COMP338 - Artificial Intelligence
Project #1: Seating Arrangement Optimization

Instructor: Dr.Radi Jarrar
Student Name: Ahmad Naser

Student ID:1201496

Introduction: In this report, we address the problem of optimizing a seating arrangement around a round table to minimize conflict based on dislike percentages between individuals. We explore three search algorithms: Uniform Cost Search (UCS), Greedy Search, and A* Search, considering a Non-Linear Dislike Cost function. The objective is to evaluate their effectiveness in finding the best seating arrangement while adhering to the requirements.

Problem Overview

Heuristic Table (Dislike Matrix)

The provided heuristic table represents dislike percentages between pairs of individuals, indicating the level of conflict or discomfort each person feels towards others.

Cost Function: Non-Linear Dislike Cost:

A non-linear dislike cost function, *f(x)* maps dislike percentages to costs. This function penalizes highly disliked pairs more severely than a linear mapping would. For this implementation, the function *f(n)* is defined as follows:

f(n)=sum(dislike_percentages_between_neighbors+sum(f(dislike_percentages_between_neighbors))

Objective:

The objective is to determine the best seating arrangement that minimizes conflict while ensuring each person Like their neighbors.

Method used:

Graph Representation:

We model the problem as a graph where each node represents a possible seating arrangement, and edges represent transitions between arrangements

Algorithms Implemented:

1. **Uniform Cost Search (UCS):** Explores nodes with the least cumulative cost first.
2. **Best First Search (Greedy):** Explores nodes with the lowest heuristic (dislike percentage) first.
3. **A * Search**: Evaluates nodes based on both the cost to reach them and the heuristic estimate of the cost to reach the goal.

Results

1. **Best First Search (Greedy)**
   - **Path:** Ahmed Fuad Salem Hani Samir Khalid Hakam Ibrahim Kamal Ayman
   - **Cost:** 21.37

2. **Uniform Cost Search (UCS)**

- **Path:** Ahmed Salem Ayman Hani Kamal Samir Hakam Fuad Ibrahim Khalid
- **Cost:** 28.84
3. *A* Search:*
    - **Path:** Ahmed Salem Ayman Hani Kamal Samir Hakam Fuad Ibrahim Khalid
    - **Cost:** 50.21
    -

Analysis

1. **Best First Search (Greedy):**
- **Strengths:**
    - Finds a relatively low-cost solution quickly due to its greedy nature.
    - Suitable for problems where the heuristic provides a good estimate of the solution quality.
- **Weaknesses:**

    - Prone to getting stuck in local optimal since it doesn't consider the overall cost.
    - May not guarantee the optimal solution.

2. **Uniform Cost Search (UCS):**
- **Strengths:**
    - Guarantees finding the optimal solution.
    - Works well when the cost function is complex and non-linear.
- **Weaknesses**

    - Can be computationally expensive, especially for large search spaces.
    - May explore a large number of nodes before finding the solution.

3. **A* Search:**

- **Strengths:**

    - Combines the benefits of both UCS and Greedy search by considering both cost and heuristic.
    - Can be more efficient than UCS in finding the optimal solution.

- **Weaknesses:**

    - Highly dependent on the accuracy of the heuristic function.
    - Still may encounter the issue of getting stuck in local optimal.

Conclusion:

In this specific problem of optimizing a seating arrangement to minimize conflict, the Best First Search (Greedy) algorithm provided the most cost-effective solution. However, it's important to note that it might not always yield the optimal solution due to its greedy nature. Uniform Cost Search (UCS) guarantees the optimal solution but might be computationally expensive. A* Search strikes a balance between these two approaches and can be more efficient in finding the optimal solution if the heuristic is well-designed. The choice of algorithm depends on the trade-offs between solution quality and computational resources available.