

# A Summary of the Current State of the Art Algorithms for the Asymmetric Traveling Salesman Problem

Alex Wong (asw2181) - asw2181@columbia.edu

April 30, 2017

## Abstract

We look at the state of the art algorithm for the general case of the Asymmetric Traveling Salesman problem and its current known inapproximability bounds. The currently best algorithm, by Asadpour et al.[2], details an  $O(\log n / \log \log n)$ -approximation algorithm for the general case of ATSP for costs satisfying the triangle inequality. An algorithm by Karpinski et al.[9] gives us a hardness of approximation bounds of  $75 / 74$  for ATSP. We also briefly touch on some constant factor approximation algorithms and recently new heuristics for special cases of ATSP, including an approach by Svensson et al.[13] that details a constant factor approximation algorithm for a special case of ATSP that contains only two different edge weights of arbitrary weight where the costs satisfy the triangle inequality.

## 1 Introduction

The traveling salesman problem is one of the most well-known and studied problems in theoretical computer science and operations research. Many real world problems, from delivery routing to latency networks to DNA sequencing, can be framed as some variation of the traveling salesman problem. In this paper, we look at the current state of the art algorithms for the asymmetric traveling salesman problem (ATSP) where the costs satisfy the triangle inequality (costs from some point  $u$  to point  $w$  is at most the cost from point  $u$  to point  $v$  + cost from point  $v$  to point  $w$ ) but the cost from some point  $u$  to point  $v$  does not necessarily equal the cost from point  $v$  to point  $u$ . From this description, we can frame the symmetric traveling salesman problem (the cost from some point  $u$  to point  $v$  does equal the cost from point  $v$  to point  $u$ ) as a special case of ATSP.

In Section 2, we provide an overview of an  $O(\log n / \log \log n)$ -approximation algorithm for the general case of ATSP for costs satisfying the triangle inequality which comes from the work of Asadpour et al. [2]. The  $O(\log n / \log \log n)$ -approximation algorithm introduces the concept of the "thinness" of a spanning tree where the thinness correlates with the approximation bound of ATSP. This is an improvement over the decades-long standing  $\Theta(\log n)$ -approximation bound given by the work of Frieze et al. [5].

In section 4

Lastly, in section 5, we conclude with opinions on the hypothesized next steps towards a constant factor approximation algorithm for ATSP.

## 2 An $O(\log n / \log \log n)$ -approximation algorithm for ATSP

Algorithm 1, on the next page, details the  $O(\log n / \log \log n)$ -approximation algorithm for ATSP (from [2] section 1). Throughout the rest of this section, we will divide each step of the algorithm into subsection(s) to describe it in more detail and/or prove its correctness.

---

**Algorithm 1** An  $O(\log n / \log \log n)$ -approximation algorithm for ATSP [2]

---

**Input:** A set  $V$  consisting of  $n$  points and a cost function  $c : V \times V \rightarrow \mathbb{R}^+$  where the costs satisfy the triangle inequality.

**Output:** An  $O(\frac{\log n}{\log \log n})$ -approximation to the ATSP instance described by the inputs  $V$  and  $c$ .

1. Solve the Held-Karp LP relaxation of the ATSP instance to get an optimum extreme point solution  $\mathbf{x}^*$ . Create a symmetrized and scaled down version of  $\mathbf{x}^*$  from (5) and define it as  $\mathbf{z}^*$ .  $\mathbf{z}^*$  is a vector which can be interpreted as a point in the relative interior (not touching any boundaries) of the spanning tree polytope  $P$  created from an undirected graph supported by  $\mathbf{x}^*$  where we disregard the directions of arcs. Another way to think of  $\mathbf{z}^*$  is the marginal probabilities on the edges  $z_e^*$  of an exponential distribution on spanning trees  $\tilde{p}(\cdot)$ .

2. Let  $E$  be the support graph of  $\mathbf{z}^*$  where we disregard the directions of arcs. We then find weights  $\{\tilde{\gamma}\}_{e \in E}$  such that  $\tilde{p}(T)$  is approximately proportional to  $\exp(\sum_{e \in T} \tilde{\gamma}_e)$  where, for any edge  $e \in E$ ,

$$\sum_{T \in \mathcal{T}: T \ni e} \tilde{p}(T) \leq (1 + \epsilon) z_e^*, \text{ for a small value } \epsilon.$$

3. Sample  $\Theta(\log n)$  spanning trees  $T_i$  from  $\tilde{p}(\cdot)$ . Orient all the edges of the sampled spanning trees to minimize its cost since the spanning tree only has to be weakly connected. Let  $\vec{T}$  represent an oriented tree. Let  $T^*$  be the spanning tree that has minimum total cost amongst all the sample spanning trees.

