Goal Trees and Ruled Based Expert Systems

Bruno Devesa, *Undergrad Student,ISEP, 1130838*, Joel Mateus, *Undergrad Student,ISEP, 1130939*

Abstract—This work involves discovering the expert based system and its goal trees process to achieve solutions for a large variety of specific problems. Mauris et odio ac dolor venenatis interdum. Nulla facilisi. Curabitur sed lectus ipsum. Praesent rhoncus diam ut nisl tempus pharetra. Vivamus ac lectus nec urna ultricies tristique ac ut dui. Duis tempor convallis risus, at fermentum magna tristique ac. Nam in turpis velit, sed elementum libero. Quisque adipiscing facilisis diam, blandit vulputate elit dignissim non.

Index Terms—Goal trees, Rule Based Expert Systems.

1 Introduction

THIS work is intended to be a generic review of the goal trees process and the expert ruled based system approach. This will have more text. Cras augue sapien, lacinia quis feugiat et, pulvinar non dolor. Nullam fermentum condimentum varius. Quisque nibh odio, varius eget bibendum quis, feugiat vitae est. Aliquam pharetra ornare tortor non imperdiet. Aliquam facilisis dui id quam pulvinar venenatis sed ut justo. Suspendisse ultricies semper sapien, id fringilla lacus condimentum eget. Vestibulum faucibus eleifend velit, ut euismod eros mattis non.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc sed velit felis, id rhoncus mi. Donec vel orci a lectus dictum ultricies. Nulla purus est, accumsan vitae congue sit amet, fermentum vestibulum massa. Pellentesque hendrerit, eros nec interdum volutpat, elit felis auctor neque, ac posuere sem libero id tortor. Vestibulum tincidunt magna non metus laoreet ac porttitor justo fringilla. Quisque eu dui arcu. In ac dui arcu, mattis dignissim erat. Phasellus ante diam, pharetra vitae pellentesque nec, commodo eget mauris. Nullam sodales sagittis lacus at rhoncus. Sed mollis rutrum pulvinar. Suspendisse non vestibulum ipsum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc sed velit felis, id rhoncus mi. Donec vel orci a lectus dictum ultricies. Nulla purus est, accumsan vitae congue sit amet, fermentum vestibulum massa. Pellentesque hendrerit, eros nec interdum volutpat, elit felis auctor neque, ac posuere sem libero id tortor. Vestibulum tincidunt magna non metus laoreet ac porttitor justo fringilla. Quisque eu dui arcu. In ac dui arcu, mattis dignissim erat. Phasellus ante diam, pharetra vitae pellentesque nec, commodo eget mauris. Nullam sodales sagittis lacus at rhoncus. Sed mollis rutrum pulvinar. Suspendisse non vestibulum ipsum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc sed velit felis, id rhoncus mi. Donec vel orci a lectus dictum ultricies. Nulla purus est, accumsan vitae congue sit ametr.

Porto,

March 11, 2016

- *Bruno Devesa and Joel Mateus are both computer science undergrad students at ISEP - Instituto Superior de Engenharia do Porto.*

1.1 Goal Trees

One of the often used techniques to solve problems is the "problem reduction technique" which is, basically to transform a complex problem into a more simple sub-problems.

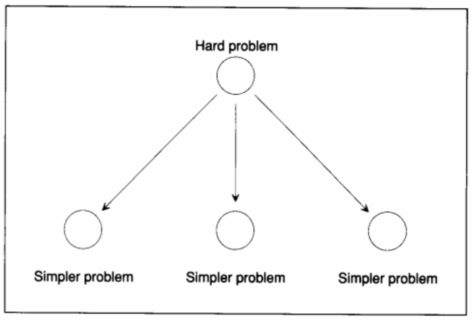

Fig. 1

A goal tree is also known as AND/OR tree because of the simple fact that its branch nature can hold AND nodes and OR nodes to represent the inference engine that supports the IF sentences which converge into a THEN clause.

Those IF/THEN clauses are called rules and they materialize the necessary knowledge to solve a particular problem, which is always an important question to make when facing a new problem:

"What knowledge is necessary to solve a problem?"

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc sed velit felis, id rhoncus mi. Donec vel orci a lectus dictum ultricies. Nulla purus est, accumsan vitae congue sit amet, fermentum vestibulum massa.

Pellentesque hendrerit, eros nec interdum volutpat, elit felis auctor neque, ac posuere sem libero id tortor. Vestibulum tincidunt magna non metus laoreet ac porttitor justo fringilla. Quisque eu dui arcu. In ac dui arcu, mattis dignissim erat. Phasellus ante diam, pharetra vitae pellentesque nec, commodo eget mauris. Nullam sodales sagittis lacus at rhoncus. Sed mollis rutrum pulvinar. Suspendisse non vestibulum ipsum. Mauris et odio ac dolor venenatis interdum. Nulla facilisi. Curabitur sed lectus ipsum. Praesent rhoncus diam ut nisl tempus pharetra. Vivamus ac lectus nec urna ultricies tristique ac ut dui. Duis tempor convallis risus, at fermentum magna tristique ac. Nam in turpis velit, sed elementum libero. Quisque adipiscing facilisis diam, blandit vulputate elit dignissim non.

