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:

•Objectives:

- ✓ Compare search algorithms on the Frozen Lake environment and optimization algorithms on the Travelling Salesman Problem (TSP).
- ✓ Record performance metrics: reward (solution cost/tour length), convergence (iterations), and execution time.

•Algorithms Implemented:

- ✓ Search: Branch and Bound (BnB) and Iterative Deepening A* (IDA*).
- ✓ Optimization: Hill Climbing (HC) and Simulated Annealing (SA).

•Visual Elements:

- ✓ Include relevant icons or images that symbolize grid search and TSP optimization..

Problem Statement & Environments

Problem Statement:

- Frozen Lake (Search Problem):

  Find the optimal path from the start to the goal in a grid-based environment while avoiding holes.

- TSP (Optimization Problem):

  Determine the shortest tour that visits each city exactly once and returns to the starting city.

Environments:

❑ Frozen Lake:

- Grid Details:

✓ 2: Start, 3: Goal, 0: Safe cells, 1: Holes.

- Performance Data :

✓ Reward (path cost) = 6 (optimal).

✓ Convergence: ~10 iterations (BnB) and ~9 iterations (IDA*).

✓ Execution Time: Ranges from ~0.00016 s to ~0.00055 s.TSP Instance:

☐ TSP Instance:

- Cities: Randomly generated points.

- Performance Data :

✓ Hill Climbing: Tour lengths vary (approx. 460–516), convergence in 13–16 iterations, execution time ~0.008–0.013 s.

✓ Simulated Annealing: Tour lengths vary (approx. 367–511) with a fixed convergence iteration count (2757, representing extensive exploration) and execution time ~0.019–0.021 s.

Algorithm Descriptions

Search Algorithms (Frozen Lake):

❑ Branch and Bound (BnB):

•Method: Systematic exploration with pruning using cost plus heuristic.

•Performance:

✓ Reward: Always finds optimal cost = 6.

✓ Convergence: ~10 iterations.

✓ Time: ~0.00025–0.00055 s.

❑ Iterative Deepening A (IDA):**

- Method: Depth-first search with iterative threshold increases, guided by heuristic.
- Performance:

✓ Reward: Also finds optimal cost = 6.

✓ Convergence: ~9 iterations.

✓ Time: ~0.00010–0.00023 s.

☐ **Heuristic Function (Frozen Lake):**

•Manhattan Distance:

$$h(x, y) = |x_{\text{current}} - x_{\text{goal}}| + |y_{\text{current}} - y_{\text{goal}}|$$

Optimization Algorithms (TSP):

❑Hill Climbing (HC):

- Method: Iteratively improves a random tour by swapping cities.
- Performance:

✓Reward: Tour lengths ranging approximately 460–516.

✓Convergence: Achieved in 13–16 iterations.

✓Time: ~0.009–0.013 s.

☐Simulated Annealing (SA):

- Method: Uses probabilistic acceptance of worse moves to escape local optima.
- Performance:

✓Reward: Tour lengths between ~367 and ~511.

✓Convergence: Recorded as 2757 iterations (reflecting extensive exploration).

✓Time: ~0.019–0.021 s.

Evaluation Metric (TSP):
Total Tour Length: Sum of Euclidean distances between consecutive cities (and back to the start).

Experimental Setup

Methodology:

•Trials: 5 runs for each algorithm.

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

•Metrics Recorded:

- Reward/Cost:

✓ Frozen Lake: Final path cost (optimal = 6).

✓ TSP: Total tour length (lower is better).

- Convergence Point:

✓ Number of iterations until the solution is stable.

- Execution Time:

✓ Recorded in seconds.

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.

•Experiment Script:

✓ The utils.py script runs all algorithms, consolidates results, and saves them to data/results.csv.

•Data Visualization:

✓ Bar charts and other graphs are generated from the CSV data.

Results & Visualizations

Summary of Experimental Results:

☐ Frozen Lake (Search Algorithms):

•BnB:

✓ Reward = 6; Convergence: 10 iterations; Average Time ≈ 0.00029 s.

• IDA:*

✓ Reward = 6; Convergence: 9 iterations; Average Time ≈ 0.00011 s.

☐ TSP (Optimization Algorithms):

• Hill Climbing:

✓ Tour lengths ~460–516; Convergence: 13–16 iterations; Average Time ≈ 0.010–0.013 s.

•Simulated Annealing:

✓ Tour lengths ~367–511; Convergence: 2757 iterations; Average Time ≈ 0.020 s.

Visualizations:

•Bar Chart:

Compare average execution times across algorithms.

•Animated GIFs:

➢ Frozen Lake GIF:

Demonstrates the algorithm's progression through the grid and the final optimal path.

➢ TSP GIF:

Shows the evolving tour as the algorithm iterates, capturing convergence and solution improvement.

Conclusions & Future Work

Conclusions:

•Search Algorithms:

✓ Both Branch and Bound and IDA* reliably found the optimal path (cost = 6) with very low execution times.

✓ IDA* showed slightly faster convergence (9 iterations vs. 10) due to effective thresholding.

•Optimization Algorithms:

✓ Hill Climbing is quick but may get trapped in local optima

✓ Simulated Annealing, with its extensive exploration (2757 iterations), is more robust at escaping local optima, achieving lower tour lengths in some runs.

Insights:

- The Manhattan distance heuristic is effective in guiding the search in grid environments.
- In TSP, there's a trade-off between speed and solution quality: HC converges quickly while SA, though slower in convergence, often finds better tours.
- Recorded metrics (reward, iterations, time) provide a clear basis for comparing algorithm efficiency.

Future Work:

•Investigate alternative heuristics or hybrid approaches to improve performance.

•Experiment with parameter tuning, especially for SA (e.g., cooling rate).

•Scale experiments to larger grids and more cities for TSP to evaluate algorithm robustness in complex scenarios.

Final Remarks:

•Summarize the overall performance trade-offs and the impact of heuristic choices.

•Invite questions and discuss potential extensions of your work.