Travelling Salesman Problem

Kaixin Hu 11129417

December 2015

1 Introduction

The travelling salesman problem(TSP) is stated as follows: given N cities, and a N*N distance matrix where each entry ij gives the distance between city i and city j, find a closed tour such that each city is visited once and the total length of the tour is minimized.

2 Metropolis Algorithm, with Simulated Annealing[1]

2.1 the distribution of configurations

A: a finite set of vectors

C(x):a non-negative cost function defined on A

C^*: C(x)'s optimal value

M: the optimal solutions

$|M|$: the number of elements in M.

$$C^* = \min_{x \in A} C(x)$$

$$M = \{x \in A : C(x) = C^*\}$$

The configuration is distributed according to a Boltzmann distribution, with the probability mass function:

$$P(configuration = x) = \frac{1}{Z(c)} exp(-\frac{C(x)}{c}), \tag{1}$$

in which C(x)is a cost function and c is the 'temperature' or control value. For a very high control value, all configurations have similar probability; As the c decreases, peaks occur with preferred configurations, i.e. with small cost.For a very low control value, in theory it is more likely to find the aimed configurations, although it may take more time.

Hence, the probability mass function on the set of values in A is:

$$P_c(x) = \frac{exp(-\frac{C(x)}{c})}{\sum_{x \in A} exp(-\frac{C(x)}{c})} = \frac{exp(-\frac{C(x)-C^*}{c})}{|M| + \sum_{x \notin M} exp(-\frac{C(x)-C^*}{c})}$$

Since $C(x) - C^* > 0$, as $c \to +0$:

$$P_c(x) = \begin{cases} \frac{1}{|M|}, & \text{if } x \in M \\ 0, & \text{if } x \notin M \end{cases} \tag{2}$$

2.2 Markov chain with the transition probability equal to $P_c(x)$

Since it's difficult to generate random variables with an arbitrary distribution. The Metropolis Algorithm enable us to generate a homogeneous Markov chainX_n that steps through a state spaces, such thatX_n is distributed according to the required probability $P_c(x)$ in function (1).

Set the transition probability of that Markov chain as:

$$P_{ij} = \begin{cases} G_{ij}\Gamma_{ij}, j \neq i \\ 1 - \sum_{alll, l \neq i} G_{il}\Gamma_{il}, j = i \end{cases} \tag{3}$$

in which, G is an irreducible Markov transition probability matrix on the integers 1, ...,m, G_{ij} representing the row i, column j element of G. When $X_n = i$, an random variable X is generated such that $P\{X = j\} = G_{ij}$. If X=j, then X_j is set to j with probability Γ_{ij}, and set to i with probability $1 - \Gamma_{ij}$.

In this report, 2-opt is used to define the neighbours of current configuration, i.e. two non-adjacent edges are deleted and the four cities are reconnected so that a new circuit is created, is implemented to search for a optimal state , so considering there are N cities:

$$G_{ij} = \begin{cases} 2/(N-1)(N-3), \text{for every j in the neighbour of i} \\ 0, \text{otherwise} \end{cases}$$

And Γ_{ij} is set as follows:

$$\Gamma_{ij} = min\left[1, \frac{P(X_j)}{P(X_i)}\right] = min\left[1, exp(\frac{-\Delta C}{c})\right],$$

Such that in the Markov chain, X_i is distributed according to probability function 1, and is time reversible, i.e. amount of transitions from i to j will be equal to those from j to i.

2.3 Implementation procedure

General procedures:

1.Start at a certain temperature(control value: c);

2. Equilibrate the system and sample configurations according to the Boltzmann distribution;

3. Slowly decrease the temperature, until the system settles into the hopefully global minimum.

2.3.1 Local minimum

First, this report set the temperature to a certain value, and implement the Metropolis Algorithm to find a good local minimum.

