++	Java
License	Public domain software	Commercial software but no cost for academic use
Developer	NASA-Johnson Space Center	E. Friedman-Hill of Sandia National Lab.
Website	http://clipsrules.sourceforge.net/	http://herzberg.ca.sandia.gov/

3. Designing Intelligent Inference System for Creative Scientific Education

We model a 3D assemblage as the output of the assembly of 3D bricks. In this section, we present an example domain of 3D assemblage and design an intelligent inference system that guides the process of assembly.

3.1 Example Domain of 3D Assemblage

Let us take a specific example of 3D car assembly domain. It consists of a number of 3D bricks, which can be used to build specific components of the car, such as the doors, the roof, bumper, trunk, bonnet, etc. To build the final 3D assemblage product, users must make an effort to use all the bricks comprising the 3D assemblage and also to maintain some sort of consistency in the order of their action sequences. In the 3D car assembly domain, for example, a steering handle and a roof are necessary parts for its assemblage, and the steering handle must be placed in the correct position before the roof can be assembled. From these perspectives, our inference system is needed in order to intelligently guide users in various domains of 3D brick assembly processes.

Fig. 3 shows a portion of the knowledge base that generated a question-answer dialog in the 3D car assembly domain. The representation of knowledge base [5, 6] consists of a series of bubble-shaped nodes. Each node has an identifier (a symbol), an associated simple question (a string), a ‘yes’ action, and a ‘no’ action. The yes actions lead to the identifiers of other nodes to go and the strings that give the necessary components for its assemblage. The hierarchy of the yes actions represent any action sequences, if exist, required to complete the 3D car assemblage. Whenever there is no new node to which to go, that is, in case of the no action, the program stops after displaying the incompleteness of the assemblage.

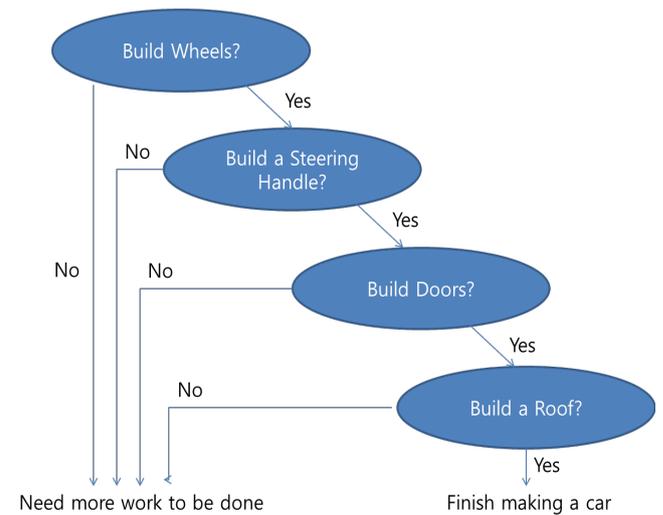


Figure 3. The representation of knowledge base in the example domain of 3D car assemblage.

3.2. Designing Intelligent Inference System

The intelligent guidance of the assembly of 3D assemblages will be supported by functions such as identifying the necessary parts and checking the integrity of sequence.

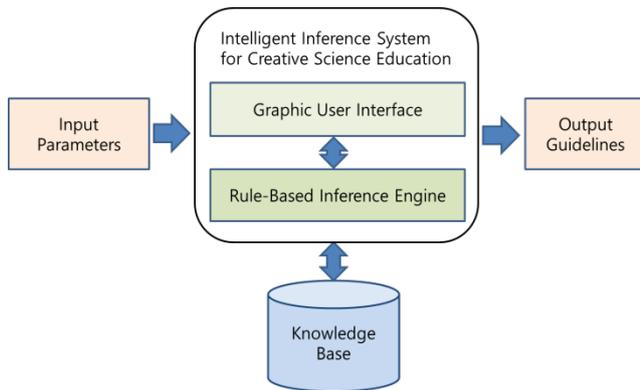


Figure 4. Architecture of intelligent inference system for creative science education.