4. Find a minimum cost integral circulation that contains the oriented tree  $\vec{T}^*$ . This will give us a multigraph. Shortcut the multigraph to get a Eulerian tour and output the result.

---

## 2.1 Preliminaries and Notation

In this subsection, we describe much of the notation that will be used throughout the rest of this section. We will also define the notion of "thinness" of a spanning tree.

Let  $a = (u, v)$  be the arc (directed edge) from  $u$  to  $v$  and let  $e = (u, v)$  be an undirected edge. Then,  $A$  (resp.  $E$ ) is the set of arcs (resp. edges) in a directed (resp. undirected) graph  $G$ .

For a given function  $f : A \rightarrow \mathbb{R}$ , the cost of  $f$  is defined as  $c(f) := \sum_{a \in A} c(a)f(a)$ . For some set of arcs  $S \subseteq A$ , we define  $f(S) := \sum_{a \in S} f(a)$ . The same notation is used for the edge set  $E$  of an undirected graph.

For a subset of vertices  $U \subseteq V$ , we define

$$\delta^+(U) := \{a = (u, v) \in A : u \in U, v \notin U\},$$

$$\delta^-(U) := \{a = (u, v) \in A : u \notin U, v \in U\},$$

$$\delta(U) := \delta^+(U) \cup \delta^-(U),$$

$$A(U) := \{a = (u, v) \in A : u \in U, v \in U\}$$

to be the set of arcs that are leaving, entering, and contained in  $U$ , respectively. We also define  $\delta^+(v) := \delta^+(\{v\})$  and  $\delta^-(v) := \delta^-(\{v\})$  for each single vertex  $v$ . For an undirected graph,  $\delta(U)$  represents the set of edges with just one endpoint in  $U$  and  $E(U)$  represents the edges that are contained within  $U$ . Lastly, all  $\log$  in equations represents the natural logarithm.

We say that a spanning tree  $T$  is  $\alpha$ -thin with respect to  $\mathbf{z}$  if and only if:

$$|T \cap \delta(U)| \leq \alpha \cdot z(\delta(U)) \quad \forall U \subset V$$

which essentially states that for all the subsets of vertices  $U$  of the spanning tree  $T$ , the number of edges in the cut between  $U$  and  $V \setminus U$  must be  $\leq$  the thinness multiplied by the cost of the edges in the cut.

We also say that  $T$  is  $(\delta, s)$ -thin with respect to  $\mathbf{z}$  if and only if it is  $\alpha$ -thin and  $c(T) \leq s \cdot OPT_{HK}$ , which essentially states that the cost of  $T$  is at most  $s$  times the cost of the optimal Held-Karp solution.

By showing that a spanning tree is "thin", we would be able to get a Eulerian augmentation of the spanning tree where the total cost is within a factor of  $\alpha$  of the cost  $OPT_{HK}$ , which implies within the same factor of the optimum solution.

## 2.2 The Held-Karp Relaxation

We detail and prove step 1 of the algorithm. Given an ATSP instance with a cost function  $c : V \times V \rightarrow \mathbb{R}^+$ , we can get a lower bound on the optimal value of ATSP by the following linear programming (LP) relaxation defined on the complete bidirected graph over the vertex set  $V$ :

$$\min \sum_a c(a) x_a \quad (1)$$

$$s.t. \quad \mathbf{x}(\delta^+(U)) \geq 1 \quad \forall U \subset V, \quad (2)$$

$$\mathbf{x}(\delta^+(v)) = \mathbf{x}(\delta^-(v)) = 1 \quad \forall v \in V, \quad (3)$$

$$x_a \geq 0 \quad \forall a.$$

This relaxation is accurate since (2) makes sure that all the vertices are strongly connected and (3) makes sure that the indegree = outdegree for all vertices of  $V$  which makes this a Eulerian graph. This relaxation is known as the Held-Karp relaxation [8] and it's well-known that an optimal solution  $\mathbf{x}^*$  to the relaxation can be computed in polynomial time (using the ellipsoid method). Thus, we can say that  $c(\mathbf{x}^*) = OPT_{HK}$ . Also, notice that (3) implies that any feasible solution  $\mathbf{x}$  satisfies

$$\mathbf{x}(\delta^+(U)) = \mathbf{x}(\delta^-(U)) \quad \forall U \subset V. \quad (4)$$

## 2.3 Proving $\mathbf{z}^* \in \text{Relative Interior of } P$

As mentioned in the algorithm,  $\mathbf{z}^*$  is the symmetrized and scaled down version of  $\mathbf{x}^*$ . So, we formally define

$$\mathbf{z}_{\{u,v\}}^* := \frac{n-1}{n} (\mathbf{x}_{uv}^* + \mathbf{x}_{vu}^*). \quad (5)$$

