**7-3 Project Two: Design Defense**

Braedan French

Department of Computer Science, Southern New Hampshire University

CS-370-16865: Current/Emerging Trends in CS

Mike Prasad

April 17, 2025

**Analyze Human vs. Machine Intelligence**

Placed inside a physical maze scenario, a human being would use exploration, memorization, and problem-solving to find the exit as quickly as possible. One approach would be to eliminate possible dead-end routes through exploration and commit them to memory. Once memorized, a human could avoid taking those routes again to meet dead ends that they had previously come across. Looking for memorable features or landmarks within a maze is another way that memorization can be used to mark pathways that lead to further progression or a dead end. Essentially, taking mental notes of previously traveled routes and visual cues can help a human find the exit to a maze. A mental map of sorts that gets created the longer exploration occurs and benefits from a human’s aptitude for recognition through memorization. With an intelligent agent, like the pirate agent from this project’s pathfinding problem, the approach is much the same as the one described above. Each attempt made by the agent is random exploration with its knowledge of the maze improving as it expands its memory. This is achieved through a learning rate decay function. A learning rate decay function helps an agent in training learn quickly in the beginning but progressively slows the learning process over time. This is because “Starting with a high learning rate, and the model might learn quickly, but it can overshoot and miss the best solution. Start too low, and it might be too slow or get stuck” (GeeksforGeeks, 2023). So instead of keeping the learning rate constant and running into these possible issues, learning rate decay was employed. The agent would then randomize routes quickly in the beginning epochs and take longer consideration about the correct move in the later epochs. As mentioned, a similarity between a human and the pirate agent is the approach of taking random pathways and committing previously traveled routes to memory to begin recognizing the correct path. The differences between a human and the pirate agent become apparent quickly as the agent lacks visibility while a human does. An example of this would be that in a maze setting, a human would be capable of seeing the walls that form the maze’s pathways. They would then be able to move about the maze without worrying about running into a wall, backtracking where necessary when a dead end is within sight. The agent on the other hand is essentially blind until it fully learns the borders and obstacles within the maze. It would not be able to avoid these obstacles until it had already come across them and failed to move through them. Only then would the agent be able to recognize which tiles are traversable and which are impassible as it commits this information to memory over the course of many epochs. This occurs because the Deep Q-Learning algorithm enabling reinforcement learning within the pirate agent is model-free. This allows the agent to try different possibilities of exploration within the maze without the knowledge of what it looks like.

**Purpose of Intelligent Agent**

Within reinforcement learning, exploration is described as the possible actions an agent can take within a scenario that allows it to learn about its surrounding environment. Exploitation is then how the agent can utilize the knowledge that it has learned to maximize the rewards it receives. A bias towards one or the other can have unintended consequences during an agent’s reinforcement learning process. “If the agent continues to exploit only past experiences, it is likely to get stuck in a suboptimal policy. On the other hand, if it continues to explore without exploiting, it might never find a good policy” (Scribbr, n.d.). As mentioned above, the pirate agent within this pathfinding scenario utilized model-free reinforcement learning techniques. Therefore, the agent has no initial knowledge about the structure of the maze. This only allows the agent to learn by making random moves in any of the possible directions one step at a time. To quickly learn the surrounding environment, the agent would need to focus more on exploration by trying random actions and committing them to memory. Once the agent has a better grasp on how to properly traverse the maze, it can focus more on exploitation and how it can optimize the solution of reaching the end for maximum reward. Looking back, this is another reason why the learning rate decay function was utilized. As the benefits of the learning rate decay mirror that of balancing exploration and exploitation within an agent’s reinforcement learning. A higher learning rate being the equivalent to the effects of exploration where the agent moves quickly with its training, trying every possibility for maximum efficiency. And a lower learning rate is like exploitation since it slows down the process, allowing the agent to take greater consideration on how to arrive at the optimal solution at each step. As for how reinforcement learning can help in determine the path to the end of the maze, the game follows the systems of reinforcement learning. The pirate agent interacts with the environment, that being the maze, by moving left, right, forward, and backward. If the action taken by the agent moves towards a favorable state, this being a traversable tile, then it is rewarded points and continues the game. If an undesired state is reached by the agent either becoming stuck or running into an obstacle, it is penalized and sent back to the starting tile. This way, the agent can begin to memorize its surroundings and determine which actions lead to favorable outcomes and avoid others. Over time, the agent’s policy will update with the knowledge it gains about the maze environment towards the goal of maximizing rewards by reaching the treasure. Once the policy is fully optimized, the agent can take the most direct route to the treasure and bring its win rate to 100%.

**Evaluating Algorithms**

Deep Q-Learning was used for committing the state-action pairs to the pirate agent’s memory. Depending on the resulting state from an action, the corresponding rewards or penalties were given so that the agent would determine the optimal policy over time. This process was further enhanced with the learning rate decay function, which guided the agent’s actions from exploration early on to exploitation towards the end. This is because the maze’s structure was not known to the agent from the start as the Deep Q-Learning network is model-free. As mentioned previously, a model-free network does not provide all the information or rules of the game to the agent, only what possible actions can be taken from the start. Therefore, exploration helped the agent to learn the surrounding maze environment quickly before accumulating enough knowledge to utilize exploitation. After switching to exploitation, the agent was able to achieve the optimal policy and find a direct path to the treasure at the end of the maze with 100% consistency by the 626th epoch.

**References**

Amin, S. (2024). *Deep Q-Learning (DQN)*. Medium. <https://medium.com/@samina.amin/deep-q-learning-dqn-71c109586bae>

GeeksforGeeks. (2023, November 4). *Learning Rate Decay*. GeeksforGeeks; GeeksforGeeks, Sanchhaya Education Private Limited. <https://www.geeksforgeeks.org/learning-rate-decay/>

Scribbr. (n.d.). *What is the exploration vs exploitation trade off in reinforcement learning?* Scribbr. Retrieved April 17, 2025, from <https://www.scribbr.com/frequently-asked-questions/what-is-the-exploration-vs-exploitation-trade-off-in-reinforcement-learning/>