Homework for Artificial Intelligence for Robotics Assignment 8

Bastian Lang

June 29, 2015

1 Practical Part

1.1 Task

Solve the same Travelling Salesman Problem as the previous assignment, but using Simulated Annealing and all cities in the file. Get the best solution for different computation time (say 5, 10, 15 minutes).

1.2 Approach

For this task I reused classes from the previous travelling salesman assignment to represent cities and solutions and to read the cities from the file. But I used another approach for the hill climbing algorithm. Instead of first computing every possible successor state and only then decide on the best one, I now create only one random successor state by swapping two cities and decide right away on sticking with it or dropping it.
As a temperature I chose the remaining runtime of the algorithm. Initially the user can give a runtime (in minutes). The longer the algorithm runs, the smaller the temperature gets. If it becomes zero the algorithm stops.
For the probability to choose a worse successor I included a constant into the formula similar to the boltzmann constant, which is roughly $8.6 * 10^{-5}$ (I choose 10^{-6}.
I executed the program for 5 min, 10 min, 15 minutes and 30 minutes. The results can be seen below.

1.3 Results

1.3.1 5 min

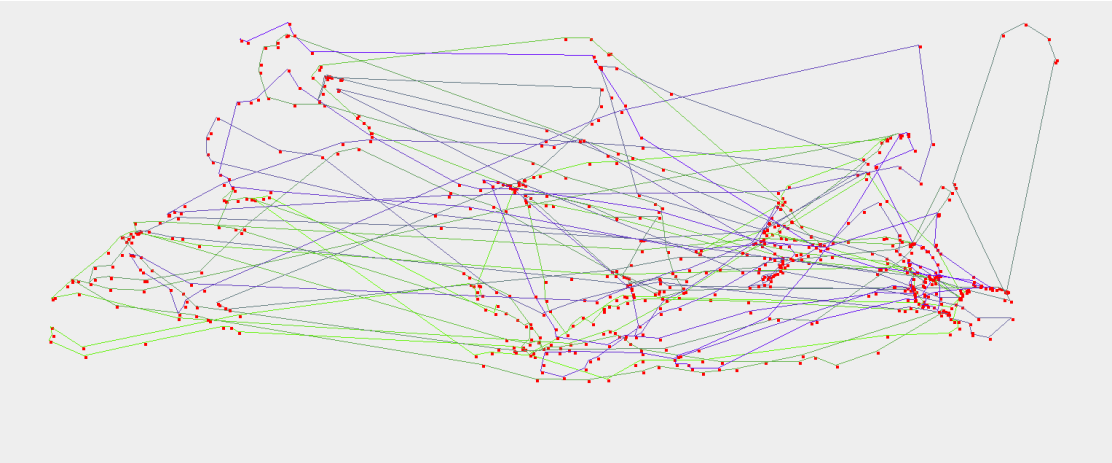

Figure 1.1: path after 5 minutes computation

Path length: 9012.441310
Number of iterations: 2798073
0.061399 percent of total iterations chose a worse successor

