Pacman Game AI Project Proposal

CSC584 - Building Game AI

Dev Patel (dmpatel3), Aditya Karthikeyan (akarthi3), and Jay Joshi (jmjoshi)

Problem Summary

In the game Pacman, the player plays as the main character Pacman, who has the ability to consume dots that are placed throughout a maze. While trying to consume all the dots in a maze, ghost opponents are also present and have the ability to kill Pacman if they touch him, so the player has to also worry about dodging the ghosts while eating the dots. Even though the ghosts can kill him, Pacman does have the ability to defeat the ghosts if he consumes a power pellet placed in the maze. Overall, the main objective of the game is to consume all the dots within the given maze/level while dodging all the ghosts in the level in order to avoid getting killed by the ghosts and beat the game. Even though the ghosts are enemies controlled by AI aimed at getting the player, the game doesn't have an actual opponent competing against the actual player in terms of getting all the dots in the level. With this project, the project focuses on creating an AI player that acts as an opponent for the main player, with both of them competing for consuming the most dots in the level as a way of introducing a Player vs. AI concept into the Pacman game.

Problem Environment

The Pacman game was made in Python, and the existing code for the game has implementation for the main Pacman player and behavior for consuming the dots in the level along with the AI behavior of the enemy ghosts. Using the existing Pacman game code in Python, the project will primarily focus on adding additional code for adding an opponent player in the game in addition to adding code for implementing the AI behavior of the opponent player. In addition, even though the ghosts have random wandering movement in the game, the AI used for the ghosts themselves isn't much and doesn't really make decisions on what path to take as the ghosts only decide what path to take once they've collided with a wall in the maze. Because of this, the AI used for the ghosts has the possibility for further improvements to help make the movement of the ghosts more unpredictable and perhaps more challenging for the player, which could be achieved through using path-finding algorithms. Additionally, code could also be added so that as the player beats the level each time, the AI behavior of the ghosts and opponent player increases and becomes more intense as a way of slowly increasing the game's difficulty with each playthrough the player experiences.

Project Research

$(CodeLink : https : //grantjenks.com/docs/freegames/\#pacman)$

The base game used for our project is from a licensed collection of free Python games from Giant-Jenks intended for education and fun. We also looked into various papers and articles including works from UC Berkeley that investigate the performances and impact of several AI techniques and algorithms on the game environment of Pacman, which included the A* pathfinding algorithm and model-based reinforcement learning algorithm. The articles also mentioned working on searching in a multi-agent environment, which is something we are looking into when we let an AI opponent play alongside a human user.

AI Implementation Approach

Since the AI opponent player is meant to play similar to an actual human player in the game, the AI opponent will need to have behavior for moving around in the maze along with consuming the dots in the level. The consuming dots behavior for the AI opponent will be the same as the existing consuming dots behavior for the main player, but the behavior for both players will have to be modified due to the desired competitive nature with the AI opponent and human player. Since the main player and AI opponent are meant to be competing to see which one can consume the most dots in the level before all have been consumed, there will have to be additional trackers for the number of dots consumed by each player respectively along with the number of existing dots in the maze. For handling the movement behavior of the AI opponent player, a combination of the A* search and deep reinforcement learning algorithms could be used to help the AI opponent determine what paths to take in the maze to obtain the most dots possible as a way of replicating the average playstyle of a human player in the game. In a similar fashion, the A* search and deep reinforcement learning algorithms could also be used with the AI ghosts to enhance their preexisting movement behavior in the game.

AI Evaluation Methods

Since the AI opponent is meant to play similar to an actual human player, the AI opponent will mainly be trying to consume as many dots as possible and also try to avoid getting hit by any of the ghosts in the level. Because the AI opponent has similar objectives with the human player, some metrics that could be used for measuring the AI's performance could be the number of consumed dots and number of deaths. Because the AI opponent is trying to consume more dots compared to the actual player, measuring the number of consumed dots can help determine if the AI is getting about the same number of dots as the main player, or higher/lower than the main player's amount of consumed dots. To also help with this, measuring the total time for completing the level by consuming all the dots in a single-agent environment could be a useful metric for determining how well the AI performs when it's playing by itself as a way of indicating its' performance compared to an average player's performance in the game. Furthermore, measuring the number of victories against a human opponent over a set number of trials could be another metric used for determining how well the AI opponent does against a human player. In addition, measuring the number of deaths helps indicate how often the AI opponent is hitting the ghosts and therefore help determine if the AI can easily adapt well to the movement of the ghosts or if it struggles with doing so. Overall, using these metrics can provide useful insight into the AI's performance compared with an average player's performance in the game, which can be helpful for determining what needs to be changed with the AI behavior in order to get the AI player to have an equal/similar performance to an average player in the game.

Project Significance

By adding an AI opponent player into the game, the project aims to achieve an enhanced single-player experience for the Pacman game by adding challenge for the main player through the AI opponent. Furthermore, the Pacman game environment is a multi-agent, dynamic and stochastic environment, so our solution to this problem might have applications to other domains such as autonomous vehicles, robotics or other video games consisting of similar environments.