1.Start at a certain temperature(control value: c);

2. Equilibrate the system and sample configurations according to the Boltzmann distribution;

Specifically, say, in configuration Xi, we generate X from Xi by choosing randomly in its neighbours (2-opt search algorithm). And suppose we select configuration j, set $r = \frac{P(X_j)}{P(X_i)} = exp(\frac{-\Delta C}{c})$, if r≥1, i.e. $\Delta C < 0$, the new configuration is accepted, if r<1, we accept the new configuration with probability r. It means that if the energy or cost of the new configuration is lower than the current one, it is unconditionally accepted, but if the energy is larger, it is accepted with a probability equal to $exp(\frac{-\Delta C}{c})$.

By repeating this Markov process, a chain of configuration $X_1, X_2, ..., X_n$ is generated, and if n is large enough, every configuration will occur in the chain with a frequency that is proportional to the probability density p(x) in function (1).

2.3.2 Simulated Annealing

Further, this report tried different cooling schedules to investigate convergence behaviour.

At high temperature, the cost of the state is not important, the system can explore the phase-space without constrains. While as the temperature lowers down, the cost gets important, and the configuration more likely occurs according to the desired probability distribution. The system will spend more time in states with low cost, so the process can be stopped if no lower cost can be found after a certain number of steps. This report will implement two cooling methods: homogenous Markov chain method and inhomogenous Markov chain method

2.3.2.1 homogenous Markov chain method In this method, the Markov chains is simulated at a series of temperatures while the temperature is kept constant during the creation of each chain. The annealing is repeated several times with different temperature and different starting states.

Variables should be set according to certain situations are as follows:

Starting temperature, which should be chosen such that a certain percent of generated tours are accepted at this value;

Method of lowering down the temperature: $c^* = c * rate$;

Stop condition or Number of Markov chains: stop when no states with lower energy is found after certain number of simulations.

Length of Markov chains : Run for certain and enough steps.

2.3.2.2 inhomogenous Markov chain method In this method, the temperature is decreased after every Metropolis step. Initially we are able to make some pretty big jumps around the solution landscape. By the end of a run we'll be jumping around less and less.

The temperature or the control value in this method:

$$c = \frac{1}{log(1+n)}$$

, with n is the number of Metropolis step in inhomogenous Markov chain.

3 Results

3.1 eli51.tsp

3.1.1 Simulated Annealing with homogenous Markov chain

Starting temperature: 10, under which 30.4 percent of generated tours are accepted at this value;

Cooling schedule: $c^* = c * rate$, rate=0.95;

Number of Markov chains: 200;

Length of Markov chains: 500 steps;

Initial tour: in the order of 1, 2, 3,...,51,1.

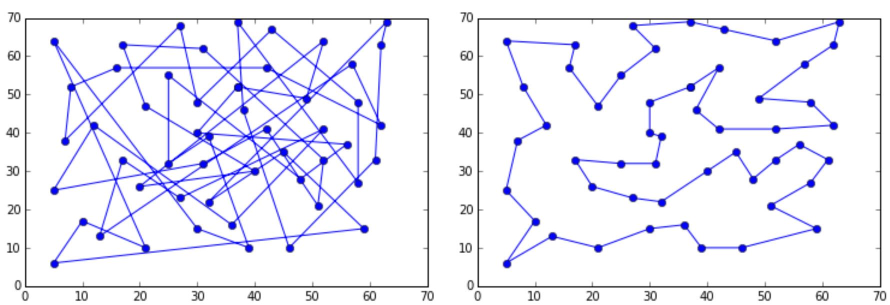

Figure 1: TSP eil-51 Cities version. The initial tour and the final tour after Simulated Annealing

By running for several times, the best outcome is shown in Figure 1, with the optimal tour in a coordinate system : [1. 22. 32. 11. 16. 21. 29. 2. 20. 35. 36. 3. 28. 31. 26. 8. 48. 6. 23. 7. 43. 24. 14. 25. 13. 41. 40. 19. 42. 44. 15. 45. 33. 39. 10. 30. 34. 50. 9. 49. 38. 5. 37. 17. 4. 18. 47. 12. 46. 51. 27. 1.], and the cost is 436.87.

Figure 2 shows the decreasing of accepted generated tours, illustrating that as the temperature cools down, the Markov chain is more likely in the states with low cost.

