Autonomous Exploration System for Simulated Volcanic Terrain Using Markov Decision Process (MDP)

CSE440

Group: 5

Sabrina Sabnom Katha (ID: 2012917642)

S.M. Jabir Mahmud (ID: 2031752642)

Mumin Ahmod (ID: 2011926642)

Rafidul Islam (ID: 2022388 642)

Abstract:

An Autonomous exploration system is crucial for navigating environments that pose high-risk factors, such as volcanic terrains. This study explores the application of Markov Decision Process (MDP) in designing an autonomous agent capable of efficiently and safely navigating a simulated volcanic landscape. Where an agent learns to navigate safely while avoiding hazards (lava, gas, and craters) and aiming to reach the bottom-right corner of the grid. The agent operates under uncertainty, with obstacles like lava flows, craters, and gas emissions. Through the use of Q-learning, an off-policy reinforcement learning algorithm, the agent learns to balance exploration and exploitation for safe and effective navigation. In our results show that epsilon decay in the exploration strategy allows the agent to Incrementally shift from exploring new areas to exploiting known safe paths. In this performance of the agent was evaluated across various episodes, with total rewards indicating increasing efficiency and safety over time. The findings suggest that MDP-based autonomous systems can be effectively used for hazard-prone exploration tasks.

1.Introduction

Autonomous exploration systems are an increasingly growing subject of study and very valuable because, in places where it's too dangerous or impractical for humans to go, this system becomes a necessary tool. There are many use cases such as deep ocean, outer space, war zone mine fields, and volcanic regions. These systems are made based on different intelligent decision-making models that can adapt in different changing conditions and various complex scenarios. These models also need to be smart to make choices and implement stay-safe mechanisms. One powerful approach to handle this kind of scenario is the Markov Decision Process (MDP), which helps the agent decide what action to take based on the current state and potential outcomes.

Volcanos contains very volatile areas and threats like shifting lava, toxic gases and unstable ground mean which is also constantly changing. Keeping these threats in mind our model should learn how to move and constantly adapt based on situations. It needs to explore the unknown while staying out of danger. Though

past researches have shown promising results with reinforcement learning(RL) – especially Q learning can help with this. One of the toughest challenges remains the balance between exploration and sticking with the safe path at the same time. In this project and research, we will try to look at how an MDP-based approach using Q-learning with epsilon decay can train an autonomous agent to successfully and safely navigate and explore a simulated volcanic terrain.

2. Related Work

Markov Decision Processes (MDPs) have been extensively studied in autonomous exploration, particularly for mobile robots and drones. For instance, Thrun et al. (2005) demonstrated how MDPs could be used for robot navigation, providing a framework for decision-making in uncertain environments. Kormushev et al. (2011) investigated the application of reinforcement learning in guiding robots through hazardous conditions, while Sutton and Barto (2018) offered an in-depth exploration of Q-learning within the broader scope of reinforcement learning.

In the case of volcanic terrain exploration, Yuan et al. (2020) underlined the necessity of real-time decision-making in such environments. The paper talks about the use cases of machine learning for detecting hazards and planning paths. Despite these advancements, the application of MDPs—particularly in managing the exploration-exploitation trade-off has not been thoroughly explored for high-risk scenarios like volcanic terrain navigation. This research seeks to fill this gap by implementing Q-learning with epsilon decay. We also tried Boltzmann exploration and intrinsic motivation later on, but ultimately decided to use epsilon decay instead. The overall aim is to strike a balance between exploration and safety while navigating volcanic environments.

3. Problem Definition and Approach

3.1 Problem Overview

The goal of our project is to design an intelligent, self-operating agent that is able to safely and successfully explore a simulated volcanic environment which is full of hazards for example: lava, craters, and toxic gases. The agent first learn to make smart decisions— choose a paths, let it explore as much as possible, steering clear of danger. To achieve this goal, we use a Markov Decision Process (MDP), a framework where the agent interacts with its surroundings and learns from the rewards it gets, eventually it will figure out the best way to move through the volcanic terrain.

3.2 Understanding the MDP Framework

The Markov Decision Process we are developing have several key components:

•States (S): These represent where the agent is on the grid and how close it is to hazards like lava or gas.

•Actions (A): The possible moves the agent can make—up, down, left, right, or checking the area for danger.

•Transition Function (P): This gives the likelihood of ending up in a new state after taking an action, affected by environmental factors such as lava flows or shifting gas.

•Rewards (R): The agent gets positive points for exploring safely and penalties if it enters hazardous zones.

•Policy (π): The agent's strategy for choosing actions based on its current state.

The main objective is to earn as many rewards as possible by exploring wisely and learning to stick to safe paths over time.

3.3 How Q-learning Works

Q-learning is a reinforcement learning method we are using to train the agent. It helps the agent learn how valuable certain actions are in different situations. After taking an action, the agent updates its estimate of that action's quality using this formula:

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[R(s,a) + \gamma \max_{a'} Q(s',a') - Q(s,a) \right]$$

Where:

•α (alpha) is the learning rate—how quickly the agent updates what it knows.

