Progress Report

Revised Project Proposal: Heuristic Search for Puzzle Solving

Submitted by: Sumel Rattan, AI-550
Date: 10/31/2024

1. Introduction: The objective of this project is to compare search-based algorithms to determine the most efficient strategies for solving specific puzzles and games, focusing on A*, Uniform Cost Search, and Greedy Best-First Search. This project will analyze how different heuristics—both simple and domain-specific—affect algorithm performance. The evaluation metrics include speed, solution quality (accuracy), and computational resource usage (memory and processing time).

2. Problem Statement: Puzzle-solving using search-based algorithms poses challenges in selecting heuristics that balance search speed and solution quality without excessive computational costs. While simple heuristics, like Euclidean distance, offer quick computations, they may not always yield the most optimal solutions. In contrast, complex, domain-specific heuristics can provide better solutions but often require higher memory and processing power.

This project will focus on three selected games, each with a tailored set of heuristics to study the trade-offs between simplicity and complexity.

3. Games and Heuristics: I will concentrate on the following three games, each allowing for unique heuristic designs:

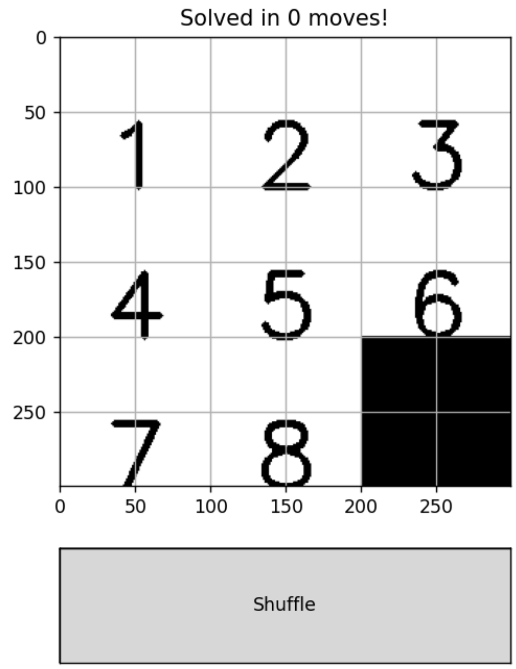
- **Sliding Puzzle (8-puzzle)**
    - **Heuristics**: Manhattan distance (simple) vs. Misplaced Tiles (more complex)
- **Maze Solving (grid-based)**
    - **Heuristics**: Euclidean distance (simple) vs. an advanced heuristic that factors in obstacle density and path complexity
- **Snake Game**
    - **Heuristics**: Distance to food (simple) vs. an extended heuristic that accounts for path safety and self-collision avoidance

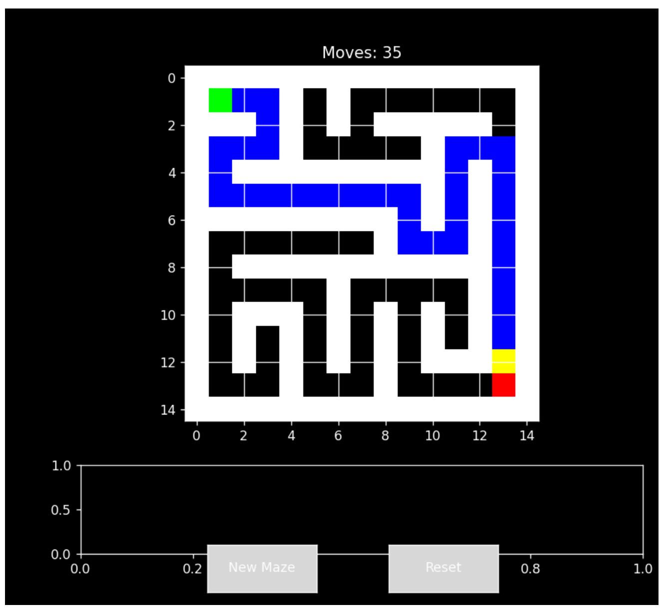
4. Algorithms:

- **A***: Combines the cost to reach the current node and the heuristic to evaluate the most promising path.
- **Uniform Cost Search**: Considers only the path cost without any heuristic, ensuring the lowest-cost path is prioritized.
- **Greedy Best-First Search**: Relies solely on the heuristic value to choose the next node, providing insight into heuristic influence on performance.