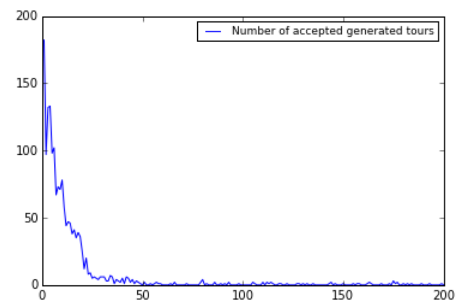

Figure 2: Accepted number of generated tours in each Markov chain, with decreasing temperature.

3.1.2 Comparison between SA homogenous and inhomogenous Markov chain method and greedy improvement algorithm

Figure 3 is the cost of generated tours, as a function of iteration number for Simulated Annealing method, homogenous and inhomogenous Markov chain respectively, and greedy improvement algorithm. Figure 4 is to zoom in for the early periods.

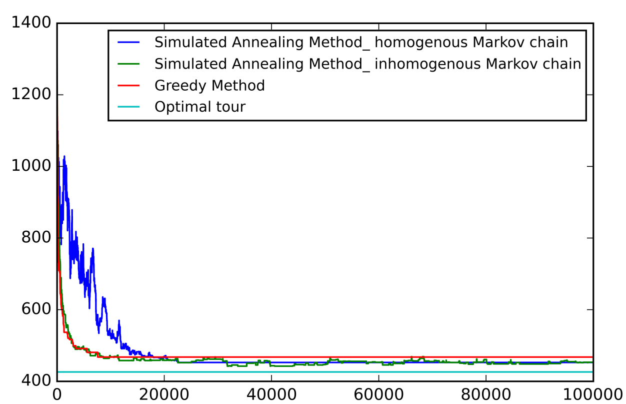

Figure 3: Comparison of the cost of tours generated from homogenous and inhomogenous Markov chain method, and Greedy improvement algorithm

In this two graphs, the greedy improvement algorithm quickly terminates in a local minimum, while the Simulated Annealing method, with homogenous Markov chains, decreases quite slowly, yet a obvious lower cost of the tour. This means the the greedy improvement algorithm may get stuck in a local minimum.

The Simulated Annealing method, with inhomogenous Markov chains, converge more quickly at early iterations, compared with homogenous Markov

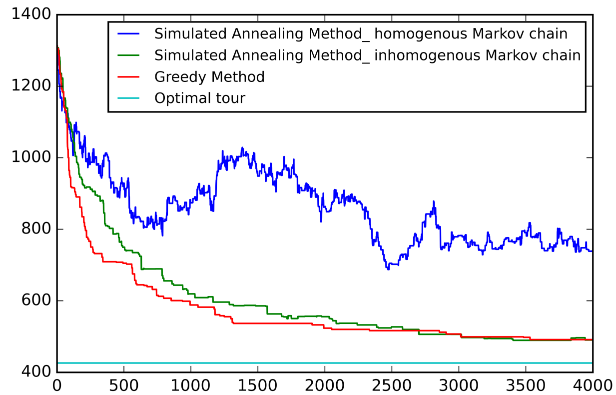

Figure 4: Comparison of the cost of tours generated from homogenous and inhomogenous Markov chain method, and Greedy improvement algorithm.(Zoom in the early steps)

chains method, but as the cost asymptotically gets to the actual optimal line, it exhibits more fluctuation, making it unreliable to decide to a certain stop criterion.

Hence, in the following part for more complicated situations, the Simulated Annealing, with homogenous Markov chains, will be analysed more closely.

3.2 a280.tsp

The variables for Simulated Annealing are set as follows:

Starting temperature: 100, 35.5 percent of generated tours are accepted at this initial temperature. As temperature cools down the number of accepted tours also declines gradually, as Figure 7 shows.

Cooling schedule: $c^* = c * rate$, rate=0.999;

Number of Markov chains and stop condition: the accumulated accepted tours in an Markov chain simulation is zero for 1000 number of Markov chains.

Length of Markov chains : 1000 steps;

Initial tour: in the order of [1, 2, 3,...,280,1].

