Euclidian Travelling Salesman Problem

Sai Ram Varma Budharaju – UFID :33374276

Abstract

Euclidian Travelling Salesman Problem is a one of the subproblems of the TSP which is considered as NP hard Problem. This paper discusses various approaches to an optimal solution for this problem. The original TSP problem along with various other subproblems of that will be presented in this paper.

Historic and current perspective

The TSP problem is first formulated in a mathematical way by William Rowan Hamilton as a puzzle designed for finding Hamiltonian cycles as recreation game. This problem used to be called as a messenger problem and many scientists and mathematicians tried solving this problem for various applications like school bus routing, tour planning and post-delivery. Many scholars from all most every field of science tried solving this and many instances of this problem but didn't find much success. Richard Karp proved that this problem is a NP complete problem in 1972.Currently there are so many models of this problem few of which are solved during the last century. Euclidean TSP is generally considered to be an easy version of a TSP problem but nevertheless it is shown that it is as tough as TSP. The status of this problem is still open but there are some polynomial time Approximation schemes with some constraints which will be discussed in the paper[1].

Introduction

In this problem, we are essentially provided with n nodes and for each node pair we are provided with the distance in between the nodes given which can be represented as 'd'. We aim to find the closest path that covers all the nodes once and should do that in minimal cost. Mathematically the sum of distance between nodes should be the least which going to all nodes at least once. Although this problem is proved to be NP hard over the decades and computationally hard, there are many versions of this problem which are easy to solve. Metric TSP is one such problem in which the nodes has to obey the triangle inequality principle. Euclidean TSP infact is one of the subproblems of the TSP[2].This problem is usually denoted in form of a graph where cities are the vertices of the graph and paths connecting the cities are edges which connects them. The goal is to find the minimum cost to reach the same vertex as minimum cost.

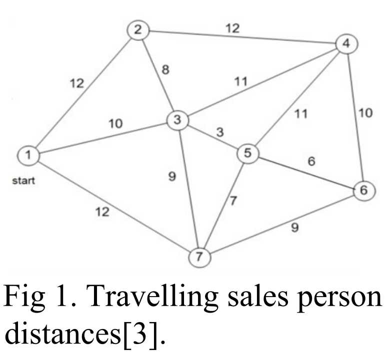

Fig 1. Travelling sales person problem representation with nodes as numerical and their distances[3].

There are numerous versions of TSP. Here we discuss two kinds of TSP before we prove that TSP is a NP complete hard problem. The first one is tour TSP where the salesperson want to return to the original destination. The other special case for this is the Path TSPO where the salesperson need not return to the start point ad only has to find the optimal path to his destination covering all nodes at minimum cost. This model is very much applicable in real life as it has applications in hole drilling and DNA modifying[2].

The TSP problem because of its huge use case is further classified to symmetric and asymmetric. The first case is symmetric where the distance between cities to and from remains same so these problems are usually solved in unweighted graphs. The second type is a case of directed graphs where the distance does not remain same so the algorithm for first type is essentially half the problem as the second one.

The applications of TSP are in many different contexts such as wiring the devices, vehicle routing so that logistics are reached everywhere. DNA mapping is also one of the most common applications of TSP along with tour planning. With these we now move to Euclidean TSP which is a very special case for the TSP.

Euclidean Travelling Salesman Problem

The main motive for developing the Euclidean case comes from case that cities are vertices in a map and follows the Euclidean distance rules. There is very less knowledge about the run time of this case as it looked easier to solve than the original case. The heuristics of this case made it feel like this is easier and the available methods championed but the paper [2] gave a proof which shows that E-TSP is equally hard as the general TSP.

This paper presents the theorems that suggests that these problems are very closely related to each other computationally.

Theorem 1. "*The problems tour TSP and path TSP reduce to each other in linear time by reductions increasing the number of cities by only an additive constant*"[2].

To this this ,consider a new city which is located at a point where distance from that to every other city is at an equal distance. Now the path result is the result of the tour TSP. Let us consider x cities with $a_1 \dots a_x$ and a p that is the largest distance between cities by x times. D is the function to denote paths and tours.

$d'(x_i,x_j)=d(c_i,c_j)$ if $i,j \neq 1$,

$d'(x_i,x_j)=d(c_i,c_j) + 2p$ for all j,

$d'(x_i,x_j)=3k$……….

we know that any optimal path will have a_1 and a_1' as the end points. For this we must establish that those two are not adjacent as adjacent as that will become the largest link. Thus, for every path we will get the optimal path to minimize the original solutions. From the above proof we can say that

The complexity of TSP which has n vertices or cities is $O(n^2 2^n)$

This implies that the complexities of running time are in constant factor of the other algorithm.

