A (Slightly) Improved Bound on the Integrality Gap of the Subtour LP for TSP

Anna Karlin
Computer Science
University of Washington
Seattle, USA
karlin@cs.washington.edu

Nathan Klein
Computer Science
University of Washington
Seattle, USA
nwklein@cs.washington.edu

Shayan Oveis Gharan
Computer Science
University of Washington
Seattle, USA
shayan@cs.washington.edu

Abstract—In this extended abstract, we show that for some \( \epsilon > 10^{-36} \) and any metric TSP instance, the max entropy algorithm studied by [1] returns a solution of expected cost at most \( \frac{3}{2} - \epsilon \) times the cost of the optimal solution to the subtour elimination LP. This implies that the integrality gap of the subtour LP is at most \( \frac{3}{2} - \epsilon \).

This analysis also shows that there is a randomized \( \frac{3}{2} - \epsilon \) approximation for the 2-edge-connected multi-subgraph problem, improving upon Christofides’ algorithm.

I. Introduction

One of the most fundamental problems in combinatorial optimization is the traveling salesperson problem (TSP), formalized as early as 1832 (c.f. [2, Ch 1]). In an instance of TSP we are given a set of \( n \) cities \( V \) along with their pairwise symmetric distances, \( c : V \times V \rightarrow \mathbb{R}_{\geq 0} \). The goal is to find a Hamiltonian cycle of minimum cost. In the metric TSP problem, which we study here, the distances satisfy the triangle inequality. Therefore, the problem is equivalent to finding a closed Eulerian connected walk of minimum cost.

It is NP-hard to approximate TSP within a factor of \( \frac{123}{122} \) [3]. An algorithm of Christofides-Serdyukov [4], [5] from four decades ago gives a \( \frac{3}{2} \)-approximation for TSP. Over the years there have been numerous attempts to improve the Christofides-Serdyukov algorithm and exciting progress has been made for various special cases of metric TSP, e.g., [6], [7], [8], [9], [10], [11], [12], [13]. Recently, [1] gave the first improvement for the general case by demonstrating that the so-called “max entropy” algorithm of [6] gives a randomized \( \frac{3}{2} - \epsilon \) approximation for some \( \epsilon > 10^{-36} \).

The method introduced in [1] exploits the optimum solution to the following linear programming relaxation of metric TSP studied by [14], [15], [16], also known as the subtour elimination LP:

\[
\min \sum_{u, v} x_{\{u, v\}} c(u, v)
\]

s.t.,

\[
\sum_u x_{\{u, v\}} = 2 \quad \forall v \in V,
\]

\[
\sum_{u \in S, v \notin S} x_{\{u, v\}} \geq 2 \quad \forall S \subseteq V, S \neq \emptyset
\]

\[
x_{\{u, v\}} \geq 0 \quad \forall u, v \in V.
\]

However, [1] did not show that the integrality gap of the subtour elimination polytope is bounded below \( \frac{3}{2} \), and therefore did not make progress towards the “4/3 conjecture” which posits that the integrality gap of LP (1) is \( \frac{3}{4} \). In this work we remedy this discrepancy by proving the following theorem, improving upon the bound of \( \frac{3}{2} \) from Wolsey [17] in 1980:

Theorem I.1. Let \( x \) be a solution to LP (1) for a TSP instance. For some absolute constant \( \epsilon > 10^{-36} \), the max entropy algorithm outputs a TSP tour with expected cost at most \( \frac{3}{2} - \epsilon \) times the cost of \( x \). Therefore the integrality gap of the subtour elimination LP is at most \( \frac{3}{2} - \epsilon \).

To prove Theorem I.1, we amend Section 4 of [1] but keep the remainder of the analysis essentially the same. Unlike [1], this argument now preserves the integrality gap by avoiding the use of the optimum solution in bounding the cost of the matching. See Section III for a discussion of our new approach.
A. Other Consequences

a) Path TSP: In recent exciting work, Traub, Vygen, Zenklusen [18] showed that an α-approximation algorithm for metric TSP can be used as a black box to get a α(1 + ε) approximation algorithm for Path TSP. Their work together with [1] implies that there is a 3/2 − ε approximation algorithm for Path TSP (for ε > 10−36). On the other hand, it is a folklore result that the integrality gap of the natural LP relaxation of Path TSP is at least 3/2. Therefore, a consequence of the above theorem is that although the best possible approximation factors of the two problems are the same (up to polynomial reductions), the natural LP relaxation of metric TSP has a strictly smaller integrality gap.

