Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Engineering

Elixir Comp. Engg. 95 (2016) 41057-41059

ISSN: 2229-712X

Performance Evaluation of Travelling Salesman Problem Based on Artificial Simulated Annealing Algorithm

Dr.R.Mala, K.Karthikeyan and M.Manikandan

Department of Computer Science Maradupandiyar College, Vallam, Thanjavur.

ARTICLE INFO

Article history:

Received: 9 May 2016;

Received in revised form:

11 June 2016;

Accepted: 17 June 2016;

Keywords

TSP, Artificial Neural, Annealing Algorithm, Sub optimal.

ABSTRACT

Given a collection of cities and the cost of travel between each pair of them, the traveling salesman problem, or TSP for short, is to find the cheapest way of visiting all of the cities and returning to your starting point. In the standard version we study, the travel costs are symmetric in the sense that traveling from city X to city Y costs just as much as traveling from Y to X. The Traveling Salesman Problem is typical of a large class of "hard" optimization problems that have intrigued mathematicians and computer scientists for years. Most important, it has applications in science and engineering I attempt to apply simulated annealing to find (Sub-Optimal) solutions to TSP with 100-200 cities randomly. The performance by changing the parameter values and try to understand how fast and effective simulated annealing algorithm can generate a solution to TSP problem.

I. Introduction

The Traveling Salesman Problem (TSP) is famous as a resource extensive algorithm due to its complex computation. Given a number of cities representing vertices and edges representing paths, this algorithm would compute the shortest path for a person to make a round trip visiting all the cities only once. This algorithm has been applied in many engineering and technological application [2],[3],[4] such as electronic circuitry design, neuron chain, oil and gas transportation application, DNA research and protein folding. The importance of gaining the shortest path in these computationally complex applications has led to the optimality of operation thus achieving high-quality and accurate results. It is fairly clear that TSP algorithm needs to run in a high performance computing facility in order to achieve comparable result within diminutive time. It is known to the mathematical and computing community that TSP is a time consuming algorithm in getting to the shortest path for few to enormous vertices[1].

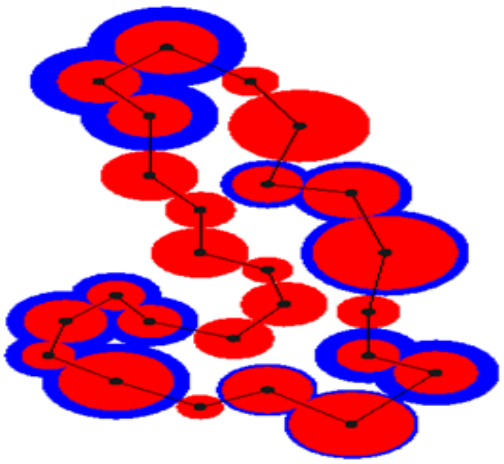


Figure 1. Shortest path structure.

II. Simulated Annealing

Simulated Annealing (SA) is motivated by an analogy to annealing in solids. The idea of SA comes from a paper published by Metropolis etc al in 1953 [Metropolis, 1953). The algorithm in this paper simulated the cooling of material in a heat bath. This is a process known as annealing.

If you heat a solid past melting point and then cool it, the structural properties of the solid depend on the rate of cooling. If the liquid is cooled slowly enough, large crystals will be formed. However, if the liquid is cooled quickly (quenched) the crystals will contain imperfections[5]. Metropolis's algorithm simulated the material as a system of particles. The algorithm simulates the cooling process by gradually lowering the temperature of the system until it converges to a steady, *frozen* state. In 1982, Kirkpatrick et al (Kirkpatrick, 1983) took the idea of the Metropolis algorithm and applied it to optimisation problems. The idea is to use simulated annealing to search for feasible solutions and converge to an optimal solution.

A good explanation of the simulated annealing algorithm is given by Robert C. Williams: "The impetus behind the SA algorithm is its analogy to the statistical mechanics of annealing solids ... When a physical system is at a high temperature, the atoms in the system are in a highly disordered state, and consequently the associated ensemble energy of the system is also high. Lowering the temperature of the system results in the atoms of the system acquiring a more orderly state, thus reducing the energy of the system."