Figure 5 shows the final tour and the initial tour after Simulated Annealing, the final tour is as follow:

[1. 2. 242. 243. 244. 241. 240. 239. 246. 245. 247. 250. 251. 230. 231. 238. 237. 236. 234. 235. 233. 232. 229. 228. 227. 226. 225. 224. 223. 222. 221. 219. 220. 218. 217. 216. 213. 215. 214. 211. 212. 207. 210. 209. 252. 255. 254. 253. 208. 206. 205. 204. 203. 202. 201. 200. 199. 198. 197. 195. 196. 194. 192. 191. 193. 190. 189. 188. 187. 186. 185. 183. 184. 182. 181. 176. 180. 179. 178. 177. 151. 152. 156. 153. 154. 155. 129. 130. 131. 20. 21. 128. 127. 126. 125. 124. 123. 122. 121. 120. 119. 157. 159. 158. 160. 175.

161. 162. 163. 164. 165. 166. 167. 168. 169. 170. 171. 172. 106. 173. 174. 107. 105. 103. 104. 108. 110. 111. 114. 115. 117. 116. 113. 112. 88. 87. 84. 83. 82. 79. 80. 81. 89. 109. 90. 91. 92. 102. 101. 100. 99. 98. 93. 94. 97. 96. 95. 78. 77. 76. 75. 74. 73. 72. 71. 70. 68. 69. 67. 66. 85. 65. 86. 64. 63. 62. 118. 61. 60. 43. 42. 45. 44. 59. 58. 57. 56. 55. 46. 54. 53. 52. 51. 50. 49. 48. 47. 41. 40. 39. 38. 37. 36. 35. 34. 33. 32. 31. 30. 29. 27. 28. 26. 22. 25. 23. 24. 14. 15. 13. 12. 11. 10. 8. 9. 7. 6. 5. 4. 277. 276. 275. 274. 273. 272. 271. 16. 17. 18. 19. 132. 133. 134. 270. 269. 135. 136. 268. 267. 137. 138. 139. 150. 149. 148. 147. 146. 145. 144. 143. 142. 141. 140. 266. 265. 264. 263. 262. 260. 261. 259. 258. 257. 256. 249. 248. 278. 279. 3. 280. 1.], and the cost is 2768.

Figure 6 is the costs of tours generated in each Markov chain step. The right-hand graph shows at early steps, as the temperature is high enough, the cost rises, meaning the initial tour is or at least near a local minimum. So in this case, the temperature should be set as very high to avoid stuck in the local minimum. As temperate cools down gradually, the cost fluctuates, and slowly goes down to the final 2768, which is lower than the initial cost 2818, although still not the same as the actually optimal cost 2587.

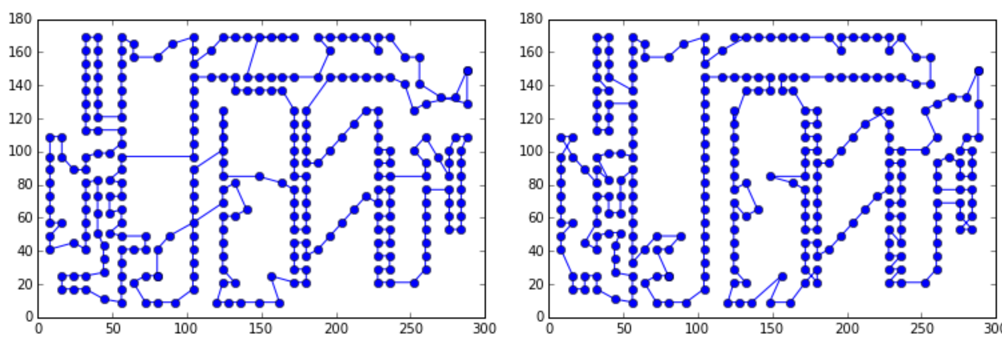

Figure 5: TSP a280 Cities version. The initial tour and the final tour after Simulated Annealing

