# CS 370 Project Two

Pathfinding and the Treasure Hunt Game

CS-370-X6160 Current/Emerging Trends in CS 23EW6

Scott Dixon

August 10, 2023

**Differences between human and machine approaches to problem solving:**

Reinforcement learning (RL) is a machine learning approach based on rewarding desired behaviors and punishing undesired ones. In RL, an agent interacts with an environment by taking actions and receiving rewards or penalties as feedback. The goal is to learn a policy that maximizes cumulative long-term rewards. Unlike supervised learning, the agent must discover optimal actions through trial-and-error, without explicit training examples.

RL problems involve sequential decision making with a randomized decision at each timestep, the agent selects an action based on the current state, transitions to a new state, and receives a corresponding reward. However, the effects of actions are uncertain, and rewards may be delayed. Similarities between the human and machine approaches include the goal of reaching the treasure and the concept of trial and error. Both approaches aim to find an optimal path by exploring different routes. However, the machine approach uses a more systematic method, relying on data and neural networks to iteratively improve its decision-making, while the human approach involves cognitive reasoning and real-time adaptation.

Finding an optimal behavioral policy poses a complex challenge that is inherent in the design of a machine learning algorithm. Human learning, by contrast, utilizes cognitive skills and human senses along with intuition and memory of past experiences. This is quite a difference when comparing a learning algorithm of a computer to that of the learning that takes place in the human mind.

Pathfinding problems like maze navigation are a good example of sequential nodes or branches with random decision state. In the case of reinforcement learning the agent must learn to navigate by trying different actions and remembering which lead closer to the goal. The reward will guide the agent towards the optimal path, but those rewards would be sparse in complex mazes. Manually engineering a policy with human-like feedback systems would be impossible since, as the author describes “for any new task that we may wish to learn, it poses a significant bottleneck to widespread adoption of reinforcement learning.” (Singh, 2019)

Algorithms like deep Q-learning are well-suited for such problems. They combine deep neural networks with Q-learning, an RL technique based on learning action values called Q-values. Deep neural networks allow the algorithm to approximate complex Q-functions and solve problems with large state spaces like mazes. Deep Q-learning optimizes policies by maximizing predicted Q-values through trial-and-error experience.

This provides a powerful way to tackle real-world problems involving uncertain, delayed rewards. Examples beyond navigation include game playing, robotics, finance. Deep Q-learning exemplifies how AI advancements enable autonomous agents to excel in complex, dynamic environments.

**Steps a human would take to solve this maze:**

1. Visually inspect the maze to get an overall perspective of the layout
2. Use spatial reasoning abilities to map connections between paths
3. Logically think through different route options based on current location
4. Plan a route by visualizing and remembering paths that lead closer to the goal
5. Iterate through trial-and-error, tracking dead-ends and re-routing
6. Leverage visual memory to avoid re-trying failed options
7. Adapt strategy incrementally based on experience in the maze

**Steps the intelligent agent is taking to solve the pathfinding maze:**

1. Initialize the maze environment and neural network model.
2. Choose a random starting cell for the agent.
3. Observe the environment and select an action using the RL strategy.
4. Perform the chosen action and receive feedback (reward, game status).
5. Store the experience in memory for experience replay.
6. Train the neural network using experiences from memory.
7. Adjust epsilon for exploration-exploitation balance.
8. Monitor win rate and training progress.
9. Terminate training upon reaching win rate or epoch threshold.

**Similarities and differences between human and machine learning:**

Similarities:

* Both are attempting to find the optimal path through the maze to reach the goal.
* Both rely on some amount of trial-and-error, adjusting their path based on feedback.
* Both leverage experience in the maze to inform future decisions.
* Both aim to solve the maze in an efficient manner.

Differences:

* Humans use visual inspection, spatial reasoning, intuition and planning. The agent relies on computational algorithms and neural networks.
* Humans make real-time deductions and alterations as they traverse the maze. The agent follows a systematic training process across iterations.
* The human approach is more flexible and improvisational. The agent's approach is more rigid but can handle larger and more complex mazes.
* The human leverages memory to avoid repeat mistakes. The agent uses experience replay.
* The human mind has a global perspective of the full maze. The agent only observes its local space in the maze.
* The human solution is more difficult to explicitly define. The agent's policy is a constrained, trained function.

**Intelligent agent’s purpose in the pathfinding maze:**

The intelligent agent's purpose is to optimize efficient navigation in the maze by maximizing rewards through iterative deep Q-learning. It should balance exploitation and exploration to continuously improve its policy based on experience. The agent's goal is to autonomously move through the maze to find the most efficient path to the treasure. Throughout the process the agent evaluates its progress based on metrics like win and loss rate and cumulative long-term rewards.

**Exploitation Exploration and the ideal proportion of the two:**

Exploitation refers to choosing actions based on the agent's current knowledge, while exploration involves trying new actions to discover potentially better strategies. “Any efficient search algorithm must use two general techniques to find the global maximum: exploration to investigate points in new and unknown regions of the search space and exploitation to make use of knowledge found at points previously visited to help find better points.” (Goshen & Shimshoni, 2008). The ideal proportion of exploitation and exploration depends on the stage of learning. Initially, the agent benefits from more exploration to discover diverse paths. As learning progresses, a shift towards exploitation is preferable to maximize the acquired knowledge.

**The use of algorithms to solve complex problems:**

For this maze pathfinding problem, an 80/20 proportion is reasonable. The agent should explore enough to find possible connections and avoid getting trapped. But the focus should be exploitation to determine the precise optimal path. Too much exploration will slow learning.

In the provided code, the agent balances exploitation and exploration using an epsilon-greedy strategy. Deep Q-learning is implemented using a neural network model to approximate Q-values. The agent iteratively learns by interacting with the environment, choosing actions based on the algorithm, and updating its knowledge through experience replay.

References:

<https://bair.berkeley.edu/blog/2019/05/28/end-to-end/>

Goshen, L., & Shimshoni, H. (2008). Balanced Exploration and Exploitation Model Search for Efficient Epipolar Geometry Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, Pattern Analysis and Machine Intelligence, IEEE Transactions on, IEEE Trans. Pattern Anal. Mach. Intell*, *30*(7), 1230–1242. <https://doi:>org.ezproxy.snhu.edu/10.1109/TPAMI.2007.70768