For example, to grow a crystal, which is highly ordered, the system needs to be heated to a temperature which allows many atomic rearrangements. Then the system must be carefully cooled, ensuring a thermal equilibrium is reached at each temperature stage, until the system is `frozen 'into a good crystal. If the cooling process is performed too quickly, extensive irregularities can be locked into the crystals structure, with the resulting energy level far greater than in a perfect crystal.

Tele:

E-mail address: kirithicraj@gmail.com

III. How it Works

In simulation, a minima of the cost function corresponds to this ground state of the substance. The simulated annealing process lowers the temperature by slow stages until the system "freezes" and no further changes occur. At each temperature the simulation must proceed long enough for the system to reach steady state or thermal equilibrium. Metropolis algorithm outlines the whole process: [Foge94]

An annealing algorithm needs four basic components: [Foge94]

1. Configurations

A model of what a legal placement (configuration) is. These represent the possible problem solutions over which we will search for an answer.

2. Move set

A set of allowable moves that will permit us to reach all feasible configurations and one that is easy to compute. These moves are the computations we must perform to move from configuration to configuration as annealing proceeds.

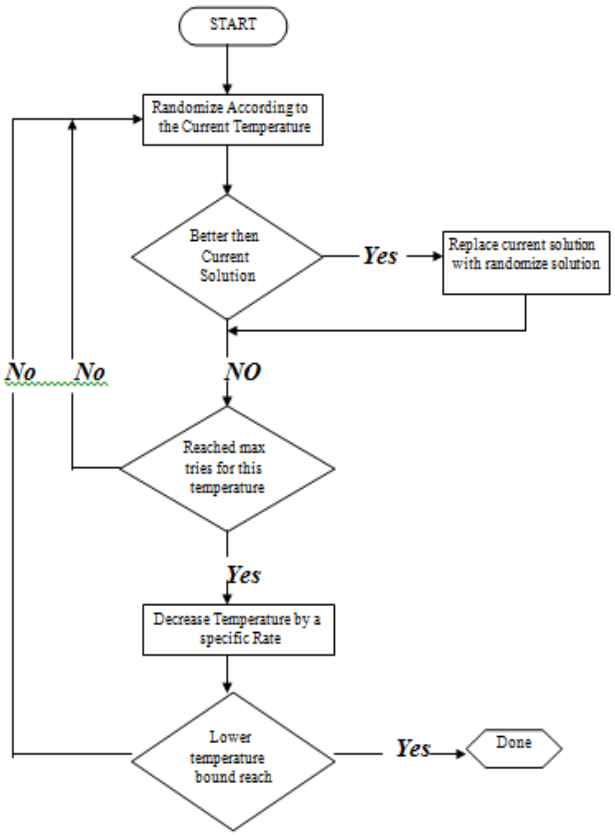
3. Cost function:

To measure how well any given placement configuration is.

4. Cooling schedule:

To anneal the problem from a random solution to a good, frozen, placement. Specifically, we need a starting hot temperature (or a heuristicfor determining a starting temperature for the current problem) and rules to determine when the current temperature should be lowered, by how much the temperature should be lowered, and when annealing should be terminated.