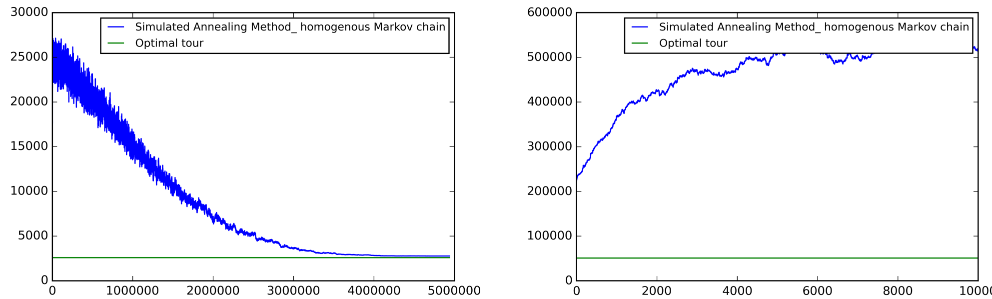

Figure 6: TSP a280 Cities version.The cost of tours generated, as a function of iteration number for Simulated Annealing method.The right one is to zoom in the early steps.

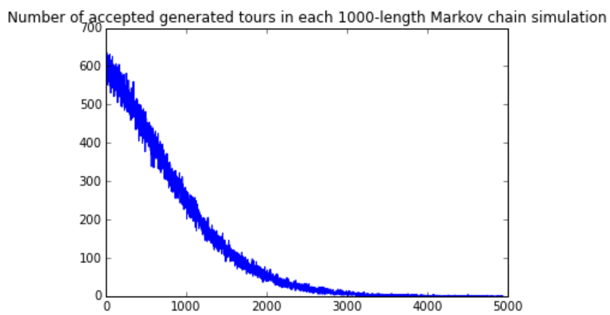

Figure 7: Number of accepted generated tours in each 1000-length Markov chain simulation

3.2.1 the convergence behavior with different cooling system

3.2.1.1 Temperature :

This part will analyse the the convergence behavior with different initial temperatures. The cooling schedule is set as follows:

Starting temperature: Range from 1 to 100, with an interval of 10;

Cooling schedule: $c^* = c * rate$, rate=0.999;

Number of Markov chains and stop condition: the accumulated accepted tours in an Markov chain simulation is zero for 1000 number of Markov chains.

Length of Markov chains : 1000 steps;

Initial tour: in the order of [1, 2, 3,...,280,1].

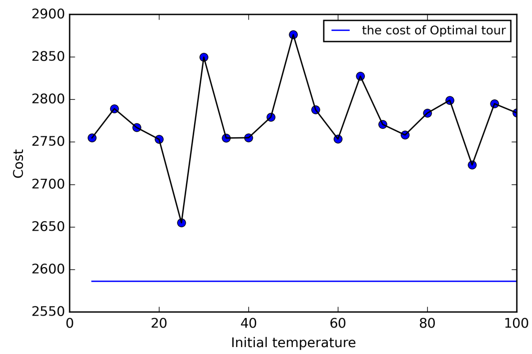

Figure 8: TSP a280 Cities version. The convergence of final tour's cost as initial temperature rises.

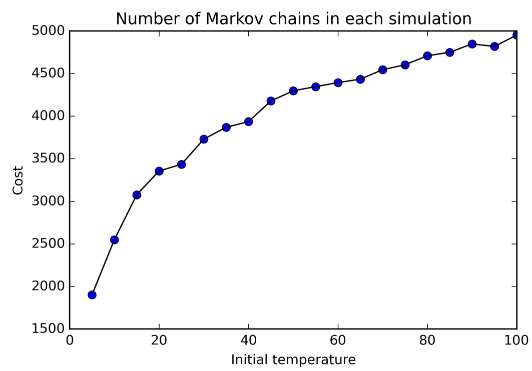

Figure 9: TSP a280 Cities version.The Number of Markov chains needed to reach the stop condition in different SA simulations with various initial temperatures.

Figure 8 plots the final tour's cost under various initial temperature. It doesn't show convergence to the minimum cost, but as the initial temperature rises, the fluctuation of the local minimum declines, that is, When the initial temperature is lower than 60, the final cost sometimes stuck in a high local minimum, sometimes relatively low minimum.