Now we try to define the problem in terms of Euclidean geometry. The location of cities is given a pair of coordinates as integers. There is no upper bound for the exact precision of the tour. Let us assume that the new matrix for distances follow some kind of a metric system so that we can easily scale it whenever required. We also take rational points assuming that it will be multiplied by large enough number to make it an integer. The representations are referred as maps in which the distance vertically and horizontally is 1 and 2 respectively.

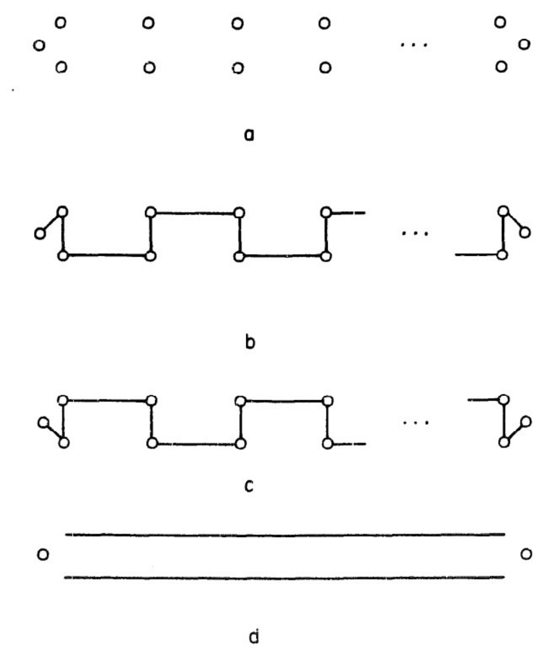

Fig 2. The 2 -chain[2]

From the above figure, A 2 chain can move along the path shown in 1(b) or as in 1(c). This will be expanded and shown from fig1(d).

The distance is always one between any closest cities within the represented chain.

Lemma 1. Among all TSP paths that have two of the cities as endpoints there are four optimal paths of length 32 and the end points are (A,A'),(B,B'),(C,C'),(D,D')[2].

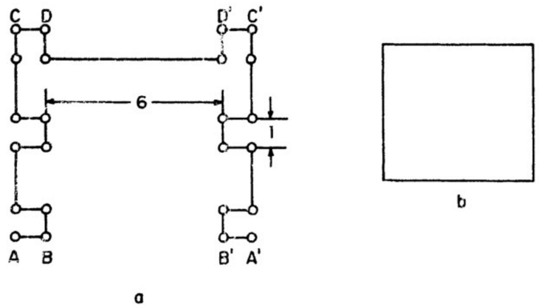

Figure 3: The configuration H

The Exact Cover problem is in a given set of subsets F which is a set of a finite set U, We have to find the subfamily F' consists of disjoint sets and its union gives the finite set U. We choose this problem because this problem is a NP hardness problem. If we can build the Euclidian TSP in such a way that it is scalable to the extra Cover problem, then we can say that E-TSP is a NP hardness problem. Let us consider a set of cities E={c1,c2…cn} and a distance function D, then fort all c belongs to G a subset G of E is called a b component.

Min{d(c,c')}:c'Ï G } >=b

Max{d(c,c'):c'ÎG } <b [2]

If any of the b components are disjoint, that won't affect E so a path is an instance that makes up k node disjoints this is all TSP paths and instance E is b compact if the output path which is optimal as a length (k+1).

Lemma 2. Let us assume we have a TSP in which distance between components is 2a and the rest of it is a compact. let us assume that there is an optimal TSP path where two a compact do not have any connection. let L1 to Ln be the length of 1 path that are optimal and L_0 be the (N+1) path then the optimal ! path length is always greater than the L = $L_0+L_1+\ldots+L_n+2Na$.

Proof : The K_i path traverses G0 by k0 band the length of that path will be $L'=L_0+\ldots L_n+2(k_0-1)a$. The L' is optimal length of the K path and the equality is valid only when Ki=1 and I = 1 to N and all the paths until then are optimal.

From the above two lemmas we can prove with further reducing as the construction is similar and we have enough proof to say that the Euclidean TSP is an NP complete problem whose proof is available in [2].

Though the Euclidian TSP is an NP hard problem, this is one of the easiest versions of the TSP for obtaining the solution with the computer or can be done manually as there are some special properties on which E-TSP is constructed .

Property 1:If the TSP tour intersects itself in the tour, then that solution is not optimal.

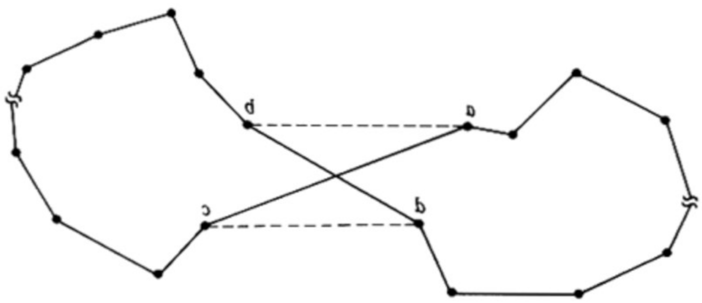

Figure 4. Intersecting links and showing Triangle inequality[4].

