Tornado-Zone Disaster Recovery Robot Prototype

Alina Hendrix

C951 – Introduction to Artificial Intelligence

A. Describe the disaster recovery environment you chose and the **two** obstacles you have added to the environment.
   - For this simulation, I focused on a tornado-stricken residential area with collapsed infrastructure and debris-blocked streets. I chose this environment because it presents real-world challenges first responders often face—especially when time is critical and human access is risky.
   - To make the simulation more realistic, I added:
     - A collapsed wall segment that blocks a direct path through the environment, requiring the robot to reroute around it
     - A simulated flooded zone, using a flat plane with altered texture and color to represent standing water or mud that can't be crossed

       These two obstacles force the robot to make real-time navigation decisions and mirror what it would encounter in an actual disaster zone.

B. Explain how the robot will improve disaster recovery in the environment from part A after you have added the **two** obstacles from part A.

- Once the obstacles were in place, I programmed the robot to act as a first-wave scout. It scans for hazards, determines viable paths, and helps map the area. In a real deployment, this kind of support could reduce search time and keep people out of danger.

  The robot:

  - Actively avoids debris like the collapsed wall
    - Identifies unsafe zones (like the simulated flood area)
    - Continues pursuing its goal while collecting data for human teams who follow up

C. Justify the modifications you made to CoppeliaSim's robot architecture, including two sensors you chose to add, and explain how these sensors will aid the disaster recovery effort.

- I modified the base BubbleRob by adding two key sensors:

  - Proximity Sensor: Attached to the torso, this allows the robot to detect and avoid physical obstacles like walls or rubble.
  - Vision Sensor: Mounted near the head, it reads terrain texture and color to detect flooded or unsafe zones.

These sensors support situational awareness and intelligent rerouting. The robot uses sensor input to respond to the environment instead of executing static pre-programmed paths.

D. Describe how the robot maintains an internal representation of the environment.

- While this prototype doesn't store a persistent map, it forms a dynamic internal model based on sensor input. As it moves, the robot interprets what it sees and adjusts behavior accordingly—building a mental snapshot that updates in real time. This lets it reroute mid-journey if hazards appear.

E. Explain how the robot implements the following four concepts to achieve its goal:

- Reasoning: The robot alters its path when obstacles or unsafe terrain are detected.
- Knowledge Representation: Sensor readings are translated into control commands (e.g., "flood zone = turn").
- Uncertainty: It operates without prior terrain knowledge, reacting to hazards as they're encountered.
- Intelligence: NAO adapts continuously to reach its goal, showing autonomous decision-making based on environmental input.

F. Explain how the prototype could be further improved, including how reinforced learning and advanced search algorithms can improve the prototype's performance and learning.

1. Reinforcement Learning (RL)
   - Reinforcement learning would enable the robot to learn optimal behaviors by interacting with the environment and receiving feedback:
     - Experience-Based Learning: The robot accumulates knowledge through trial and error—receiving rewards for successful navigation and penalties for entering hazard zones or inefficient paths.

    - Action Optimization: Algorithms like Q-learning or Deep Q-Networks (DQN) help the robot identify and favor actions that lead to better long-term outcomes, such as faster navigation or safer rerouting.
    - Environmental Adaptation: Over time, the robot learns to recognize patterns—such as certain textures indicating flood zones—and proactively adjusts its decisions to avoid obstacles.

   This learning process transforms static responses into dynamic, experience-driven intelligence, resulting in safer and more efficient mission outcomes.

2. Advanced Search Algorithms
   - Advanced pathfinding algorithms improve the robot's ability to navigate complex or changing terrain:
     - A*: Evaluates multiple path options using a heuristic (e.g., distance-to-goal), enabling efficient path selection when obstacles are known.
     - D* Lite: Supports real-time path recalculations, making it ideal for unpredictable environments where conditions may change mid-task (e.g., additional debris falling).

   By using D* Lite, the robot can respond dynamically to new hazards while maintaining progress toward its goal.

3. SLAM Integration
   - Adding Simultaneous Localization and Mapping (SLAM) capabilities would allow the robot to:
     - Build and update maps in real time
     - Navigate previously unexplored zones with greater precision
     - Maintain environmental memory across missions

   Together, reinforcement learning, SLAM, and advanced search algorithms transform the robot into a continuously improving disaster recovery tool.

G. Code Submission

Included in separate file

H. Panopto Video

Narration covers:

1. Statement of the disaster recovery problem
2. Environment overview and obstacles
3. Robot’s mission and goals
4. Description of architecture and sensors
5. Demonstration of behavior and rerouting
6. Performance summary
7. Proposed improvements

   Link included separately.

I. References

Coppelia Robotics. (n.d.). *CoppeliaSim User Manual*. https://www.coppeliarobotics.com/helpFiles/en/index.html

Russell, S., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach (4th ed.). Pearson.

https://aima.cs.berkeley.edu