Figure 9 shows, as initial temperature rises, it needs more number of Markov chains to meet the stop condition. This is reasonable in that, with same level of cooling rate, the higher initial temperature needs more time to cool down, and this generates more stable final costs according to Figure 8. But those final tours still get stuck in some local minimum, for the costs is higher than actually minimum cost.

3.2.1.2 Cooling rate :

This part will analyse the the convergence behavior with different cooling rate.

Starting temperature: 100;

Cooling schedule: $c^* = c * rate$, rate ranges from 0.960 to 0.996, with an interval of 0.004 ;

Number of Markov chains and stop condition: the accumulated accepted tours in an Markov chain simulation is zero for 1000 number of Markov chains.

Length of Markov chains : 1000 steps;

Initial tour: in the order of [1, 2, 3,...,280,1].

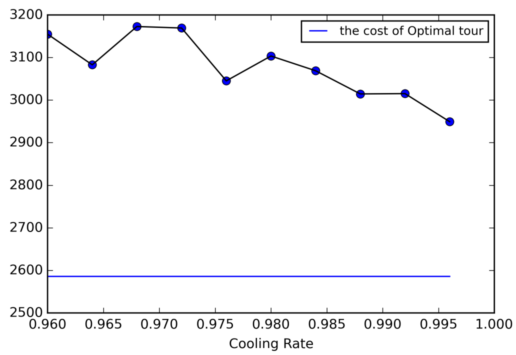

Figure 10: TSP a280 Cities version. The convergence of final tour's cost as cooling rate rises.

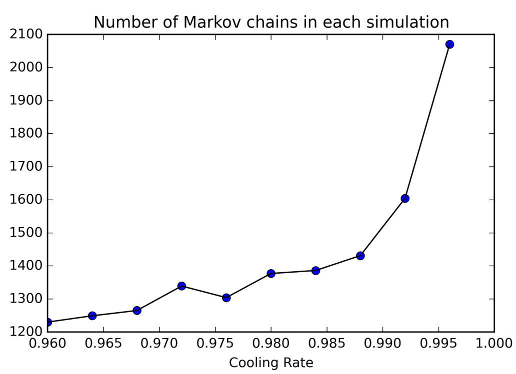

Figure 11: TSP a280 Cities version.The Number of Markov chains needed to reach the stop condition in different SA simulations with various cooling rate

Figure 10 plots the final tour's cost under each cooling rate, and Figure 11 plots the number of Markov chains needed to meet the same stop condition. As rate goes to near 1, the final cost also goes near to the optimal tour's cost. This is reasonable, for slower the cooling process, the more likely the SA method could escape a local minimum, and at the same time not violate too much from previous Markov chain simulation in the following simulation, making it quicker to find the local minimum in the following simulation. But the adopt of slower cooling process, thus more desirable final tour is at the expense of larger number of Markov chains needed as it shows in Figure 11.

3.2.2 Effects of Markov Chain's length on the convergence behavior

This part will analyse the the convergence behavior with various Markov Chain's length.

Starting temperature: 100;

Cooling schedule: $c^* = c * rate$, rate=0.999;

Number of Markov chains and stop condition: the accepted tours in an Markov chain simulation is zero for 1000 number of Markov chains.

Length of Markov chains :ranging from 100 to 1000 , with an interval of 100 steps;

Initial tour: in the order of [1, 2, 3,...,280,1].

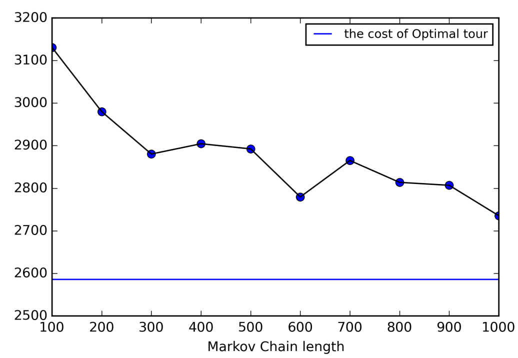

Figure 12: TSP a280 Cities version. The convergence of final tour's cost as Markov Chain's length increases.