b) 2-ECSM: In the 2-edge-connected multi-subgraph problem, or 2-ECSM for short, we are given a weighted graph G and we want to find a minimum cost 2-edge-connected spanning subgraph, where an edge can be chosen multiple times. The classical Christofides-Serdyukov algorithm gives a 3/2-approximation for 2-ECSM and despite significant attempts [19], [20], [21], [22] improved algorithms were designed only for special cases of the problem. Since in [16] it is shown that LP (1) is a valid relaxation for 2-ECSM, we obtain:

**Corollary I.2.** For some absolute constant ε > 10−36 the max entropy algorithm is a randomized 3/2 − ε approximation for the 2-edge-connected multi-subgraph problem.

B. New techniques and contributions

This paper can be seen as a case study on how to reason about and deal with near minimum cuts. One can deduce from the classical cactus representation of a graph G [23] (i) the structure of all min cuts of G and (ii) the structure of the edges of G in the sense that every edge {u, v} maps to a unique path in the cactus between the images of u and v. Furthermore, such a path intersects every cycle of the cactus on at most one cactus edge. The theory has found many application from designing fast algorithms [24], [25] to the analysis of approximation algorithms for TSP [11] and connectivity augmentation [26], [27].

Two decades later, the theory of min cuts was extended to near min cuts in works of Benczúr and Goemans [28], [29] where they introduced the polygon representation which represents all cuts of a graph with at most $\frac{k}{2}$ edges, where k is its edge connectivity. Although these works completely classify the structure of all near min cuts of a given graph G, they do not characterize the structure of the edges of G with respect to these cuts, which can be important in applications (for example, in many of the recent applications of min cuts, one also needs to exploit the structure of the edges in relation to the cactus). The structure on the edges turns out to be highly relevant in this work as well, and as a byproduct of our analysis we make progress towards classifying the way in which the edges of G relate to the structure of the polygon representation.

For motivation, consider a generic family of network design problems in which we want to construct a network such that every pair u, v of vertices has connectivity at least c_{u,v}. A natural approach is to write an LP relaxation to find a (minimum cost) vector $x : E \rightarrow \mathbb{R}_{\geq 0}$ such that for every set S separating u and v, $x(\delta(S)) \geq c_{u,v}$. We can round this LP using independent rounding or a dependent rounding scheme such as sampling from max entropy distributions. Using classical concentration bounds one can show that if $x(\delta(S)) \geq c_{u,v}$ then with high probability the rounded solution has at least $c_{u,v}$ edges across this cut. So the main challenge is to “fix” near tight cuts, i.e., cuts where $x(\delta(S)) \approx c_{u,v}$. For an explicit instantiation of this scheme see [30]. A better understanding of the global structure of the family of near tight cuts has the potential to significantly simplify or even improve the approximation factor of such rounding algorithms. A classical technique to design algorithms for such network design problems is to apply uncrossing to extreme point solutions of the LP. One can view our contribution as an approximate uncrossing technique that deals with all near tight cuts (instead of just tight cuts) as we explain next.

a) An Approximate Uncrossing Technique.: A fundamental technique in the field of approximation algorithms is the uncrossing technique of Jain [32]. Given a graph $G = (V, E)$, a weight vector $x : E \rightarrow \mathbb{R}_{\geq 0}$, and a function $f : V \rightarrow \mathbb{R}$, suppose that $x(\delta(S)) \geq f(S)$ for all $S \subseteq V$. Let $\mathcal{N}$ be the family of sets S such that $x(\delta(S)) = f(S)$, i.e., the family of tight sets with respect to f. The uncrossing technique says that if f is (weakly) supermodular then we can refine $\mathcal{N}$ to a laminar family of sets, $\mathcal{H}$, such that if all sets of $\mathcal{H}$ are tight, then all sets of $\mathcal{N}$ are tight as well. For a concrete example, 1

---

1See e.g. [31] for a number of applications of this technique.
suppose \( f \) is a constant function, say \( f(S) = 2 \) for all \( \emptyset \subseteq S \subseteq V \). Then, sets of \( H \) can be constructed using the cactus representation [23] of cuts in \( \mathcal{N} \). The significance of this method is that if \( x \) is a basic feasible solution to a LP with constraints \( x(\delta(S)) \geq f(S) \) for all \( S \), one can use this machinery to argue that the support of \( x \) has size \( O(|V|) \).