We know that sum of lengths of two sides in a triangle is always greater than the third side from the triangle inequality. This holds true for the Euclidean distance between the cities. So, the intersecting links in the map is replaced with lines that are non-intersecting guaranteeing that total length will be less.

Property 2: In a convex hull with m of n points in the TSP ,the order in which the TSP parses should be the same as they are in the convex hull if the solution is optimal.

The proof for this is convex hull is a set of coordinates that can be represented in a 2-dimensional space that can fit all the points in the plane in a smallest polygon possible. This can be directly inferred from the property 1. These properties really go hand in hand when the solutions to TSP are calculated manually and provide basis for some heuristic algorithms for the Euclidean TSP for vehicle routing problems.

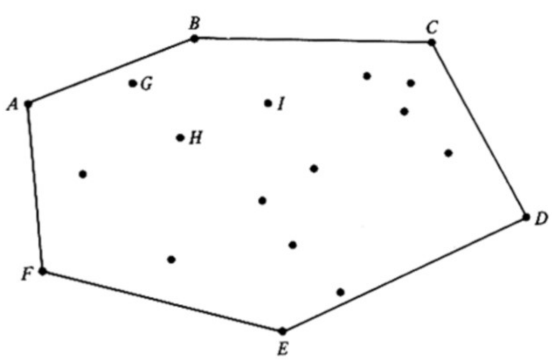

Figure 5. Convex hull of a polygon [4].

Even though the TSP is a NP hard problem we have some polynomial time approximation schemes ,For c>0,when randomized this algorithm can compute (1+1/c) approximation with a runtime of $O(n(\log n)^{O(c)}$. The running time rises to $O(n(\log n)^{(O(c(\mathrm{sqrt}(d))^{d-1}})$ if the nodes are in Âd[1].

This is called a polynomial-time approximation scheme (PTAS).

Structure theorem for Euclidean TSP: For any constant c, minimum internode distance of an instance Âd and the size of bounding box L, and L can be picked randomly Then at least with 50% the dissecton shift has an salesman path with (m,r)-light

Where $m = (O(\mathrm{sqrt}(dc)\log L))^{d-1}$ and $r = (O(\mathrm{sqrt}(dc))^{d-1}$ [1]

The dynamic programming for R^2 is same as discussed above, The 2d tree has $O(2^d n\log n)$ regions and has 2d facets for each region so the total time to enter and leave the region is $m^{O(2dr)}$ so the total run time comes to the

$$O(n(\log n)^{(o(\mathrm{sqrt}dc))^{d-1})}$$ [1].

The dynamic program works by slightly modified divide and conquer algorithm. The instances are well rounded and then the optimal tour with quadtree. So working only on quadtree at a moment, the program modifies the tour until it finds the best available route for the Travelling sales person.

Conclusion:

Through this survey paper on Euclidian Travelling salesperson problem which brushed through the work done and the latest developments and a proof that it an NP hard problem .Along with the general properties, and new Polynomial time approximations for this Euclidean salesperson problem. This paper tried to cover the importance and the status of the problem in the present times.

References

- Arora, Sanjeev (Links to an external site.) (1998), "Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems", *Journal of the ACM* (Links to an external site.), **45** (5): 753–782, doi (Links to an external site.):10.1145/290179.290180 (Links to an external site.), MR (Links to an external site.) 1668147 (Links to an external site.).
- Papadimitriou, Christos H. (Links to an external site.) (1977), "The Euclidean traveling salesman problem is NP-complete", *Theoretical Computer Science*, **4** (3): 237–244, doi (Links to an external site.):10.1016/0304-3975(77)90012-3 (Links to an external site.)

- https://ieeexplore.ieee.org/document/7938898 Travelling sales person for a bi directional graph using Dynamic Programming.
- Larson, Richard C.; Odoni, Amedeo R. (1981), "6.4.7: Applications of Network Models § Routing Problems §§ Euclidean TSP" (Links to an external site.), *Urban Operations Research*, Prentice-Hall, ISBN (Links to an external site.) 9780139394478 (Links to an external site.), OCLC (Links to an external site.) 633142 (Links to an external site.)
- Arora, S., Grigni, M., Karger, D., Klein, P., AND Woloszyn, A. 1998. A polynomial-time approximation scheme for weighted planar graph TSP. In Proceedings of the 9th Annual ACM- SIAM Symposium on Discrete Algorithms. ACM, New York, pp. 33-41.
- Mitchell, J. S. B. (Links to an external site.) (1999), "Guillotine subdivisions approximate polygonal subdivisions: A simple polynomial-time approximation scheme for geometric TSP, *k*-MST, and related problems" (Links to an external site.), *SIAM Journal on Computing*, **28** (4): 1298–1309, doi (Links to an external site.)