ensia The National School of Artificial Intelligence المدرسة الوطنية العليا للذكاء الاصطناعي

Introduction to Artificial Intelligence

Lab 8 (Week 10) -Search Algorithms: Part 5 - Local search (Hill Climbing)

2023 - 2024

April 22^{nd}, 2024

Objectives

- Implementation of local search algorithms for Travel Planning Problem (Shortest path).

Overview:

LAB 8 delves into another search family known as local search, aimed at optimizing memory usage, which is often seen as a drawback of classical search strategies. In local search, the algorithm focuses solely on the current state (hence the term "local"), selecting from among its children without considering sibling states.

This lab explores two variants of local search: Steepest Ascent, Stochastic Hill Climbing.

Your mission

Your mission in this lab is to perform the following tasks:

Task 1

In this task, you are provided with the following pseudo algorithms in 1, 2, describing Steepest Ascent, Stochastic Hill Climbing.

Your task involves:

- Implementing each of the provided strategies as standalone algorithms to solve the Travel Planning problem from LAB 7, utilizing the straight-line distance (crow flew distance) as a cost function heuristic.
- Combining these algorithms into a single local search algorithm. The resulting algorithm is expected to function similarly to the Graph Search algorithm from previous labs.

Algorithm 1: Steepest Ascent algorithm

```
Input: Problem ;
Output: State (local maximum);
Initialize current_state = Problem_initial_state;
while True do
    neighbor_state = Get_best_neighbor(current_state) ;     /* return the best state among
     successors */
    if neighbor_state.value <= current_state.value then
        return current_state.value ;              /* No better node among children states */
    else
        current_state = neighbor_state ;    /* Assign the child node to explore further */
    end
end
```

Algorithm 2: Stochastic Hill Climbing algorithm

```
Input: Problem ;
Output: State (local maximum);
Initialize current_state = Problem_initial_state;
while True do
    neighbor_state = Get_random_neighbor(current_state);  /* return a random state among
     successors */
    if neighbor_state.value <= current_state.value then
        return current_state.value ;              /* No better node among children states */
    else
        current_state = neighbor_state ;    /* Assign the child node to explore further */
    end
end
```

Task 2

Given that the only difference between local search and classical search strategies is that the frontier in local search contains only the current node and its children in each iteration,

- adjust the Graph search algorithm to account for local search strategies.

Extra Task

Modify the local search strategies to consider the following characteristics:

- In the algorithms, the current state is considered as a whole path (solution) from the initial to the goal node (see Figure 1).
- Any other valid state resulting by tweaking two nodes, adding or removing a node from the path, is considered a neighbor (child) state.
- The cost function is the traveled distance from the initial to the goal node in the path.

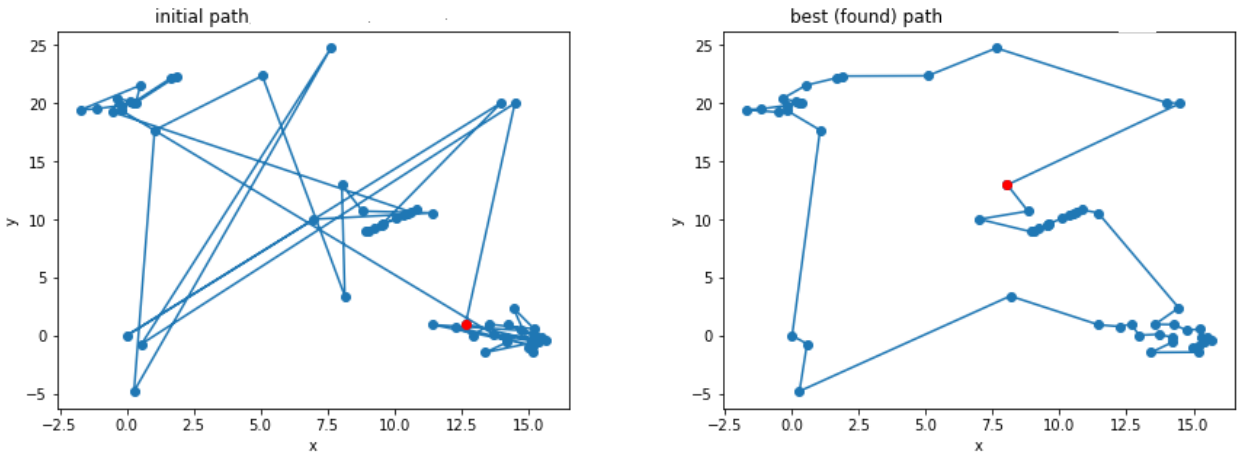

Figure 1: Example of path states