Jaahanavee Sikri

jxs163330

**Project Report**

**Self-driving Car**

The objective of this project is to simulate an expert driving a car as accurately as possible, i.e. rules have been written for a self-driving car.

It is fascinating how well self-driving cars are performing. There are extremely minute details that need to be considered in every situation, which helps with building such a system using s(ASP), because formulating specific rules and defining facts is what the system is based on.

It is assumed that the car has various sensors, which receive appropriate information that is then converted into the facts used in this project.

Here it is also considered that cameras provide information regarding what the current surroundings are, which have been converted to predicates namely road\_left/1, road\_right/1, road\_front/1 and road\_back/1. These predicates are then used to identify the position of the car depending on what road markings are visible. The position of the car is a lane(right/left/middle) the car is in; at present.

E.g. The car will be in the middle lane if the road marking to the left and right of it are white broken lines.

Depending on the weather and visibility, rules have been provided so that the correct car components (headlights and wipers etc.) can be used in the correct manner.

E.g. The car headlights will be used when the visibility is lower than 150 meters or if there is rain or snow.

Rules have been formulated for various car actions like parking, reversing, making turns, stopping at signals, yielding in certain situations etc. In these situations, there are various predicates that need to evaluate to true for a certain action to be performed. Various possibilities regarding the surroundings must be considered.

If no information is provided regarding the change in inputs, then the output for car\_action(X) is X= keep\_going\_straight.

E.g. For the car to make any turn, it should be in the appropriate lane, the correct turn signal should be on so that other cars are aware. At that intersection, that turn should be allowed and there should be no oncoming traffic, i.e. it should be safe to turn.

Rules exist for changing lanes and checking if a turn/lane is safe.

All the specific information has been obtained from the online driving course, which in turn uses information from the Texas Manual on Uniform Traffic Control Devices (TMUTCD). My driving experience has also helped me in fine tuning the rules.

It was a completely different experience using the s(ASP) system to make the project. Driving is a learned behavior and the only way to learn it, is to follow the specific rules and regulations that exist. While driving every possibility needs to be assessed, so each rule has various predicates in its body.

To make a certain decision, the program needs to be given every small detail since there is no notion of learning through practice here.

Working with this system, I learned how important the small details are when we consider machines doing the work that we do.

I also observed that not a lot of code needs to be written when using the s(ASP) system, which was a welcome change.