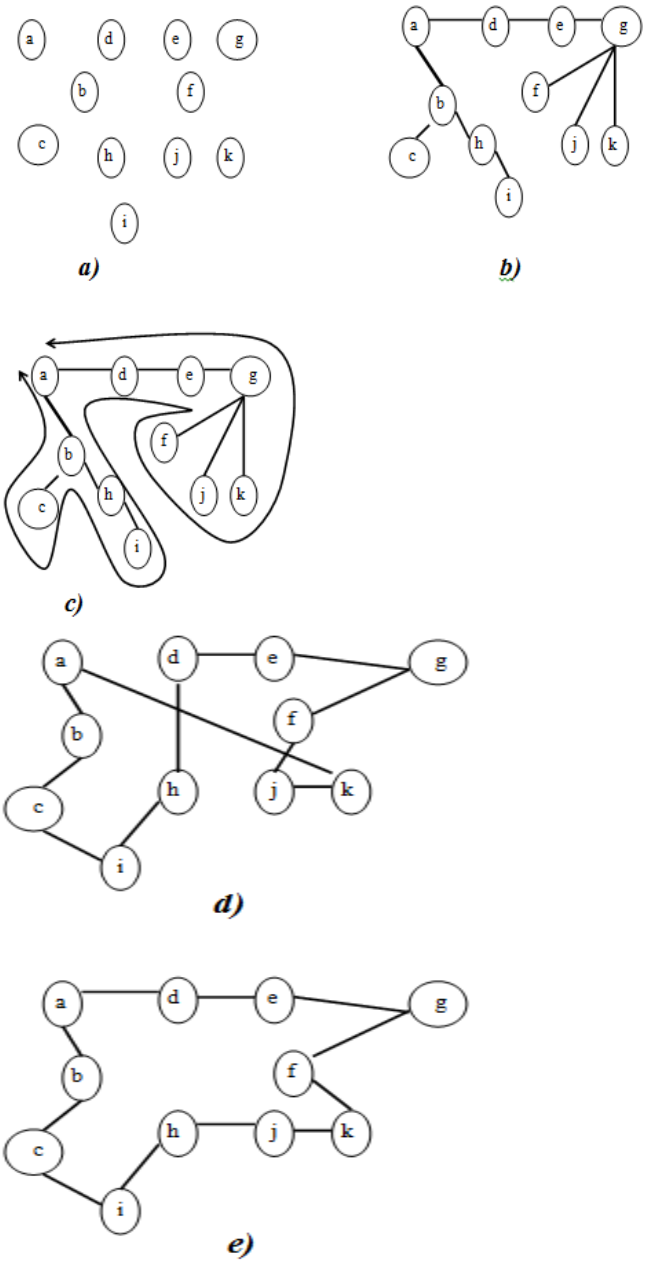
IV. Overview of Simulated Annealing Process



V. Algorithm

repeat
Choose two cities at positions i, j of the tour
read cities i, i+1,i-1,j,j+1,j-1
compute inter city distance
Let detaC = difference in cost
choose random number R
If R < exp(-deltaC/T) then
Swap cities I and j
Re-write inter-city distance
End if
Until system is frozen

Working



a) According to the Simulating Annealing Algorithm, there are eleven cities a,b,c,d,e,f,g,h,i,j,k in which f is one unit to the right and two units up from j. The ordinary Euclidean distance is used as the cost function between two points.

b) A minimum span tree T of these points. Vertex a is the root vertex. The vertices happen to be labeled in such a way that they are added to the main tree by Annealing algorithm in a alphabetical order.

c) A walk of T , starting at a. a full walk of the tree visit the vertices in the order a,b,c,b,h,I,h,b,a,d,e,g,f,g,j,g,k.

d)A tour of the vertices obtained by vesting the vertices in the order given by the preorder walk.

e) An optimal tour H* for the given set of vertices. Its total cost is approximately 14.715.

V. Concluding Remarks

The simulated annealing algorithm can be a useful tool to apply to hard combinatorial problems, and although one appeal of the algorithm is its apparent universality, one must keep in mind that some care must be taken in application as each implementation requires choices that essentially determine the actual efficiency of the procedure. [6]. I might find a good set of parameters for simulated annealing algorithm it is still worthwhile trying to improve the performance (where performance could mean the quality of the solution returned, the time taken by the algorithm etc.) of the algorithm.

References

[1] Izzatdin Abdul Aziz, Low Tan Jung, and Mazlina Mehat " Parallelization Of Traveling Salesman Problem Using Rocks Cluster ", Proceedings of the International MultiConference of Engineers and Computer Scientists 2008 Vol I IMECS 2008, 19-21 March, 2008, Hong Kong

[2] P. Borovska,(2006,June). Solving the Travelling Salesman Problem in Parallel by Genetic Algorithm on Multicomputer Cluster. International Conference on Computer Systems and Technologies 2006. [Online]. Available: http://ecet.ecs.ru.acad.bg/cst06/index.php?cmd=dPage&pid=cpr

[3] K.Bryant, A. Benjamin, "Genetic Algorithms and Traveling Salesmen Problem", Thesis, Harvey Mudd College, Dept. of Mathematics, 2000.

[4] M. Manfrin, M. Birattari, T. Stutzle, and M. Dorigo, (2011, April). Parallel Ant Colony Optimization for the Traveling Salesman Problem, ANTS 2011: Fifth International Workshop on Ant Colony Optimization and Swarm Intelligence. [Online].

[5] Aarts, E.H.L., Korst, J.H.M. 2009. Simulated Annealing and Boltzmann Machines. Wiley, Chichester.

[6] Zakir H. Ahmed, Genetic Algorithm for the Traveling Salesman Problem using Sequential Constructive Crossover Operator.