

Cole Linke

ID: 011917160

C951 – Intro to Artificial Intelligence

Task II

A. Disaster Environment

My disaster recovery situation is a fire breaking out in an office. I have three major sets of obstacles that are composed of many pieces. The first being an office table represented as a rectangular cuboid, and eight chairs represented by cylinders. The second being three sets of desks and chairs, again represented by rectangular cuboids and cylinders. Finally, four long cuboids in the shape of a square represent the walls of the office. The fire is represented by a red cone.

B. Improved Disaster Recovery

The robot's job in this situation is to search the office and locate the fire. Once the fire is located, it could alert the authorities to its exact location or even put out the fire itself. The robot's behavior is dependent on which entity it encounters. If it comes in contact with a wall, table, desk, or chair, it will reverse in a curved shape and continue forward. When the robot senses a fire in front of it, it will stop and produce a print statement saying, "Fire Detected".

C. Architecture

My submission is a modified version of the bubbleRob tutorial.

Sensor 1: The first sensor is the same as the original one in the tutorial. This sensor is used to detect walls and obstacles so it can maneuver around the room. It has a small detection cone, just large enough so that it doesn't bump into things.

Sensor 2: The second sensor is one that I added. This sensor is used to scan the room for a specific cone labeled "Fire" and will ignore all other objects. Once the object has been detected, the robot will stop and produce a line of text in the console. The detection cone

for this sensor is a bit larger than the first sensor, so that the robot does not move into the fire.

D. Internal Representation of Environment

The robot does not store information about the environment in memory but instead calculates all decisions in real-time. One sensor is used to detect obstacles, and the other is used to detect a specific object representing a fire. When the obstacle detecting sensor detects an object, it will back up in a curved fashion and try again. When the fire detecting sensor detects the fire object, it will stop.

E. Implementations

- Reasoning – The robot makes decisions using its two sensors. There are two questions that the robot is answering constantly: “Is this a fire?” and “Is this an object?”. Each sensor is responsible for one of those questions. If at any point the questions trigger a true response, the robot will react accordingly by either adjusting itself to avoid the obstacle or stopping and displaying a message.
- Knowledge Representation – The robot calculates all of its actions in real-time and does not store any information about the environment. Each of the two sensors is continually testing the environment to determine if it needs to react.
- Uncertainty – Because the robot does not store any information about the environment, it is already continually adapting to the environment. If the environment were to change, the robot would function in the exact same way as in the previous environment. It uses sensors to test the environment for particular conditions or obstacles and reacts accordingly.
- Intelligence – The robot uses real-time decision-making to determine whether its sensors have come into contact with an obstacle or have detected a fire. It will move around in an environment, avoiding obstacles while searching for an object representing a fire.

F. Improvements

One major improvement the robot would benefit from is having a way to store the location of objects in memory. This would streamline the process of sweeping through a room because it wouldn't have to blindly search for the objective object while bumping into random obstacles. This registration could be done at the time of installation with a scan that logs the location of all obstacles into memory, so during a crisis scenario, the robot is more efficient.

One way the robot could benefit from implementing reinforced learning is by cataloging locations. Positive reinforcement would be when the robot locates a fire, and that data could be used to identify areas that are more prone to fires. Negative reinforcement would be when the robot gets stuck, and that data could be used to avoid areas that are difficult to navigate.

One way the robot could benefit from implementing an advanced search algorithm is by using the data gathered from reinforced learning to assist with navigating the environment.

An example algorithm could be that fire-prone locations are added to a matrix, and the difficulty of navigation is added as weighted lines. The algorithm would find the path of least resistance by adding the weights of each possible path. This way, the robot would avoid areas that are overly difficult to navigate while still visiting each location where a fire could be.

G. Code

Included in submission

H. Panopto Video

[Panopto Recording](#)

I. Sources

No external sources used