1.1.1 *Moving blocks problem reduction illustration*

Moving blocks problem reduction illustration bla bla.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc sed velit felis, id rhoncus mi. Donec vel orci a lectus dictum ultricies. Nulla purus est, accumsan vitae congue sit amet, fermentum vestibulum massa. Pellentesque hendrerit, eros nec interdum volutpat, elit felis auctor neque, ac posuere sem libero id tortor. Vestibulum tincidunt magna non metus laoreet ac porttitor justo fringilla. Quisque eu dui arcu. In ac dui arcu, mattis dignissim erat.

Phasellus ante diam, pharetra vitae pellentesque nec, commodo eget mauris. Nullam sodales sagittis lacus at rhoncus. Sed mollis rutrum pulvinar. Suspendisse non vestibulum ipsum.

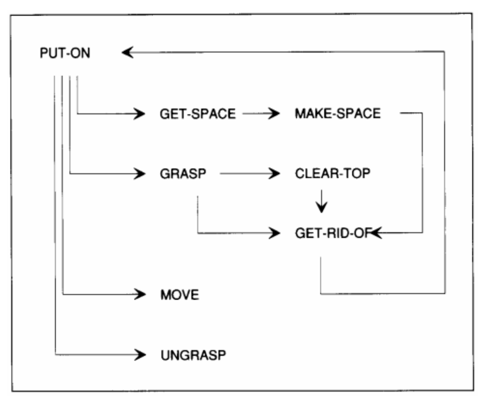

Fig. 2

Mauris et odio ac dolor venenatis interdum. Nulla facilisi. Curabitur sed lectus ipsum. Praesent rhoncus diam ut nisl tempus pharetra. Vivamus ac lectus nec urna ultricies tristique ac ut dui. Duis tempor convallis risus, at fermentum magna tristique ac.

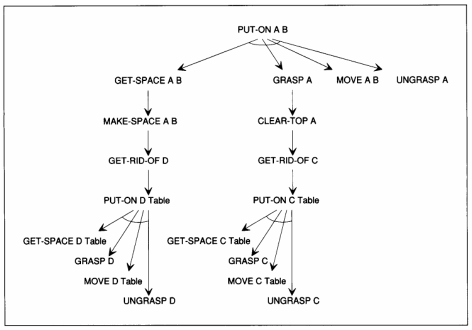

Fig. 3

Nam in turpis velit, sed elementum libero. Quisque adipiscing facilisis diam, blandit vulputate elit dignissim non.

1.2 Expert Based Systems

Conventional problem-solving computer programs make use of well-structured algorithms, data structures, and crisp reasoning strategies to find solutions. For the difficult problems with which expert systems are concerned, it may be more useful to employ heuristics: strategies that often lead to the correct solution, but that also sometimes fail. Conventional rule-based expert systems, use human expert knowledge to solve real-world problems that normally would require human intelligence. Expert knowledge is often represented in the form of rules or as data within the computer. Depending upon the problem requirement, these rules and data can be recalled to solve problems. Rule-based expert systems have played an important role in modern intelligent systems and their applications in strategic goal setting, planning, design, scheduling, fault monitoring, diagnosis and so on. With the technological advances made in the last decade, today?s users can choose from dozens of commercial software packages having friendly graphic user interfaces (Ignizio, 1991). Conventional computer programs perform tasks using a decision-making logic containing very little knowledge other than the basic algorithm for solving that specific problem. The basic knowledge is often embedded as part of the programming code, so that as the knowledge changes, the program has to be rebuilt. Knowledge-based expert systems collect the small fragments of human know- how into a knowledge base, which is used to reason through a problem, using the knowledge that is appropriate. An important advantage here is that within the domain of the knowledge base, a different problem can be solved using the same program without reprogramming efforts. Moreover, expert systems could explain the reasoning process and handle levels of confidence and uncertainty, which conventional algorithms do not handle (Giarratano and Riley, 1989). Some of the important advantages of expert systems are as follows: ? ability to capture and preserve irreplaceable human experience; ? ability to develop a system more consistent than human experts; ? minimize human expertise needed at a number of locations at the same time (especially in a hostile environment that is dangerous to human health); ? solutions can be developed faster than human experts. The basic components of an expert system are illustrated in Figure 1. The knowledge base stores all relevant infor- mation, data, rules, cases, and relationships used by the expert system. A knowledge base can combine the knowl- edge of multiple human experts. A rule is a conditional statement that links given conditions to actions or out- comes. A frame is another approach used to capture and store knowledge in a knowledge base. It relates an object or item to various facts or values. A frame-based repre- sentation is ideally suited for object-oriented programming techniques. Expert systems making use of frames to store knowledge are also called frame-based expert systems. The purpose of the inference engine is to seek information and relationships from the knowledge base and to provide answers, predictions, and suggestions in the way a human expert would. The inference engine must find the right facts, interpretations, and rules and assemble them correctly. Two types of inference methods are commonly used ? Backward chaining is the process of starting with conclusions and working backward to the supporting facts. Forward chaining starts with the facts and works forward to the conclusions.

