Report: Playing a Snake Game Using Search Algorithms

By:

Akmal bin Mohamed Ikraam (18010157)
Tan Pee Aun (17037177)
Tan Li Siang (17103060)
Daniesha Jayasinghe (16044760)

Introduction

Snake is a popular game that has existed for more than 20 years and was first introduced to the public on old mobile devices. Now it has become one of the most popular games available. Players control a snake using movement keys to grab food while avoiding walls and itself. While the game itself is not too challenging and is accessible to almost everyone, it is yet to be determined whether an AI can play the game as well as, if not surpass, a human can.

Problem Description And Formulation

The aim of this test is to create an AI that can play the Snake game in a reliable manner. The AI is considered reliable when it fulfills the following conditions:

- The AI is able to find the food (goal)
    - The most basic function that the AI should be able to perform is to reach its goal regardless of the total cost path. As long as the AI is able to reach the goal, further improvements to the pathing logic can be implemented.
- The AI does not collide with the game boundaries while searching
- The AI does not collide with itself while searching
- The AI does not perform any illegal moves i.e anything that a human player could not perform under the same circumstances

Algorithm Implementation

The AI for the Snake game should be able to create a path towards the goal, regardless of its starting position or goal position. This can be done using a search algorithm. Depending on the implementation, the time taken to reach the goal could differ. Search algorithm to be used for this experiment is the Tree Search Algorithm. The two variants used will be the “Informed

Search" and "Uninformed Search". "Informed Search" lets the AI know where the goal is so that it can create a path beforehand. "Uninformed Search" forces the AI to brute force its way through the entire map to look for the goal. It is expected that due to the nature of "Uninformed Search", the time taken to reach the goal will be significantly higher compared to "Informed Search".

For ease of implementation, the following rules has been implemented for the search algorithm

- The check for whether the search is an "Informed Search" is toggleable with a boolean. Setting it to false will trigger an "Uninformed Search"
- Boundaries were also placed in the "Visited portion of the algorithm to prevent any collision issues.
- For "Uninformed Search" path taken is chosen at random with respect to current direction.
- To prevent illegal moves, paths opposite of the current direction are ignored when performing searches (e.g if the snake is facing north, then the "south" node is ignored during node selection). Paths containing the body of the snake are also excluded from node selection to prevent collision issues.
- All nodes on the map will have a cost of 1. Therefore, the cost of the path is directly proportional to the number of nodes visited during the search. AI will prioritise the shortest path to the goal based on the cost of the path.

Algorithm Results

Informed Search

For "Informed Search", the time taken to reach the goal is faster when compared to an Uninformed Search approach. The time taken to reach the goal is also much longer as the snake grows longer with each additional score since the AI always attempts to avoid its body whenever possible. There is no difference in time taken to reach the goal when the map size is increased. However, there is a chance for the AI to fail the longer the snake grows. AI will occasionally collide into itself if it grows to more than a length of 3.

Uninformed Search

The time taken for an "Uninformed Search" is significantly longer as the AI attempts to reach the goal in a random manner. Occasionally, the AI will fail as it enters an empty node, but the neighboring nodes are considered as "visited", thus freezing the AI. Length of the snake does not seem to affect the time taken to reach the goal very much as the AI still randomly tries to find the goal while avoiding its own body. The time taken for the snake to reach the goal increases as the map size increases due to the larger node space that it has to explore to find the goal. AI rarely collides with itself, due to the random nature of its traversal. However, the chances of failure is higher due to it being "stuck" due to running out of "free" nodes to travel through.

Performance Evaluation and Possible Improvements

Evaluation

Search Algorithm Comparisons

Based on the search types used, the “Informed Search” has a more consistent chance of reaching the goal compared to an “Uninformed Search”. However, the snake’s movement, when the body grows with each goal reached, for the “Informed Search” and “Uninformed Search” does take into account the body movements when deciding on the best path, however after a certain number of food consumed and the snake’s length has grown, the algorithm is proved to be not so efficient in calculating its body position, as the snake is only able to perceive one node tile at a time. As a result, the snake will occasionally crash into itself when attempting to take what is considered the best path, especially if the food appears to be on the same row as the snake’s head position. This is because the snake only considers the initial position of its body before deciding on a path to take.

As mentioned in the results, the time taken for the “informed Search” algorithm is much faster due to having predetermined knowledge of where the goal is, and therefore, is able to create a path directly to the goal. “Uninformed Search” takes longer due to having to brute force its way towards the goal as it has no knowledge of the goal.

Meanwhile, the shortcomings of the “Uninformed Search” may be due to its implementation, since it uses the Breadth First Search Algorithm to try to find the goal. Also, due to the random nature of node selection in this method, the consistency of the method drops significantly since it will not visit any nodes marked as “visited” to avoid any potential looping. This occasionally results in the AI pathing itself into a dead end.

Performance Against A Human Player

Due to the different playstyles humans tend to have, it is hard to determine the efficiency of the AI. However, it can be surmised that humans will usually head straight towards the goal, similar to how an “Informed Search” will perform its pathing. As such, it is feasible to say that both humans and AI can work in a similar fashion. That said, the AI has the advantage of knowing the shortest path while a human will only usually take what they perceive is the shortest path. The AI also has more precise reaction times if the game speed was to be increased during the gameplay. As such, in a competitive scene, the AI with the “Informed Search” trait is technically more efficient than a human in that scenario. Despite that, humans can adapt, and therefore change its playstyle to fit the situation, while AI follows a set of predetermined logic, and is therefore predictable. As the game progresses, humans should have the better advantage due to their ability to adapt to the situation.

Possible Improvements

Additional logic and rule sets for Uninformed Search

The main weakness of an Uninformed Search is it's lack of knowledge of the goal, and is thus less efficient than its informed counterpart. Therefore, a potential improvement that can be done is to implement better search functions for "Uninformed Search". An example being the Greedy Breadth First Search, which should theoretically improve the pathing issues the current AI has.

Choosing "Longest Path" for when the snake grows

One of the biggest issues for both search algorithms is its insistence of choosing the shortest path to the goal regardless of how future states will play out. As such, a possible improvement that can be done to fix this issue is to change the priority to taking the longest path when the snake reaches a certain length. This ensures consistency in reaching the goal without the risk of collision at the cost of increased time taken to reach the goal.

Conclusion

This experiment has shown that it is possible to create an AI to play the snake game. An AI with an "Informed Search" method is objectively better at reaching the goal for the game, thus fulfilling the conditions set at the start of the experiment. However, the consistency for reaching the goal decreases as the game progresses into a much longer phase due to the simplicity of the logic used in its search algorithm. As such, to ensure a more consistent AI agent for future experiments, additional rules and logic should be considered when implementing search algorithms.