In our system, given a specific 3D assemblage, its knowledge base defines the necessary components of the assemblage in the form of rules. While users make any progress to achieve their final assemblage, our rule-based inference system is automatically trying to check out its necessary components, and to continuously maintain the sequences of the components to be completed. The

architecture of our system is depicted in Fig. 4. The users' actions could be input into the system through its GUI and identified by our system. The inference engine of our system then uses the logical rules in knowledge base, and entails new facts from the knowledge base. The process of inference goes through three steps of firing rules, choosing rules that should be executed, and executing the rule matched, until no more new rules are generated into the knowledge base. Our system thus provides users with the guidelines for the necessary components of the output and its consistent sequence to make the correct assemblage.

4. Preliminary Implementation

We have implemented a prototype of our inference system using CLIPS and JESS as a rule-based inference engine, which runs on Microsoft Windows PC. This preliminary implementation simply shows the prototype of our inference system, and tests its capabilities of facilitating proper components and their correct sequence to be assembled.

```

Jess, the Java Expert System Shell
Copyright (C) 2001 E.J. Friedman Hill and the Sandia Corporation
Jess Version 6.1p4 7/8/2003

Jess> (reset)
TRUE
Jess> (run)
Did you build 'wheels' (yes or no)?yes
So far, you have assembled 'wheels'.
Did you build 'a steering handle' (yes or no)?yes
So far, you have assembled 'wheels', and 'a steering handle'.
Did you build 'doors' (yes or no)?yes
So far, you have assembled 'wheels', and 'a steering handle', and 'doors'. You finished making a car.
Jess> (reset)
TRUE
Jess> (run)
Did you build 'wheels' (yes or no)?yes
So far, you have assembled 'wheels'.
Did you build 'a steering handle' (yes or no)?no
To make a car, you need more work.
    
```

Figure 5. Example run using JESS.

4.1. Prototype Implementation in a Simple Domain

The prototype of our inference system provides guidelines for a simple assembly process. Fig. 5 shows the example run of our system using JESS in a simple 3D car assembly domain, as depicted in Fig. 3. In this case, the number of rules was less than 20 in both of CLIPS and JESS. To guarantee consistency throughout the sequences, the hierarchy of the components to be assembled was important. In Fig. 5, we show two example runs of a question-answer dialog, that is, one for a proper sequence of making a car and another for an incomplete case where an improper sequence is followed.

The GUI of our system intelligently identifies and automatically displays the required components of the assemblage, instead of requesting user answers, as depicted in Fig. 6. Given a final assembly, the system can provide various levels of difficulties by presenting different partial assemblages to users with varying capabilities of assembly. In other words, our system is able to present more advanced users with more parts, for instance, with six parts, which can be used to assemble partial assemblages that are closer in form to the final product. Conversely, for less experienced users, our system is also able to break down the assembly process into smaller steps by reducing the number of parts given. We will also construct a variety of knowledge bases for 3D assembly domains; for example, anything from cars, ships, airplanes, and tanks, to houses, palaces, and temples could become the final assemblage for

our system.

5. Conclusions

In a virtual reality environment, we model the 3D assemblage as the output of 3D brick assembly process, which could be applied to creative scientific education. To intelligently guide the assembling process of 3D bricks, given a specific 3D assemblage, we formulated logical rules represented as if-then rules into knowledge base. The knowledge base, thus, contains facts about the intermediate components of building bricks and their sequences given a specific domain. The prototype of our system was implemented and tested in a simple 3D car assembly domain.

Based upon the design of our system, we will develop and construct various knowledge bases for 3D assemblages, and apply them to the assemblages built with 3D bricks, which are made for creative science education. Further, we will extend our inference system for the operational control of creative 3D assemblage. Given vague input variables, our system using fuzzy logic [7, 8] could be used to control fuzzy operations, for instance, steering, accelerating, and braking, which usually could happen in operational control settings. Our inference system to support the assembly of 3D bricks and the operational control of creative 3D assemblages, as well, then can be equipped with both crisp rules and fuzzy rules to be practical in real-world domains.

