An Approximate Algorithm for Triangle TSP with a Four-Vertex-Three-Line Inequality

Yong Wang, School of Renewable Energy, North China Electric University, Beijing, China

ABSTRACT

Traveling salesmen problem (TSP) is a classic combinatorial optimization problem. The time complexity of the exact algorithms is generally an exponential function of the scale of TSP. This work gives an approximate algorithm with a four-vertex-three-line inequality for the triangle TSP. The time complexity is O(n^2) and it can generate an approximation less than 2 times of the optimal solution. The paper designs a simple algorithm with the inequality. The algorithm is compared with the double-nearest neighbor algorithm. The experimental results illustrate the algorithm find the better approximations than the double-nearest neighbor algorithm for most TSP instances.

Keywords: Approximate Algorithm, Four-Vertex-Three-Line Inequality, Travelling Salesman Problem

1. INTRODUCTION

Given a map with a set of n cities and the distances d_{ij} (1 \leq i, j \leq n and i \neq j) between each pair of them, find the shortest tour which visits each of the cities once and exactly once. Here n is a finite integer. This is the classic traveling salesman problem (TSP). A tour with each of the cities once is known as the Hamiltonian cycle (HC) or Hamiltonian tour. The objective of TSP is to detect the shortest HC, i.e., the best tour with length l_{min} = \text{Min}(l(\text{HC})) and where l(\text{HC}) means the length of the HC. In graph theory, the TSP map is usually represented as a complete weighted graph. The weights on the edges denote the distance, time or expense for various kinds of TSP. Therefore, a mathematical formulation of TSP is to find the HC with the minimum (or maximum) weights in a complete weighted graph. In this paper, we take the minimum HC as the best tour and the weight is also called the cost.

TSP is easy to describe but hard to resolve. It has been proven to be NP-complete. For the symmetrical TSP with n cities, the number of HCs reaches (n-1)!/2. It is a challenging work to design an efficient algorithm for TSP. The time complexity of the exact algorithms is generally an exponential function of the scale of TSP. Finding the best tour is our desired goal whereas the exact algorithms are time-consuming for large scale of TSP instances. In 1962, Held and Karp

DOI: 10.4018/ijamc.2015010103
(1962) and Bellman (1962) used the dynamic programming method to detect the best tour in \(O(n^22^n)\) time. Till 2010, the time is reduced to \(O(1.657^n)\) by Andreas (Andreas, 2010) with a Monte Carlo method. Please refer to the literatures (Woeginger, 2003; Matai, Prakash & Mittal, 2010; Applegate, Bixby, Chvatal & Cook, 2006) to acquire more information about the exact algorithms for TSP. Although some TSP with thousands of cities has been resolved by the exact algorithms, such as the improved cutting plane method (http://www.math.uwaterloo.ca/tsp/index.html), it is believed the exact algorithm does not exist unless \(NP=\text{P}\). The experiments on exact algorithms show that they are feasible for TSP with less than 1,000 cities (Singh, & Oudheusden, 1997; Klerk & Dobre, 2011; Gouveia, Voß, 1995; Perez, & Gonzalez, 2004). Once the scale of TSP becomes larger, the computation time will increase tremendously even the super computers are utilized (Levine, 2000; Ergan & Orlin, 2006).

On the other hand, the approximate algorithms are extensively studied to find an approximation of the best tour. They are classified into the tour improvement methods and the tour construction methods. The representative tour improvement method is the \(k\)-opt algorithms, such as 2-opt (Croes, 1958; Verhoeven, Aarts & Swinkels, 1995), 3-opt (Luc, Patrick, Dirk, & Dirk, 2005) and Lin-Kernighan heuristics (LKH) (Helsgaun, 2012). These approximate algorithms have been tested by large scale of TSP with millions of cities (Helsgaun, 2012; Johnson, 2002). Some approximate algorithms, such as the LKH, are competitive to generate the satisfactory approximations quickly. The tour construction methods include the nearest neighbor algorithms (Adrian, Joseph; 2003), nearest insertion algorithm, doubling minimum spanning tree (DMST) (Thomas, Charles, Ronald & Clifford, 2006) and the Christofides heuristics (Christofides, 1976) etc. More tour construction methods are summarized by Johnson (Johnson, 2002) and their performance is also examined. The constraints play an important role to generate a good approximation, such as the triangle inequality. If the costs of TSP obey the triangle inequality, i.e., \(c_{ij}+c_{jk} \leq c_{ik}\) holds for arbitrary three cities \(i, j, k\), we call this kind of TSP as triangle TSP. For triangle TSP, the approximations less than 2 and 3/2 times of the best tour will be found with the DMST heuristics and Christofides heuristics, respectively. Many other heuristics are designed based on the triangle inequality besides the DMST and Christofides heuristics. A path cover algorithm is given by Berman and Karpinski (2006) for the 1, 2-TSP, it searches the approximation of 8/7 times of the best tour. The asymmetrical triangle TSP is studied by Frieze et al. (2003), Bläser (2008), Kaplan, et al. (2005) and their algorithms produce the approximations of \(\log n\), \(0.999\log n\) and \(0.842\log n\) times of the best tour, respectively. Give a parameter \(\gamma>0\), the parameterized triangle TSP satisfy the triangle inequality \(c_{ij} \leq \gamma(c_{ij} + c_{jk})\) with a real \(\gamma\). Thomas and Hans (1995) use Christofides heuristics to search the approximation. They found the approximations change according to the parameter \(\gamma\). In worst case, the approximations becomes infinite if \(\gamma\) is bigger than 1. In the cases of \(1/2 \leq \gamma < 1\), Böckenhauer et al. (2000) gave two approximate algorithms to find the \(1+2(2\gamma-1)/3\) and \(2/3+\gamma/(3-3\gamma)\) times of the best tour. For asymmetrical \(\Delta\)-TSP, Chandran and Ram (Chandran & Ram, 2002) introduced the heuristics which can find the \(\gamma/(1-\gamma)\) times of the best tour.

In this paper, we introduce an approximate algorithm with a four-vertex-three-line inequality for the triangle TSP. The four-vertex-three-line inequality is derived from a sub-graph composed of four connected vertices. Given a sub-graph with four connected vertices, it includes \(4!/2\) paths with four vertices. However, half of these paths are useless to find the best tour. For example, the path with the bigger cost of the two paths \(P_i=(h,i,j,k)\) and \(P_{j,k}=(h,j,i,k)\) will not be included by the best tour. The path with the smaller cost is convenient to compute with a four-vertex-three-line inequality, i.e., \(c_{hi}+c_{ij}+c_{jk} \geq c_{bj}+c_{pj}+c_{ik}\) or \(c_{hi}+c_{ij}+c_{jk} \leq c_{hi}+c_{pj}+c_{ik}\). We only consider the half number of paths with smaller cost for TSP. We use local optimal path (LOP) to represent this kind of useful paths.
Investigating of Hybrid Meta-Heuristics to Solve the Large-Scale Multi-Source Weber Problems and Performance Measuring of them with Statistical Tests
[www.igi-global.com/chapter/investigating-hybrid-meta-heuristics-solve/69885?camid=4v1a](www.igi-global.com/chapter/investigating-hybrid-meta-heuristics-solve/69885?camid=4v1a)

Privacy Preserving Clustering for Distributed Homogeneous Gene Expression Data Sets
[www.igi-global.com/article/privacy-preserving-clustering-distributed-homogeneous/47316?camid=4v1a](www.igi-global.com/article/privacy-preserving-clustering-distributed-homogeneous/47316?camid=4v1a)