As mentioned in step 2 of the algorithm, we let  $E$  be the support of  $\mathbf{z}^*$ . We also let  $A$  be the support of  $\mathbf{x}^*$  where  $A = \{(u, v) : x_{uv}^* > 0\}$ . We also define the costs of all edges  $e \in E$  where  $c(e) = \min\{c(a) : a \in \{(u, v), (v, u)\} \cap A\}$ ,  $\forall e \in E$ . This shows that  $c(\mathbf{z}^*) < c(\mathbf{x}^*)$ . Now we can prove the rest of step 1 of the algorithm and define a lemma:

**Lemma 2.1:** The vector  $\mathbf{z}^*$  belongs to the relative interior of the spanning tree polytope  $P$ .

**Proof:** From the characterization of the base polytope of a matroid given by Edmonds[4], it follows that the spanning tree polytope  $P$  is defined by the following inequalities (see Corollary 50.7c of [11]):

$$P = \{z \in \mathbb{R}^E : z(E) = |V| - 1, \quad (6)$$

$$z(E(U)) \leq |U| - 1 \quad \forall U \subset V \text{ where } U \neq \emptyset, \quad (7)$$

$$z_e \geq 0 \quad \forall e \in E. \quad (8)$$

From this, we see that  $\mathbf{z}^*$  satisfies inequality (6) because:

$$\forall v \in V, \mathbf{x}^*(\delta^+(v)) = 1 \Rightarrow \mathbf{x}^*(A) = n = |V| \Rightarrow \mathbf{z}^*(E) = n - 1 = |V| - 1$$

## 2.4 Maximum Entropy Distribution

In the latter part of step 2 of the algorithm, to find the weights  $\tilde{\gamma}_e, \forall e \in E$ , requires lengthy non-trivial calculations which will not be detailed within this paper. There are two methods to find  $\tilde{\gamma}_e$ 's: either the combinatorial approach (see [2] section 7) where the basic idea is to iteratively tweak the weights of  $\tilde{\gamma}_e$  until their marginals  $= (1 + \epsilon/2)z_e$  or the ellipsoid method (see [2] section 8) where the basic idea is to find a near optimal solution of the dual of the convex program.

## 3 Inapproximability Bounds of ATSP

## 4 Special Cases of ATSP

### 4.1 ATSP on Graphs with Bounded Genus

### 4.2 Polyloglogn Integrality Gap of ATSP

### 4.3 Local-Connectivity ATSP

## 5 Concluding Remarks

## References

- [1] ANARI, N., AND GHARAN, S. O. Effective-resistance-reducing flows, spectrally thin trees, and asymmetric TSP. *In: Proceedings of FOCS* (2015).
- [2] ASADPOUR, A., GOEMANS, M. X., MADRY, A., GHARAN, S. O., AND SABERI, A. An  $\mathcal{O}(\log n / \log \log n)$ -approximation algorithm for the asymmetric traveling salesman problem. *In: Proceedings of SODA* (2010), pages 379–389.
- [3] CHRISTOFIDES, N. Worst-case analysis of a new heuristic for the traveling salesman problem. *Technical Report, DTIC Document* (1976).
- [4] EDMONDS, J. Matroids and the greedy algorithm. *Mathematical Programming* (1971), pages 127–136.
- [5] FRIEZE, A. M., GALBIATI, G., AND MAFFIOLI, F. On the worse-case performance of some algorithms for the asymmetric traveling salesman problem. *Networks* (1982), 12:23–39.
- [6] GHARAN, S. O., AND SABERI, A. The asymmetric traveling salesman problem on graphs with bounded genus. *In: Proceedings of SODA* (2011), pages 967–975.
- [7] GHARAN, S. O., SABERI, A., AND SINGH, M. A randomized rounding approach to the traveling salesman problem. *In: Proceedings of FOCS* (2011).
- [8] HELD, M., AND KARP, R. The traveling salesman problem and minimum spanning trees. *Operations Research* (1970), 18:1138–1162.
- [9] KARPINSKI, M., LAMPIS, M., AND SCHMIED, R. New approximability bounds for TSP. *J. Comput. Syst. Sci.* (2015), 81(8):1665–1667.
- [10] PAPADIMITRIOU, C. H., AND YANNAKAKIS, M. The traveling salesman problem with distances one and two. *Math. Oper. Res.* (1993), 18(1):1–11.
- [11] SCHRIJVER, A. Combinatorial optimization. *vol. 2 of Algorithms and Combinatorics* (2003).

- [12] SVENSSON, O. Approximating ATSP by relaxing connectivity. *In: Proceedings of FOCS* (2015).
- [13] SVENSSON, O., TARNAWSKI, J., AND VÉGH, L. A. Constant factor approximation for ATSP with two edge weights. *In: Proceedings of IPCO* (2016), pages 226–237.