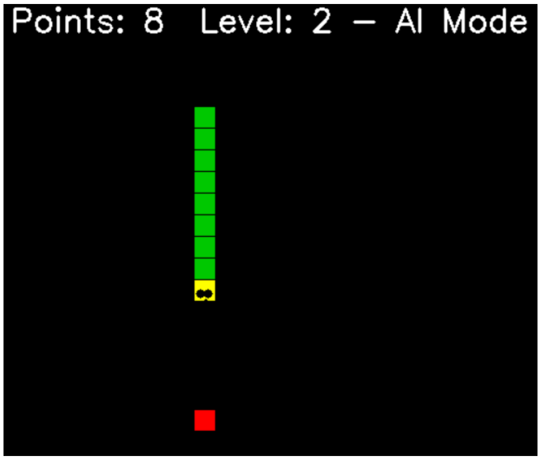
5. Implementation Progress: The current progress includes:

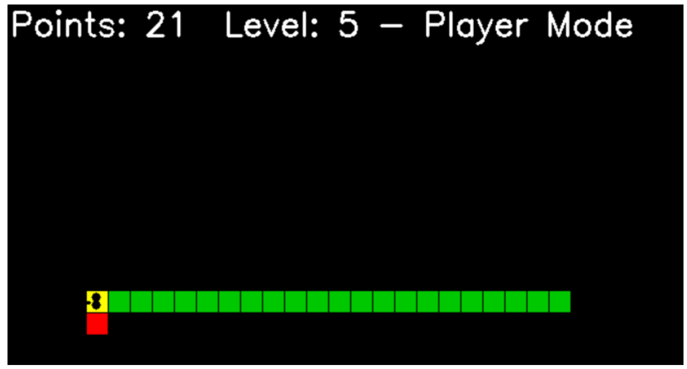
- **Sliding Puzzle and Maze Solving**: Both games are fully implemented with user input functionality.





- **Snake Game**: Includes an AI toggle *(t)* that switches between user input and **Greedy Best-First** Search-based gameplay.





Next Steps: I will implement the A* algorithm and Uniform Cost Search with the outlined heuristics for all game problems in the next and final phase of the project.

GitHub Repository: GitHub Repository

6. Hypothesis: I hypothesize that:

- A* will provide a balanced approach, excelling in both speed and solution quality due to its combined cost and heuristic consideration.
- Uniform Cost Search will be slower but will always find the least-cost solution due to its focus on path cost alone.
- Greedy Best-First Search will be fast but may sacrifice solution quality because it only uses heuristics.

7. Methodology:

- **Algorithm Implementation**: Implement A*, Uniform Cost Search, and Greedy Best-First Search for each game, testing each with simple and complex heuristics.
- **Performance Metrics**: The metrics for evaluating each game and algorithm combination are:
    - **Search Speed**: Time taken to find the solution.
    - **Solution Quality**: Optimality of the solution (e.g., number of moves or game success).
    - **Memory Usage**: Peak memory consumption during the search.
- **Benchmarking Environment**: All tests will be run under identical computational conditions to ensure consistent results.

8. Expected Outcomes:

- A* is expected to strike a balance between speed and solution quality.
- Uniform Cost Search will be optimal in finding the lowest-cost path but will be slower, especially in complex puzzles.
- Greedy Best-First Search will be faster but may not consistently produce optimal solutions, especially in more complex games.

9. Conclusion: This project aims to reveal the impact of heuristic design on the performance of search algorithms in puzzle-solving scenarios. By comparing A*, Uniform Cost Search, and Greedy Best-First Search across three distinct games, the project will uncover trends in speed, solution quality, and computational cost. These findings can be extended to other AI search problems where heuristic choice is crucial for performance optimization.