Informally, we prove the following, which can be seen as an approximate uncrossing technique:

**Theorem I.3 (Informal).** Suppose we have a vector \( x : E \to \mathbb{R}_{\geq 0} \) such that \( x(\delta(S)) \geq f(S) \) for all \( S \); define \( \mathcal{N} \) to be sets \( S \) where \( x(\delta(S)) \leq f(S)(1 + \epsilon) \) for some fixed \( \epsilon > 0 \). If \( f(.) \) is constant, say \( f(S) = 2 \) for all \( S \), then there is a set \( \mathcal{N}^* \subseteq \mathcal{N} \) and a collection of edge sets \( F_1, \ldots, F_m \subseteq E \) such that the following hold:

- \( |\mathcal{N}^*| = O(|V|) \), \( m = O(|V|) \).
- \( x(F_i) \geq 0 - \epsilon \) for all \( 1 \leq i \leq m \).
- Every edge \( e \) is in at most \( O(1) \) of the \( F_i \)’s.
- For every set \( S \in \mathcal{N} \setminus \mathcal{N}^* \) there exists \( 1 \leq i < j \leq m \) such that \( F_i \cap F_j = \emptyset \) and \( F_i \cup F_j \subseteq \delta(S) \) and for every \( S \in \mathcal{N}^* \), there exists \( 1 \leq i \leq m \) such that \( F_i \subseteq \delta(S) \).

In words, although we cannot simply refine \( \mathcal{N} \) to a linear number of sets, we can refine the edges in cuts of \( \mathcal{N} \) to a linear number of sets \( F_1, \ldots, F_m \) such that we can essentially capture the edges of \( \delta(S) \) for any \( S \in \mathcal{N} \setminus \mathcal{N}^* \) by a pair of disjoint \( F_i \)’s. We give a slightly weaker condition for cuts in \( \mathcal{N}^* \); namely we only capture half of their edges by \( F_i \)’s.

**Example I.4.** For a simple example of the above theorem, suppose \( \epsilon = 0 \), i.e. \( \mathcal{N} \) is the set of min cuts of a graph \( G \). Furthermore, suppose that every proper cut in \( \mathcal{N} \) is crossed (recall that \( S \) is proper if \( 1 < |S| < |V| - 1 \)) and that \( \mathcal{N} \) has at least one proper cut. Then, one can use an uncrossing technique, namely that if \( A, B \in \mathcal{N} \) then \( A \cap B \in \mathcal{N} \), to prove that \( G \) must be cycle, namely we can order vertices of \( G, v_0, \ldots, v_{|V| - 1} \) such that \( x(v_0, v_i_{\mod |V|}) = 1 \). In such a case we let \( \mathcal{N}^* = \emptyset \) and \( F_i = E(v_i, v_{i_{\mod |V|}}) \).

**Example I.5.** For a second example, suppose again \( \epsilon = 0 \) and \( \mathcal{N} \) is the set of min cuts of a graph \( G \) where \( \mathcal{N} \) forms a laminar family (no two cuts cross). It turns out that we cannot decompose edges of cuts of \( \mathcal{N} \) into a linear sized collection of sets where every edge appears only a constant number of times. The main reason is that some edges may appear in an unbounded number of cuts. In this case we let \( \mathcal{N}^* = \mathcal{N} \) and for every \( A \in \mathcal{N} \) (with immediate parent \( B \in \mathcal{N} \) in the laminar family) we add a set \( F_A = \delta(A) \setminus \delta(B) \) to our collection. It is straightforward to show, using the structure of min cuts, that \( x(F_A) \geq 1 \); furthermore, since the size of a laminar family is linear in \( V \), this gives a valid decomposition in the sense of above theorem.

Lastly, if \( \epsilon = 0 \) and \( \mathcal{N} \) is the set of min cuts of an arbitrary graph, one can represent all min cuts of \( \mathcal{N} \) by a cactus [23] which can be seen as a tree of cycles. In such a case, one can use a construction similar to Example I.4 for each cycle where instead of a vertex \( v_i \) we have a set \( a_i \subseteq V \) and one similar to Example I.5 for the tree part of the cactus. For a concrete application of such a decomposition of min cuts see [11].