Figure 12 plots the final tour's cost under various Markov chain's length, when the length is shorter than 600 steps, as the length increases, the final tour's cost is more closer to the optimal; But as the length is large enough,the cost doesn't improve as the length increases.

Figure 13 plots the number of Markov chains needed to meet the same stop condition. As the length increases, the number first increases, but after reaching a certain point, the number stops increases. This is correspondent with the phenomenon in Figure 12 .

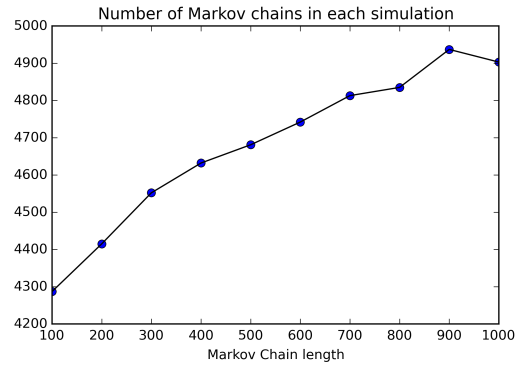

Figure 13: TSP a280 Cities version.The Number of Markov chains needed to reach the stop condition in different SA simulations with various Markov Chain's length

3.3 pbc442.tsp

The variables for the Simulated Annealing are set as follows:

Starting temperature: 1000, 30.6 percent of generated tours are accepted at this initial temperature.

Cooling schedule: $c^* = c * rate$, rate=0.999;

Number of Markov chains and stop condition: the accumulated accepted tours in an Markov chain simulation is zero for 1000 number of Markov chains.

Length of Markov chains : 1000 steps;

Initial tour: in the order of [1, 2, 3,...,280,1].

Figure 15 is the costs of tours generated in each Markov chain step, the final tour's cost: 57452.5.

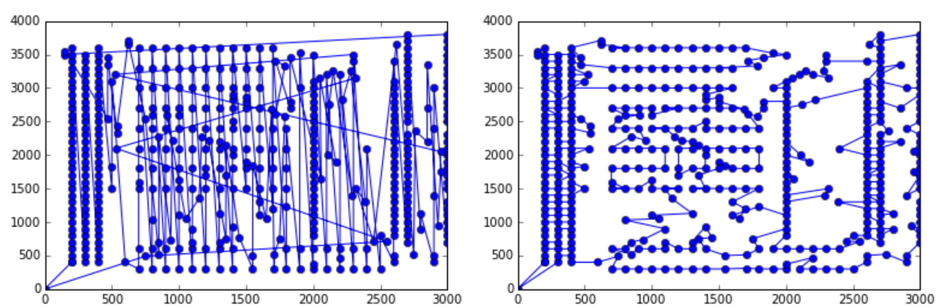

Figure 14: TSP pbc442 Cities version.The initial tour and the final tour after Simulated Annealing

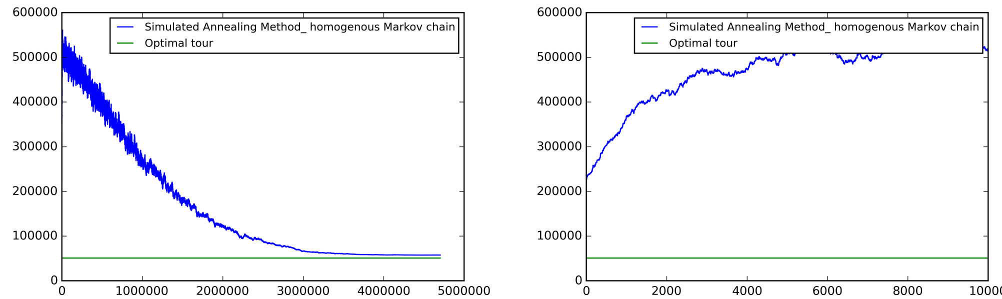

Figure 15: TSP pbc442 Cities version.The cost of tours generated, as a function of iteration number for Simulated Annealing method.The right one is to zoom in the early steps.

References

[1] Peter J Van Laarhoven and Emile H Aarts. *Simulated annealing: theory and applications*, volume 37. Springer Science & Business Media, 1987.