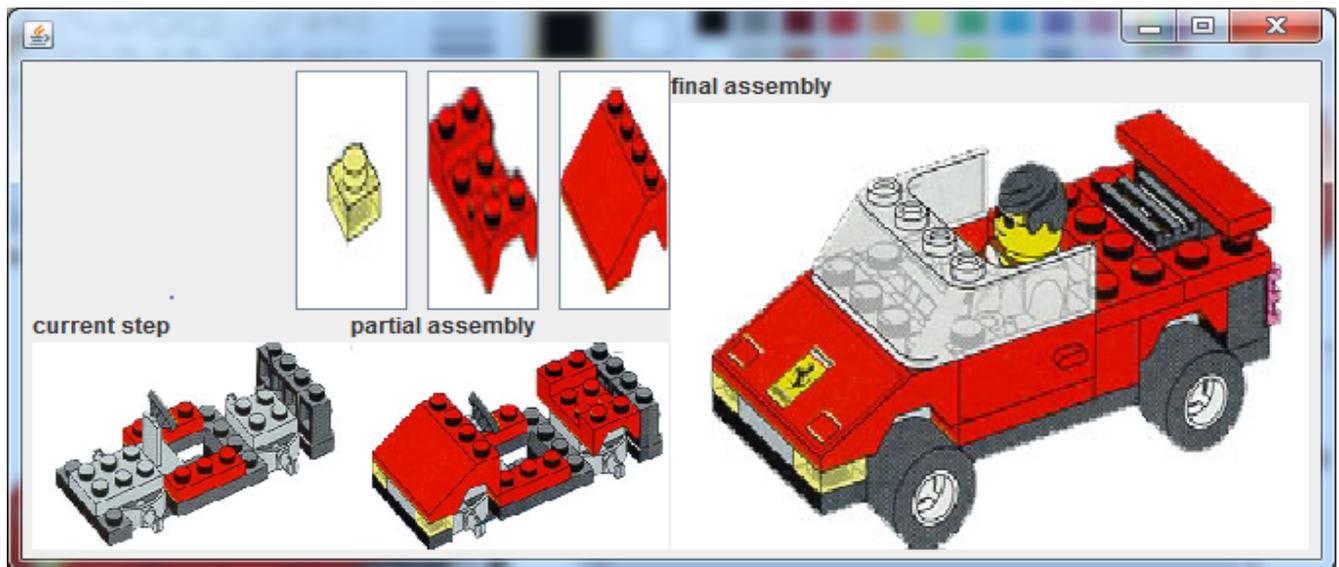


Figure 6. GUI for a 3D car assembly domain built using LEGO® bricks.

Acknowledgements

This paper is an extended version of our paper, which has been selected as an outstanding paper of APCSEE 2014, Nov. 7-9, 2014, Kuala Lumpur, Malaysia. We have added a graphic user interface to show the 3D brick assembly process. This work has been supported by the ICT R&D program of MSIP/IITP, Republic of Korea, [2014-044-063-003, "Development of the graphic engine and framework of 3D contents for the creative hands-on science education"]. We would like to thank our students, Jiyoung Lee, Seokgeun Hwang, and Wonyong Lee within the Center for Intelligent Systems at the Catholic University of Korea, for their help in preliminary implementation of our intelligent inference system.

REFERENCES

- [1] Abraham, A., *Intelligent Systems: Architectures and Perspectives*, Recent Advances in Intelligent Paradigms and Applications, in Studies in Fuzziness and Soft Computing, Chapter 1, (eds A. Abraham, L. Jain and J. Kacprzyk), Springer Verlag, Germany, pp. 1–35, 2002.
- [2] Giarratano, J. and Riley, G., *Expert Systems: Principles and Programming*, Fourth Edition, Course Technology, October 15, 2004.
- [3] Riley, G., *CLIPS: A Tool for Building Expert Systems*, Available from: <http://clipsrules.sourceforge.net/>, 2013.
- [4] Friedman-Hill, E., *Jess in Action: Java Rule-Based Systems*, Manning Publications, July 2003.
- [5] Hao, C., *Research on Knowledge Model for Ontology-Based Knowledge Base*, In Proceedings of International Conference on Business Computing and Global Informatization, 2011.
- [6] Shin, K., Kim, H., and Kim, H., *Development of the Knowledge-Based Systems for Anti-Money Laundering in the Korea Financial Intelligence Unit*, Journal of Intelligence and Information Systems, Vol. 14, No. 2, 2008.
- [7] Cordon, O., Herrera, F., Hoffmann, F. and Magdalena, L., *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*, World Scientific Publishing Company, Singapore, 2001.
- [8] Kim, J., *Extraction of Expert Knowledge Based on Hybrid Data Mining Mechanism*, Journal of Fuzzy Logic and Intelligent Systems, 2004.