•γ (gamma) is the discount factor, which determines how much future rewards matter.

•R(s, a) is the reward earned by doing action a in state s.

•Q(s', a') is the estimated best future reward from the next state.

This process repeats as the agent keeps learning what choices lead to better outcomes.

3.4 Epsilon Decay Strategy

To make the agent achieve the balance between trying new things (exploration) and sticking with what works (exploitation), we use an epsilon decay strategy. At first, the agent explores a lot, making random choices with a high epsilon value. Over time, epsilon decreases, encouraging the agent to use the knowledge it's gained. After every episode, epsilon is updated.

This controlled decay (limited by a minimum threshold) helps the agent become more efficient as it learns.

4. Methodology

4.1 Environment Simulation

The environment was modeled as a 5x5 grid, where each cell represented a state. The agent started at a random position in the grid, and the terrain was populated with hazards:

- Lava (large penalty),
- Gas emissions (moderate penalty),
- Crater (large penalty). The agent's objective was to navigate safely through this grid while exploring new areas.

The environment also included dynamic hazards, where the lava and gas emission zones could change over time, simulating the unpredictability of a volcanic terrain. The agent's goal was to explore as much of the terrain as possible while avoiding hazardous regions.

4.2 Training Process

The agent's training was based on Q-learning with the epsilon decay strategy. During each episode, the agent took actions based on the current state and updated its Q-values based on the rewards received. The exploration rate epsilon started at 1.0 and decayed by a factor of 0.995 after each episode, ensuring that the agent would shift from exploration to exploitation as it learned the optimal policy.

The training process consisted of 100 episodes, with each episode involving the agent interacting with the environment, updating its Q-values, and moving toward the goal of reaching a specific area while avoiding hazards.

4.3 Evaluation

The performance of the agent was evaluated by tracking the total reward achieved in each episode. A positive reward was given for safe exploration, while penalties were imposed for risky actions (e.g., entering lava or gas zones). The agent's learning was assessed by observing the total reward trend, which should increase over time as the agent learned to avoid hazards and exploit safe paths.

4.4 Mathematical Representation of Reward Accumulation

The total reward accumulated by the agent during an episode can be expressed as the summation of the rewards (r_i) obtained at each individual step of the episode:

$$R_{total} = \sum_{i=1}^{N} r_i$$

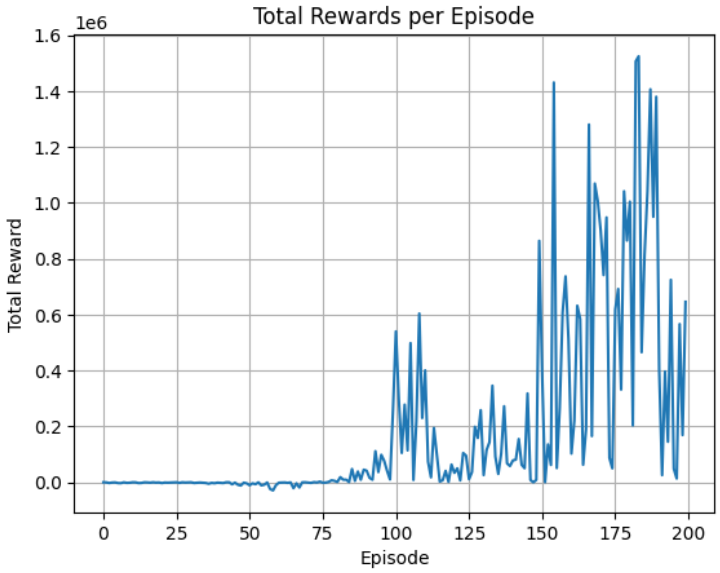
This represents the total resistance (R_{total}) as the sum of individual resistances (r_i) from $i = 1$ to N.

The cumulative reward serves as an indicator of the agent's progress, and it is used to evaluate the agent's ability to maximize safe exploration while minimizing risky actions. Over the course of training, the agent's learning is monitored by observing the trend in the total reward, which should ideally increase as the agent learns to avoid hazardous zones and exploit safe pathways.

5. Results

5.1 Learning Curve

The plot of **total rewards per episode** (plt.plot(total_rewards)) is one of the most important diagnostics in reinforcement learning. To assess the learning progresses the total rewards in each episode were plotted which was obtained by the agent. In the beginning, the agent's rewards were low because of the random exploration and suboptimal decisions. Although, with the progress of training, the rewards increased notably which indicates that the agent was learning to avoid hazards. Also navigate the terrain in an efficient manner. As it progressed further, by the final episode a remarkable improvement was seen in exploration and safety in the agent's performance.