One of the main challenges in dealing with near min cuts relative to min cuts is that if \( x(\delta(A)), x(\delta(B)) \leq 2 + \epsilon \) then \( x(\delta(A \cap B)) \leq 2 + 2\epsilon \). Therefore, if \( \epsilon = 0 \), then min cuts are closed under intersection, set difference and union, but this is no longer true when \( \epsilon > 0 \). So, to employ the classical uncrossing machinery one should be very careful to “uncross” only a constant number of times (independent of \( \epsilon \)) to make sure that every cut remains within \( 2 + O(\epsilon) \). This is the main reason that the polygon representation of near min cuts (see below) is more sophisticated, e.g., we can no longer argue \( x(E(a_i, a_{i+1})) \approx 1 \), see Fig. 5.

Although we don’t study it here, we believe it may be worthwhile to find generalizations of Theorem I.3 which hold for any (weakly) supermodular function.

**Remark I.6.** We do not explicitly prove Theorem I.3 in this extended abstract, as it is not used to prove Theorem I.1. However it can be deduced from arguments in the full version of the paper.

**b) Extensions to the Polygon Representation:** To obtain our uncrossing framework we prove new properties of the polygon representation. Given a graph \( G = (V, E) \), let \( k \) be the edge-connectivity of \( G \), i.e. the number of edges in a minimum cut of \( G \). For \( \epsilon > 0 \), consider the set of \( (1 + \epsilon) \)-near minimum cuts of \( G \); cuts \( (S, \overline{S}) \) where \( |E(S, \overline{S})| < (1 + \epsilon)k \). Benczúr [28] and Benczúr, Goemans [29] proved that if \( \epsilon \leq 1/5 \) then the near minimum cuts of \( G \) admit a polygon representation. Namely, every connected component \( \mathcal{C} \) of crossing \((1 + \epsilon)\) near min cuts can be represented by the diagonals of a convex polygon. In this polygon, the vertices of \( G \) are partitioned into sets called atoms, and every atom is mapped to a cell of this polygon defined by the diagonals and the boundary of the polygon itself.
The polygon representation can be seen as a generalization of the well-known cactus representation [23] of minimum cuts where a cycle of the cactus is replaced by a convex polygon. Unlike a cycle, some vertices/atoms map to the interior of the polygon, which are called “inside” atoms. The inside atoms at first look like a mystery and one can ask many questions about them such as how many can exist and what structures they can exhibit.

Here, we explain two lemmas we proved which might find further applications beyond TSP in the future. First, we give a necessary condition for a cell of a polygon to contain an inside atom:

**Lemma I.7 (Informal).** Consider a polygon $P$ for a connected component $C$ of a family of $1 + e$ near min cuts for $e \leq 1/5$ (where representing diagonals correspond to cuts in $C$). Any cell of $P$ that has an inside atom must have at least $\Omega(1/e)$ many sides.

This can be seen as a generalization of [29, Lem 22] to the case in which the cell is allowed to be adjacent to vertices of the polygon $P$.

Now, we explain our second extension: it follows from the cactus representation of minimum cuts that for a graph $G$ and a min cut $S$ one can partition the set of all min cuts that cross $S$ into two groups $A = \{A_1, \ldots, A_k\}$ and $B = \{B_1, \ldots, B_l\}$ for some $k, l \geq 0$ such that $S \cap A_1 \subseteq S \cap A_2 \subseteq \cdots \subseteq S \cap A_k$ and, similarly, $S \cap B_1 \subseteq \cdots \subseteq S \cap B_l$. We prove a generalization of this fact for near min cuts:

**Lemma I.8 (Informal).** Consider the set of $1 + e$ near min cuts of a graph $G$ for $e \leq 1/10$; for any such near min cut $S$, one can partition the $1 + e$ near min cuts crossing $S$ into two groups $A = \{A_1, \ldots, A_k\}$ and $B = \{B_1, \ldots, B_l\}$ such that $S \cap A_1 \subseteq S \cap A_2 \subseteq \cdots \subseteq S \cap A_k$ and similarly for cuts in $B$.

C. Outline of rest of paper

After reviewing preliminaries in Section II, we give a high-level overview of our proof technique in Section III, deferring the details to the full version. The main new technical contributions of this paper involve the uncrossing of constraints using the polygon representation. The remaining content of the paper essentially follows from [1].

II. Preliminaries

In the interest of getting quickly to the overview in Section III, on a first pass reading of this paper, the reader may wish to skip over the (short) proofs later in this section.