1.2.1 What is a rule-based system ?

A rule-based system consists of if-then rules, a bunch of facts, and an interpreter controlling the application of the rules, given the facts. These if-then rule statements are used to formulate the conditional statements that comprise the complete knowledge base. A single if-then rule assumes the form ?if x is A then y is B? and the if-part of the rule ?x is A? is called the antecedent or premise, while the then-part of the rule ?y is B? is called the consequent or conclusion. There are two broad kinds of inference engines used in rule-based systems: forward chaining and backward chaining systems. In a forward chaining system, the initial facts are pro- cessed first, and keep using the rules to draw new conclu- sions given those facts. In a backward chaining system, the hypothesis (or solution/goal) we are trying to reach is pro- cessed first, and keep looking for rules that would allow to conclude that hypothesis. As the processing progresses, new subgoals are also set for validation. Forward chaining systems are primarily data-driven, while backward chain- ing systems are goal-driven. Consider an example with the following set of if-then rules Rule 1: If A and C then Y Rule 2: If A and X then Z Rule 3: If B then X Rule 4: If Z then D If the task is to prove that D is true, given A and B are true. According to forward chaining, start with Rule 1 and go on downward till a rule that fires is found. Rule 3 is the only one that fires in the first iteration. After the first iteration, it can be concluded that A,B, and X are true. The second iteration uses this valuable information. After the second iteration, Rule 2 fires adding Z is true, which in turn helps Rule 4 to fire, proving that D is true. Forward chaining strategy is especially appropriate in situations where data are expensive to collect, but few in quantity.

1.3 Current state of the art

Current state of the art bla bla Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc sed velit felis, id rhoncus mi. Donec vel orci a lectus dictum ultricies. Nulla purus est, accumsan vitae congue sit amet, fermentum vestibulum massa. Pellentesque hendrerit, eros nec interdum volutpat, elit felis auctor neque, ac posuere sem libero id tortor. Vestibulum tincidunt magna non metus laoreet ac porttitor justo fringilla. Quisque eu dui arcu. In ac dui arcu, mattis dignissim erat. Phasellus ante diam, pharetra vitae pellentesque nec, commodo eget mauris. Nullam sodales sagittis lacus at rhoncus. Sed mollis rutrum pulvinar. Suspendisse non vestibulum ipsum. Mauris et odio ac dolor venenatis interdum. Nulla facilisi. Curabitur sed lectus ipsum. Praesent rhoncus diam ut nisl tempus pharetra. Vivamus ac lectus nec urna ultricies tristique ac ut dui. Duis tempor convallis risus, at fermentum magna tristique ac. Nam in turpis velit, sed elementum libero. Quisque adipiscing facilisis diam, blandit vulputate elit dignissim non.

2 Conclusion

The conclusion goes here. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc sed velit felis, id rhoncus mi. Donec vel orci a lectus dictum ultricies. Nulla purus est, accumsan vitae congue sit amet, fermentum vestibulum massa. Pellentesque hendrerit, eros nec interdum volutpat, elit felis auctor neque, ac posuere sem libero id tortor. Vestibulum tincidunt magna non metus laoreet ac porttitor justo fringilla. Quisque eu dui arcu. In ac dui arcu, mattis dignissim erat. Phasellus ante diam, pharetra vitae pellentesque nec, commodo eget mauris. Nullam sodales sagittis lacus at rhoncus. Sed mollis rutrum pulvinar. Suspendisse non vestibulum ipsum. Mauris et odio ac dolor venenatis interdum. Nulla facilisi. Curabitur sed lectus ipsum. Praesent rhoncus diam ut nisl tempus pharetra. Vivamus ac lectus nec urna ultricies tristique ac ut dui. Duis tempor convallis risus, at fermentum magna tristique ac. Nam in turpis velit, sed elementum libero. Quisque adipiscing facilisis diam, blandit vulputate elit dignissim non.

Appendix A
Proof of the First Zonklar Equation

Appendix one text goes here.

APPENDIX B

Appendix two text goes here.

ACKNOWLEDGMENTS

The authors would like to thank professor .. and professor..

REFERENCES

[1] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.

PLACE PHOTO HERE

Michael Shell Biography text here.

John Doe Biography text here.

Jane Doe Biography text here.