Assignment 1 - Traveling Salesman Problem

Sergio Catalano s3540294
Emmanouil Gionanidis s3542068

September 17, 2017

Introduction

For this assignment we worked in pair. We wrote the code on Matlab together during the lab session and tested it many times to be sure it worked properly (being exchange students, neither of us had ever used matlab before this course). Then we met a second time to write a proper latex document and to answer the question about the temperature. To write the code with Matlab we did pair programming, switching our roles different times. For the rest of the assignment, one of us wrote the .tex file and the other one elaborated a proper answer to the temperature question.

Basic description

The features requested in the assignment have been obtained by commenting a few lines of the provided code and by adding some new lines to calculate mean and variance and to run the main part of the code for different temperatures, saving the intermediate results in vectors. As for the plot, we used the errorbar(x,y,e) function, as mentioned in the assignment, and labeled the axes as "Temperature" and "Mean", while the error was labeled "Std. Deviation". To obtain a more accurate plot, as suggested, we used a higher value of N(100), of the steps(300) and of different temperatures(12). We also added the special case of temperature = 0, which leads to a normal stochastic descent, since an edge will be substituted only if it shortens the path, without any exception.

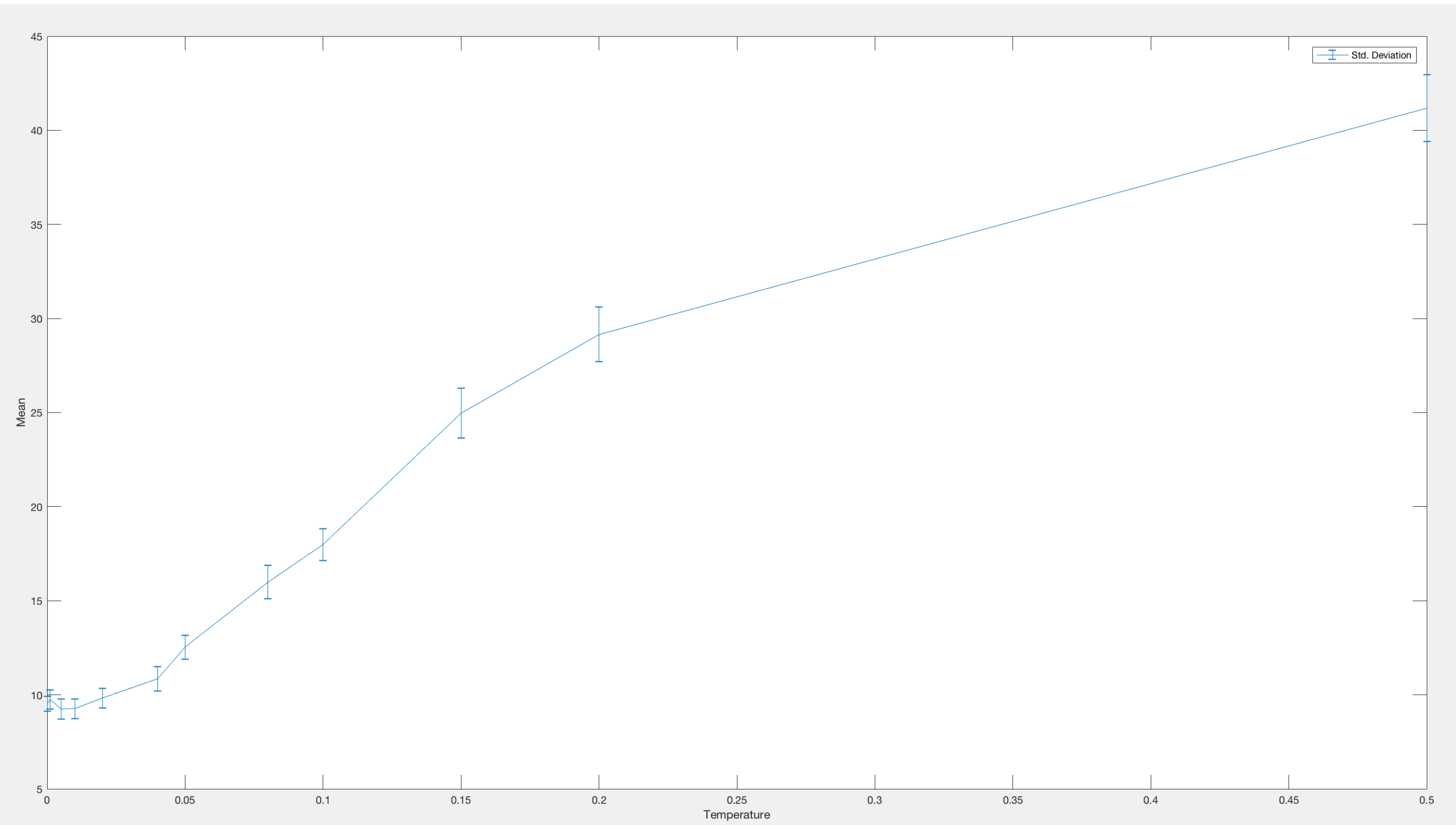

Figure 1: Plot showing the Mean vs. the Temperature, with standard deviations as errorbars

Looking at the plot, it is possible to notice the influence of the temperature on the algorithm. When the temperature is 0, the standard deviation is at its minimum value, since there are no changes allowed, unless they shorten the path, and, after 250000 steps, there should not be many shortening changes left to try. As the temperature grows, the standard deviation also grows, since more and more lengthening changes will be allowed. By increasing the value of the temperature we allow the algorithm to leave a local minimum and to try to find the global one. But if the value of the temperature is not properly decreased after the iterations, the algorithm will just move around the vector space without really finding a local minimum. This is why increasing the value of the temperature also increases the mean of the last 50 lengths of the path. The mean reaches its minimum value with a temperature slightly higher than 0, but small enough to avoid exiting a good local minimum (or preferably the global minimum). The ideal algorithm would start with a relatively high temperature, and it would decrease it very slowly after each group of iterations: with this method the algorithm should converge to the global minimum (if the temperature is decreased slowly enough). But the algorithm must also be fast enough to achieve a solution within realistic computig time.