ASP Challenge Problem: Automated Warehouse Scenario

Shraddha Nandakishor Joshi
ASU ID: 1225456531
snjoshi3@asu.edu

Problem statement

The problem statement for the project is from ASP Challenge 2019. The challenge is to find an optimal solution for the Automated Warehouse Scenario. The core idea is to make robots deliver packages inside warehouses to a picking station. Robots can pick up the shelves containing the products that need to be delivered as per the order, then drop them at the picking station. The order should be completed as soon as possible, based on the product's location. This represents a real-time business warehousing scenario for all that need optimal resource management.

Here is the problem in more detail, a warehouse is reimagined as a rectangular grid with X and Y coordinates. In the grid, robots have the freedom to move in vertical and horizontal directions but not diagonally. The robots can move into the shelf's cell to pick the shelf. A robot is considered flat with the ability to go under the shelf and pick it up. The robots can do the following actions, move around in the cells, pick up and put down the shelves, deliver shelves, or deliver products. While performing the actions robots should not collide, that is there can be only one robot in one cell. Robots cannot move into another robot's cell or switch its shelf with another robot from one step to another. The grid can have highways. On highway cells, robots cannot put down the shelves. Additionally, all robots need not be active in every time step. For each order, only one pickup station is mapped. The robots deliver the shelves to the pickup station.

The main goal is to complete the received orders in the minimum time possible. Time taken to complete the order is counted in terms of the number of steps taken by robots.

Summary of progress

For the Knowledge Representation and Reasoning course, we are expected to implement the given problem statement using CLINGO and ASP. Answer Set Programming is a kind of declarative programming. CLINGO is the collection of ASP-related tools created at the University of Potsdam in Germany, called Potassco (for Potsdam Answer Set Solving Collection). To learn more about ASP and understand concepts I started watching lecture modules. The modules and the Answer Set Programming book helped me better grasp concepts like stable models, choice rules, and constraint encoding. The description files in the project provide various constraints and rules of motion. I could define the base state of the grid and implement the robot's motion based on principles learned from the multiple examples provided in the coursework.

The description.pdf file and introductory video of the automated-warehouse-scenario are given under the course project section. This has helped me understand provided input and expected output. The given example aids in imagining how a robot would move on a real grid and deliver a product to its destination.

Further, based on my understanding I divided the complete project into small chunks. Firstly, I prepared an outline of steps to be encoded in simple English sentences. I followed the ASP Generate-Define-Test model to design the flow of code. Then by creating small custom instances I was able to write and test the robot's move action. I have planned to continue by adding more constraints for shelves and expanding grid sizes as per the given instance files.

Challenges encountered

After going through the problem statement, I faced the following challenges while organizing, planning, and designing the solution:

- Issue understanding provided instance files and the init code.
- Difficulty in forming hard constraints.
- Difficulty understanding concepts of transition systems like fluents and exogenous states.
- Difficulty finalizing the plan of action.

Plan to resolve these issues

Took the below measures to overcome the challenges while completing milestone 3:

- Referred examples provided in syllabus book on Answer Set Programming by Lifschitz.
- Followed example provided in the project explanation file to understand instance files.

- Referred to module 5 video lectures to better understand transition systems fluents and exogenous states
- Wrote down steps in hard constraints and function behaviors as English sentences to get a sense of direction in which efforts are needed.
- Referred blocks world problem implementation as beginners code for warehouse problem.

Completed tasks

After resolving the issues faced in understanding the problem statement. I was able to complete the following steps:

- Create a frame of steps to be implemented by listing down all the actions and constraints to solve the problem statement.
- Encode the basic grid with nodes
- Encode highway nodes
- Encode highway constraints for robots
- Encode the robot's movement action
- Add constraint so only empty robot can move under the shelf
- Add constraint so that robot can move only in the horizontal and the vertical direction
- Add a constraint to avoid collision between two robots and avoid two robots occupying the same node.
- Create custom instances with a 2*2 grid to test implemented logic and constraints.
- Encode the law of inertia for robots.

Future plan

Following is a step-wise plan to complete the remaining implementation of constraints and actions:

- Encode shelves onto the grid.
- Encode a constraint so that robot can pick only shelves and not products,
- Encode shelf pickup.
- Encode shelf put down.
- Encode deliver action.
- Encode effects of actions.
- Encode the law of inertia for shelves, products, and objects.
- Test the code on provided input instance files
- Minimize robots' time steps to improve the solution.
- Identify corner cases and scenarios.

References

Lifschitz, V. (2019). *Input Language of CLINGO. In: Answer Set Programming* Springer, Cham.

Lifschitz, V. (2019). *Introduction. In: Answer Set Programming* Springer, Cham.

Nguyen, V., Obermeier, P., Son, T. C., Schaub, T., Yeoh, W. (2017). *Generalized Target Assignment and Path Finding Using Answer Set Programming* IJCAI (pp. 1216-1223). ijcai.org.