Greedy Algorithm Drasil Case Study

Don Chen

March 1, 2021

This is the proposal of greedy algorithms/heuristics to my drasil case study.

1 Problem Description

- Use greedy heuristics to find the "smallerst" number in a tree. The algorithm can be summarized as:
  - At each node, pick out the note contains the smallest result.
  - Repeat first step until reachs to any leaf.
- Use greedy heuristics in the Traveling Salesman Problem(TSP): Find the shortest distance tour passing through each node of the network exactly once. Nearest Neighbor Alorithm:
  - Start at any city
  - Visit the nearest node not yet yet visited
  - Return to the start node when all other nodes are visited

2 Goal Statements

Find relatively good solution with less costly way.

- This maybe too general, how to define a good solution

3 Assumptions

All types in tree or graph could be quantified. Some type could be hard to be quantified, like boolean. In some decision tree, some nodes will ask yes or no quesiton.

4 Theoretical Model

5 General Definition

6 Supporting Data Definitions

7 Instance Models

- Tree
  - Input: Start node
  - Output: Number
- TSP
  - Input: Start city
  - Output: Number

8 Question

- Greedy algorithms/heuristics is more like a technique rather than an algorithm. Are we looking for two specific algorithm? The reason I aked this question is it is eaiser to list a problem, then try to find a algorithm to solve the problem.

9 Reference

- Nearest Neighbor Alorithm