1.3.2 10 min

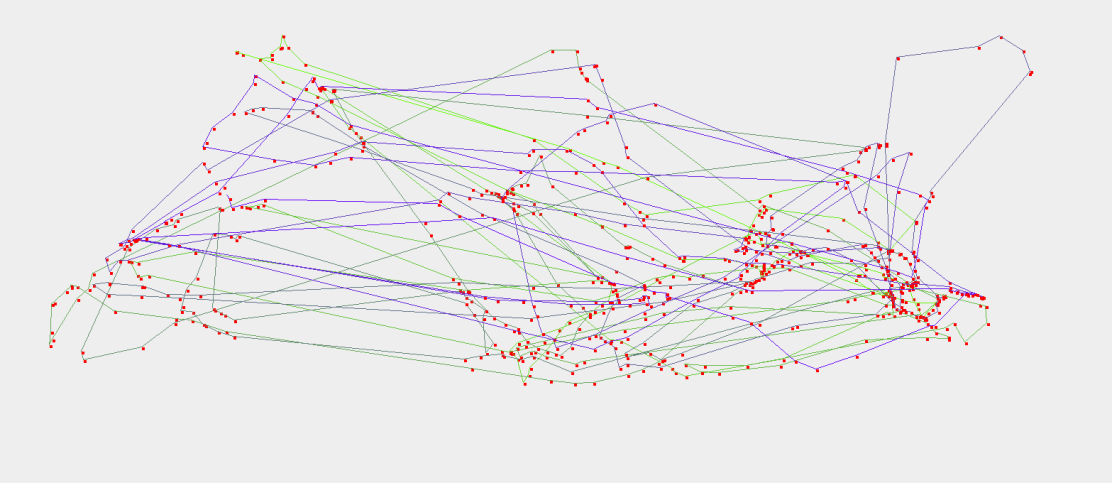

Figure 1.2: path after 10 minutes computation

Path length: 8194.656366

Number of iterations: 5587871
0.092987 percent of total iterations chose a worse successor

1.3.3 15 MIN

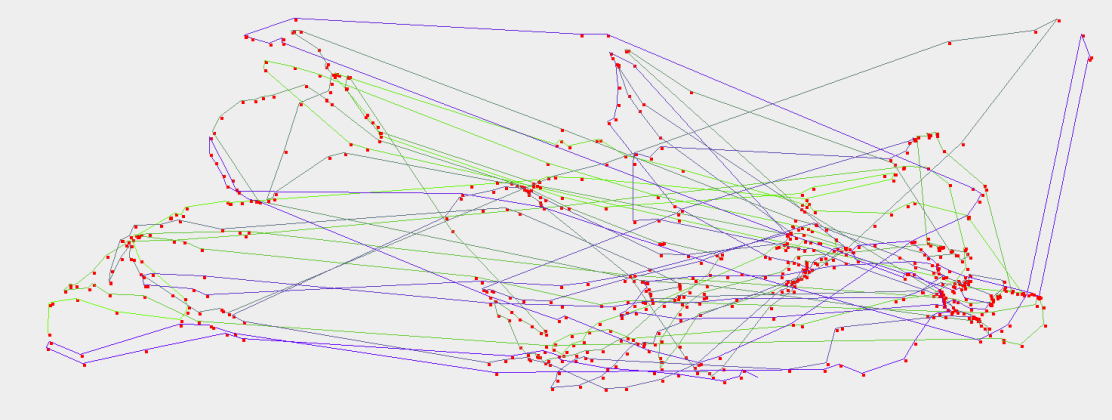

Figure 1.3: path after 15 minutes computation

Path length: 7959.778290
Number of iterations: 8363361
0.127281 percent of total iterations chose a worse successor

