CONESTOGA

Connect Life and Learning

Reinforcement Learning Programming

CSCN8020 - Fall 2024 - Section 1

Project Proposal

Submitted By: Group 7

Christina Arokiaswamy (8923489)

Li-Cheng, Wang (8990916)

Autonomous Lawn Mower Using Reinforcement Learning

---

1. Project Overview

The purpose of this project is to develop an autonomous lawn mower that can mow a lawn by cutting the grass in a grid environment while avoiding obstacles like trees, rocks, and uncuttable terrain in a efficient way. The environment will be modeled as a discrete grid with specific cells representing grass, obstacles, and agent position. The agent (lawn mower) will interact with this environment by taking actions that either move it to adjacent grid cells or stay in place. The task will be to maximize the amount of grass cut while avoiding penalties associated with obstacles or idle actions.

The project will focus on using Reinforcement Learning (RL) algorithms to train the lawn mower agent. We aim to explore and compare multiple RL algorithms for this task for different grid, focusing on both the design of the environment and the algorithmic approach used for training.

---

2. Problem Description

An agent must find an optimal path to mow as much grass as possible in a predefined grid while avoiding obstacles like tree, stone and terrain etc. The environment is dynamic, and the agent must make decisions based on its current state to achieve its goal efficiently. Key elements of the problem are:

- **Grass-Cutting Objective**: The agent needs to maximize the area of grass it cuts within the grid.
- **Obstacle Avoidance**: The agent must avoid obstacles such as trees, rocks, and uncuttable terrain.
- **Efficiency**: The agent should avoid redundant movements, minimize idle steps, and avoid penalties.

The lawnmower must interact with a simulated environment that mimics the real-world problem of autonomous navigation in a bounded space with obstacles.

---

3. Project Scope

This project will explore:

- **Algorithm Implementation**: We will implement or leverage existing reinforcement learning algorithms to train the lawn mower. The RL algorithms will be used to find optimal policies for navigating the grid efficiently while avoiding obstacles.
- **Environment Design**: Creating a custom OpenAI Gym environment that simulates a lawn where specific cells are grass, obstacles, and empty spaces. The environment will also have a reward system that encourages the agent to mow the grass while avoiding obstacles.
- **Comparisons and Analysis**: Various RL algorithms will be compared based on their performance in this environment. Metrics such as the total grass cut, number of steps taken, penalties incurred, and training time will be used for comparison.

---

4. Algorithms to be Used

1. Deep Q-Learning (DQN):

The agent chooses actions on the Q-values, attempting to maximize cumulative rewards. DQN is a value-based reinforcement learning method used to estimate the future rewards for each state-action combination. This algorithm is more appropriate for assessing how successfully the lawnmower learns to navigate the grid, cut grass, and avoid obstacles in situations with discrete state-action pairings.

2. Proximal Policy Optimization (PPO):

More stable and sample-efficient than other policy gradient techniques, PPO is a policy-based reinforcement learning system that aids in policy optimization by modifying the probability distribution over actions to maximize rewards. This will be applied in settings where policies need to be adjusted and the agent needs to stay out of trouble all the time.

3 **Comparative Focus**:

We will compare the performance of **DQN** and **PPO** on the following metrics:

- **Total Reward**: The cumulative reward the agent receives throughout the episode.
- **Steps Taken**: The number of steps required to complete the task or the episode.
- **Penalties**: The number of penalties incurred due to idle actions or hitting obstacles.
- **Training Efficiency**: The number of episodes required for the agent to converge on an optimal policy.

---

5. Project Focus

The project will focus on two main aspects:

1. **Algorithm Implementation**:
    - W will focus on testing and evaluating the algorithms for optimal performance in this environment.
    - **Primary Focus**: The RL algorithm itself—testing how well different algorithms can learn efficient mowing strategies in the custom environment.
    - **Secondary Focus**: Tuning the hyperparameters of each algorithm to improve their learning rate, exploration behavior, and convergence efficiency.
2. **Environment Design**:
    - Developing a robust simulation environment with realistic obstacles like trees, rocks, and uncuttable terrain. This environment will be designed using OpenAI Gym, which allows us to simulate agent-environment interaction in a grid-based world.
    - The reward structure and obstacles are critical to creating a challenging and dynamic environment that encourages intelligent decision-making.

---

6. Comparative Experiments and Evaluation

To evaluate how of **DQN** and **PPO** RL algorithms perform under various conditions. Specifically, with the following parameters:

- **Grid Size**: Small (5x5), medium (10x10), and large (20x20) grids
- **Obstacle Density**: The number of obstacles will be increased to analyze the algorithms on how its handle complex environments with limited space for movement.
- **Reward Structures**: Testing different reward structures, including:
    - Larger rewards for covering uncut areas efficiently.
    - Heavy penalties for idle behavior or hitting obstacles.

Conclusion

This project will focus more on the **reinforcement learning algorithm performance** and **environment** for an autonomous lawn mowing task. By comparing the algorithms and different grid size and environment conditions, we aim to provide insights into how different RL approaches handle path-planning and obstacle avoidance in complex environments. The end goal is to develop an agent that can effectively and efficiently mow a lawn by avoiding obstacles and optimizing its path.