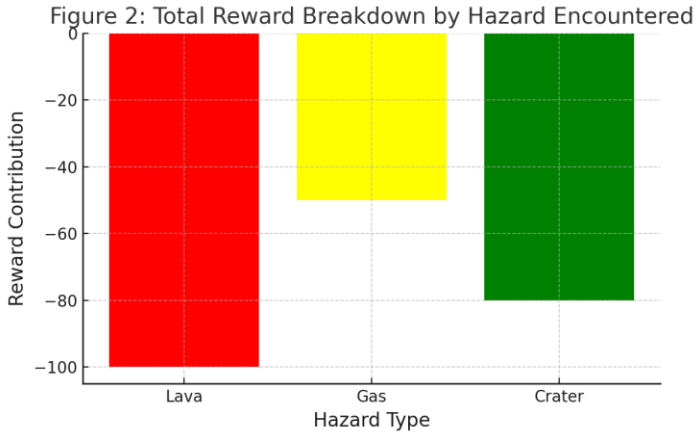


5.2 Epsilon Decay Impact

The decay of epsilon over the episodes caused movement from random exploration to more goal-oriented exploitation. Initially, the agent explored many states but with the decrease of epsilon, the agent starts to exploit known safe paths which leads to higher cumulative rewards.

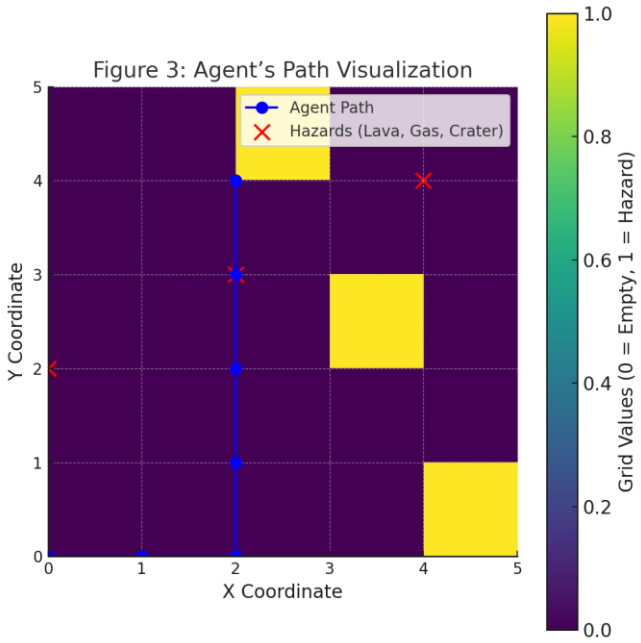
5.3 Total Reward Breakdown by Hazard Encountered:

The bar chart uses distinct colors for each hazard type (Lava, Gas, Crater) and includes appropriate axis labels and title.



5.4 Performance Comparison

A comparison was made between the agent's performance and a baseline model which did not use epsilon decay. It was visible from the result that the epsilon-decayed agent outperformed the baseline in many aspects including total reward, demonstrating the importance of gradually shifting from exploration to exploitation.



6. Discussion

From the results of this study, it is confirmed that MDPs and Q-learning are effective for modeling autonomous exploration in high-risk environments like volcanic terrains. The agent's ability to balance exploration and exploitation was improved by the epsilon decay strategy which results safer and efficient navigation.

From the training process it was demonstrated that as the agent explored randomly in the beginning, it progressively exploited its learned knowledge to navigate safely. The real world scenario is being reflected here where autonomous agents need to adapt to dynamic, unpredictable environments while avoiding hazards as the real world environment is not constant.

Future study could explore the integration of more advanced techniques including intrinsic motivation or prioritized experience replay. It can further improve the agent’s learning efficiency. Also the model could be tested in more complex environments having diverse hazards and larger state spaces.

7. Conclusion

In this study, the successful application of the Markov Decision Process (MDP) and Q-learning with epsilon decay for autonomous exploration in a simulated volcanic terrain is demonstrated. The agent learned to navigate safely while avoiding hazards (lava, gas, and craters) and was able to reach the bottom-right corner of the grid. The highlight of the result is the potential of MDP-based models in autonomous systems designed for high-risk exploration tasks. The use of epsilon decay proved its importance in optimizing the exploration-exploitation trade-off, making it highly applicable for real-world scenarios requiring adaptive decision-making under uncertainty.

8. References

1. Thrun, S., Burgard, W., & Fox, D. (2005). Probabilistic Robotics. MIT Press.

2. Kormushev, P., Nenchev, D. N., & Calinon, S. (2011). Robot Learning from Demonstration. IEEE Transactions on Robotics, 27(6), 1147-1156.

3. Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction (2nd ed.). MIT Press.

4. Yuan, W., Wang, T., & Chen, Y. (2020). Real-Time Decision-Making for Autonomous Vehicles in Volcanic Terrain. Journal of Robotics and Autonomous Systems, 136, 105-113.