Case Discussion (Human Robot Game of Yama)

Syntax Analysis

Divya Kundra and Ashish Sureka, *An Experience Report on Teaching Compiler Design Concepts using Case-Based and Project-Based Learning Approaches*, International Conference on Technology for Education (T4E 2016)

GOLEMS is Humanoid Robotics Lab at Georgia Institute of Technology. The lab works towards developing robots having human and even super human capabilities. The current task of GOLEMS lab is to build and demonstrate a robot system playing the game of Yama with the human. In this scenario a system needs to be built in which the robot repeatedly competes with a human opponent in a sustained game- play.

As robots come into increasing contact with humans, it is absolutely vital to prove that these potentially dangerous machines are safe and reliable. Furthermore, applying robots to increasingly complicated and varied tasks requires a tractable way to generate desired behaviors. Typical reactive strategies do not provide a way to prove how the system will respond during complicated tasks with uncertain outcomes. Existing deliberative planners often simplify the control system or have prohibitive computational cost

Thus developers at GOLEMS have come up with controlling of robotic systems using Context-Free Grammars. The name they have given to the Context Free Grammar is -Motion Grammar which enables robots to handle uncertainty in the outcomes of control actions through on-line parsing. . For example to build the grammar for a robot that does loading and unloading task is as shown below in Fig. 1. The associated parse tree is also shown. In production (1), the system will repeatedly perform [load] operations until receiving a [full] token from production (2). Then the system will perform [unload] operations of the same number as the prior [load] operations. This simple use of memory is possible with Context- Free systems.

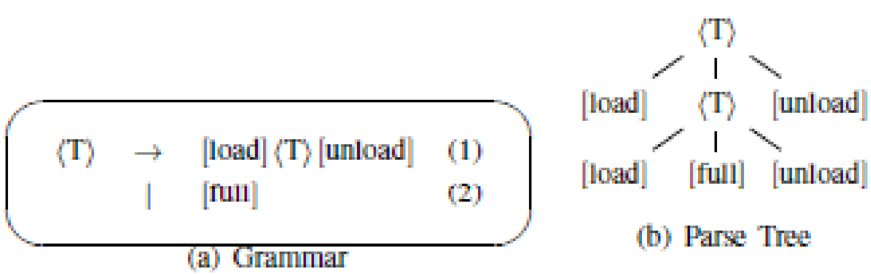

Fig. 1 : CFG and parse tree for loading and unloading task

The grammar makes it possible to prove that the system will respond to all possible situations. Imagine a robot searching for earthquake survivors. This robot must carefully remove pieces of rubble while ensuring that the larger structure does not collapse. It must also coordinate its efforts with other robots and human rescuers. A simplified version of this scenario is considered in the two-player game of Yamakuzushi (Yama). The precise requirement of the system is that to implement and evaluate the performance of Motion Grammar on Japanese game Yama. This game is similar to Jenga. In yama, a mountain of pieces is randomly piled in the middle of a table. Each of the two players tries to clear the pieces from the table. Each player is only allowed to use one finger to move pieces. If the player causes the pieces to make a sound, it becomes the other players turn. The winner is the player who removes the most pieces. Thus in the implementation human plays against the robot arm. The robot has a mesa swiss ranger that allows it to locate the pieces and a microphone that allows it to detect sounds. The robot used a speaker and text-to-speech program to communicate with its human opponent. Thus Motion Grammar has to be built for the robot. The game contains many of the challenging elements of physical human-robot interaction. For instance, both the robot and human make careful contact with pieces without causing them to fall. The robot slides pieces without losing contact. It might happen that the robot loses contact with the piece. In such an uncertain scenario it will have to backtrack to the last contact position to reacquire it. By taking care of such a situation it allows the robot to move pieces across the table and recover any pieces that it loses. The robot should also know how to pick the target pieces. The algorithm followed to solve this problem favors the target pieces on bases of the relative distance of the pieces from the ground. The pieces highest above the ground will be picked up. Robot also observes human actions and handles dangerous conditions. The key to all these interactions is that the robot must handle uncertainty. The Motion Grammar guarantees that the robot responds to the complete range of uncertain outcomes online. Most robot tasks can be recursively divided into a number of simpler subtasks. The hierarchical nature of grammatical productions and the corresponding parse trees are well suited to representing this hierarchical task decomposition. While designing the grammar, care has to be taken that the grammar provides the functionality of alternating turns between human and robot when a sound is detected. This process needs to be followed till the board is cleared. During the robot turn it will repeatedly act to remove pieces until it causes sound by making noise exceeding a threshold or until it clears the board. During the human turn, the robot will simply wait until it detects a sound or sees that the board has been cleared. After the human makes a sound, the robot will wait until the human is out of the workspace before it begins its turn. The grammar should also clearly declares the winner. This can be done when a count is kept of the number of pieces removed by human and robot. Thus after identifying the requirement of the system to be build and think about some more conditions that are needed to be taken care of, suggest a suitable Motion Grammar.