Title & Overview

Title:

AI Assignment 2: Search and Optimization

Subtitle:

Implementation and Performance Analysis of Search & Optimization Algorithms

Presented by:

Aman Kumar(CS24M114)

Shashi Mani(CS24M116)

Overview:

Implements search algorithms (Branch and Bound, IDA*) on the Frozen Lake environment.

Implements optimization algorithms (Hill Climbing, Simulated Annealing) for the Travelling Salesman Problem (TSP).

Compares performance based on cost, execution time, and convergence behavior.

Problem Statement & Environments

Problem Statement:

- Search Problem: Find an optimal path from the start to the goal in a grid (Frozen Lake) while avoiding holes.
- Optimization Problem: Find the shortest possible tour that visits every city once (TSP).

Environments:

❑ Frozen Lake:

- A grid where cells represent safe zones, holes, start (2), and goal (3).
- Uses Manhattan distance as a heuristic.

❑ TSP Instance:

- A set of randomly generated cities with Euclidean distances.
- The objective is to minimize the total tour length.

Algorithm Descriptions

Search Algorithms (Frozen Lake):

❑ Branch and Bound (BnB):

•Explores all paths systematically.

•Prunes paths using a heuristic (Manhattan distance) to avoid non-promising routes.

❑ Iterative Deepening A* (IDA):*

•Combines the benefits of iterative deepening and A* search.

•Uses a cost threshold (f = g + h) that is incrementally increased until the goal is found.

Optimization Algorithms (TSP):

❑ Hill Climbing (HC):

•A local search method that iteratively improves the current solution.

•May get stuck in local optima.

❑ Simulated Annealing (SA):

•Uses a probabilistic acceptance rule to allow worse solutions occasionally.

•Gradually reduces “temperature” to refine the solution and escape local minima.

Experimental Setup

Procedure:

•Trials: 5 runs for each algorithm.

•Timeout: Each run is terminated if it exceeds 10 minutes.

•Metrics Recorded:

✓ Final cost (reward or tour length).

✓ Number of iterations until convergence.

✓ Execution time per run.

Tools and Environment:

•Programming: Python 3 with NumPy and Matplotlib.

•Development: GitHub repository for code versioning.

•Data Output: Results are saved to data/results.csv and visualized with bar charts.

Results & Visualizations

Performance Metrics:

•Frozen Lake (Search Algorithms):

✓ BnB and IDA* both consistently found solutions with a cost of 6 in under 10 iterations and milliseconds of execution.

•TSP (Optimization Algorithms):

✓ Hill Climbing: Final tour costs ranged from ~400 to ~580 with 12–18 iterations.

✓ Simulated Annealing: More iterations (e.g., 2757) but with robust performance in escaping local optima.

Visuals:

•Bar Chart: Compare average execution times of all algorithms.

•Animated GIFs:

✓ Frozen Lake GIF: Demonstrate grid navigation, algorithm exploration, and final path visualization.

✓ TSP GIF: Show the evolving tour as the algorithm iterates, highlighting improvements in the tour configuration.

Conclusions & Future Work

Conclusions:

•Search Algorithms:

✓ Both Branch and Bound and IDA* reliably found optimal solutions quickly.

•Optimization Algorithms:

✓ Hill Climbing is faster but more susceptible to local optima.

✓ Simulated Annealing, while performing many iterations, provides a more robust search for the global optimum.

Future Work:

•Explore alternative heuristics and tuning parameters (e.g., cooling schedule for SA).

•Scale experiments to larger environments and more complex instances.

•Integrate additional optimization techniques to further improve solution quality.

Final Thoughts:

•This project demonstrates the trade-offs between exploration, exploitation, and computational efficiency in search and optimization problems.

•Open for questions and further discussion.