Project Evaluation Report

Artificial Intelligence for Games

Academy of Interactive Entertainment

Bonface Njuguna, June 2022

# Project Overview

The project is about Pathfinding and Decision making techniques.

The pathfinding project involves an agent that can move from one location to another while avoiding obstacles.

The Decision making AI is based on two agents, enemy and player. The enemy wanders around and chases the player when within range, the player is moved around with the keyboard and auto switches to flee state when the enemy is within range.

The AI algorithm used in the project was finite state machines for Decision making project and Dijkstra’s shortest path algorithm for the pathfinding project

The pathfinding algorithm was a bit difficult to understand all the details well, also it was a bit challenging to test how the algorithm works.

The FSM algorithm was also challenging because the enemy agent was getting incorrect agent behaviour when testing. It was also challenging to figure out the current states of the agents and the transition from one state to the other

The project took longer than expected to complete due to some challenges faced while studying but I was able to implement the techniques needed. These challenges are internet connection issues, the recent pandemic and my general knowledge of programming was still very low as compared to where I am now.

# Performance Analysis

Pathfinding

Dijkstra’s pathfinding is the one of the best for finding shortest path but its limitations include:

* It is not fast and consumes resources because it does a blind search
* It does not handle negative edges.
* Setting up the map is difficult and can cause issues when loading the map. The map has to be set correctly for pathfinding to work
* It is slow to do pathfinding when having multiple agents because the algorithm is not fast enough and has to do pathing for all the agents.

FSM

Finite state machines is a good algorithm for decision making techniques but it has some cons which include:

* When having a lot of agents we start experiencing performance issue
* To improve this maybe try and use decision trees instead.
* Setting up the state machine quite hard because one has to figure out the states, condition and the transitions of the agents which can be challenging.

# Future Improvements

Due to time and other personal issues I was not able to implement the AI as planned, I was to use the decision and behaviour trees but it was easier to implement FSM.

The FSM I used had two states for each agent. I plan to have more states for the agents which can be very interesting to test.

The Dijkstra’s algorithm is really a good one and I plan to reuse it in the future projects. Dijkstra’s algorithm is simple to implement compared to other algorithms. I plan to implement multithreading which will help support more agents. I also plan to use priority queue instead of lists of nodes to improve the speed of the program.

I would use another algorithm because Dijkstra’s take more time to do the search.