1.3.4 30 MIN

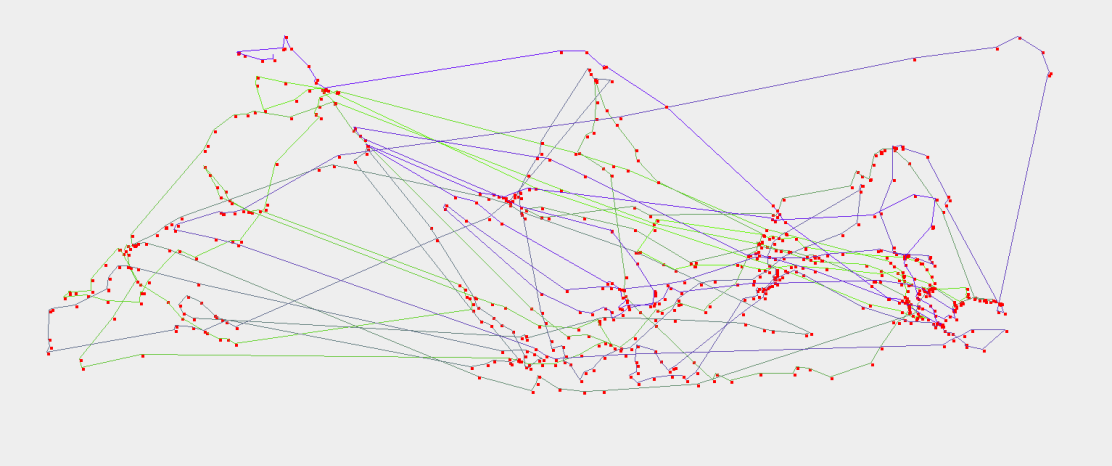

Figure 1.4: path after 30 minutes computation

Path length: 5677.814568

Number of iterations: 16751338
0.196319 percent of total iterations chose a worse successor

2 Theoretical Part

2.1 Task

Consider the following puzzle and represent it as a Constrain Satisfaction Problem. Do not solve/implement it. Provide only the variables, the values of those variables, and the constraints.

- There are five houses.
- The Englishman lives in the red house.
- The Spaniard owns the dog.
- Coffee is drunk in the green house.
- The Ukrainian drinks tea.
- The green house is immediately to the right of the ivory house.
- The Old Gold smoker owns snails.
- Kools are smoked in the yellow house.
- Milk is drunk in the middle house.
- The Norwegian lives in the first house.
- The man who smokes Chesterfields lives in the house next to the man with the fox.
- Kools are smoked in a house next to the house where the horse is kept.
- The Lucky Strike smoker drinks orange juice.
- The Japanese smokes Parliaments.
- The Norwegian lives next to the blue house.

The question posed by the puzzle is: Who drinks water? Who owns the zebra?

2.2 Solution

A **Constraint Satisfaction Problem** is described by defining the **possible states** and the **goal test**. The state consists of **variables** and their **domains**. The goal test consists of all combinations of value assignments to variables that have to be fulfilled for a state to be considered a solution to the problem.

2.2.1 VARIABLES

The Variables are the five houses:
Variables = {First, Second, Third, Fourth, Fifth }

2.2.2 VARIABLE DOMAINS

Each variable consists of exactly five values: The color of the house, the person living in the house, the pet of the person living in the house, the drink the person in the house drinks and the brand of the smoke the person in the house smokes.

Domain = {
{red, green, ivory, yellow, blue},
{Englishman, Spaniard, Ukrainian, Norwegian, Japanese},
{tea, coffee, milk, orange juice, water},
{dog, horse, snails, fox, zebra},
{Lucky Strike, Parliaments, Kools, Old Gold, Chesterfield}
}.

2.2.3 GOAL TEST

Assume that each house has an index according to it's place (First = 1, Second = 2, Third = 3, Fourth = 4 and Fifth = 5) and that *index{...}* returns the house's index of a given value, then the constraints are the following:

- index{Englishman} == index{red}
- index{Spaniard} == index{dog}
- index{coffee} == index{green}
- index{Ukrainian} == index{tea}
- index{green} == index{ivory -1}
- index{Old Gold} == index{snails}
- index{Kools} == index{yellow}
- index{milk} == 3
- index{Norwegian} == 1
- index{Chesterfield} == index{fox} + 1 || index{Chesterfield} == index{fox} - 1
- index{Kools} == index{horse} + 1 || index{Kools} == index{horse} - 1
- index{Lucky Strike} == index{orange juice}
- index{Japanese} == index{Parliament}
- index{Norwegian} == index{blue} + 1 || index{Norwegian} == index{blue} - 1

3 Practical Part

3.1 Task

Solve the same Travelling Salesman Problem as the previous assignment, but using Simulated Annealing and all cities in the file. Get the best solution for different computation time (say 5, 10, 15 minutes).

3.2 Approach

3.3 Result