A. Algorithm

Let $x^0$ be an optimum solution of LP (1). Without loss of generality we assume $x^0$ has an edge $e_0 = \{u_0, v_0\}$ with $x^0_{e_0} = 1, c(e_0) = 0$. (To justify this, consider the following process: given $x^0$, pick an arbitrary node, $u$, split it into two nodes $u_0, v_0$ and set $x_{\{u_0, v_0\}} = 1, c(e_0) = 0$ and assign half of every edge incident to $u$ to $u_0$ and the other half to $v_0$.)

Let $E_0 = E \cup \{e_0\}$ be the support of $x^0$ and let $x$ be $x^0$ restricted to $E$ and $G = (V, E)$. Note $x^0$ restricted to $E$ is in the spanning tree polytope (2) of $G$.

For a vector $\lambda : E \rightarrow R_{\geq 0}$, a $\lambda$-uniform distribution $\mu_\lambda$ over spanning trees of $G = (V, E)$ is a distribution where for every spanning tree $T \subseteq E$, $\mathbb{P}_{\mu_\lambda}[T] = \frac{\prod_{e \in T} \lambda_e}{\sum_{T' \subseteq E} \prod_{e \in T'} \lambda_e}$. The second step of the algorithm is to find a vector $\lambda$ such that for every edge $e \in E$, $\mathbb{P}_{T \sim \mu_\lambda}[e \in T] = x_e(1 \pm e)$, for some $e < 2^{-n}$. Such a vector $\lambda$ can be found using the multiplicative weight update algorithm [33] (see Theorem II.1) or by applying interior point methods [9] or the ellipsoid method [33]. (We note that the multiplicative weight update method can only guarantee $e < 1/\text{poly}(n)$ in polynomial time.)

Finally, similar to Christodides’ algorithm, we sample a tree $T \sim \mu_\lambda$ and then add the minimum cost matching on the odd degree vertices of $T$.

**Algorithm 1** Max Entropy Algorithm for TSP

Find an optimum solution $x^0$ of Eq. (1), and let $e_0 = \{u_0, v_0\}$ be an edge with $x^0_{e_0} = 1, c(e_0) = 0$. Let $E_0 = E \cup \{e_0\}$ be the support of $x^0$ and $x$ be $x^0$ restricted to $E$ and $G = (V, E)$. Find a vector $\lambda : E \rightarrow R_{\geq 0}$ such that for any $e \in E$, $\mathbb{P}_{T \sim \mu_\lambda}[e \in T] = x_e(1 \pm 2^{-n})$. Sample a tree $T \sim \mu_\lambda$. Let $M$ be the minimum cost matching on odd degree vertices of $T$. Output $T \cup M$.

The above algorithm from [1] is a slight modification of the algorithm proposed in [6]. While the proof of Theorem I.1 heavily utilizes properties of max entropy trees, we note that the new arguments in this paper only use the fact that the spanning tree distribution respects the marginals of $x$.

**Theorem II.1 ([33]).** Let $z$ be a point in the spanning tree polytope (see (2)) of a graph $G = (V, E)$. For any $e > 0$, a vector $\lambda : E \rightarrow R_{\geq 0}$ can be found such that
the corresponding \(\lambda\)-uniform spanning tree distribution, \(\mu_\lambda\), satisfies
\[
\sum_{T \subseteq \mathcal{T}} \mathbf{P}_{\mu_\lambda}[T] \leq (1 + \varepsilon)z_e, \quad \forall e \in E,
\]
i.e., the marginals are approximately preserved. In the above \(\mathcal{T}\) is the set of all spanning trees of \((V, E)\). The running time is polynomial in \(n = |V|, -\log \min_{e \in E} z_e\) and \(\log(1/\varepsilon)\).

B. Notation

We write \([n] := \{1, \ldots, n\}\) to denote the set of integers from 1 to \(n\). For a set of edges \(A \subseteq E\) and (a tree) \(T \subseteq E\), we write\(^2\)
\[
A_T = |A \cap T|.
\]
For a set \(S \subseteq V\), we write
\[
E(S) = \{\{u, v\} \in E : u, v \in S\}
\]
to denote the set of edges in \(S\) and we write
\[
\delta(S) = \{\{u, v\} \in E : |\{u, v\} \cap S| = 1\}
\]
to denote the set of edges that leave \(S\). For two disjoint sets of vertices \(A, B \subseteq V\), we write
\[
E(A, B) = \{\{u, v\} \in E : u \in A, v \in B\}.
\]
For a set \(A \subseteq E\) and a function \(x : E \to \mathbb{R}\) we write
\[
x(A) := \sum_{e \in A} x_e.
\]
For two sets \(A, B \subseteq V\), we say \(A\) crosses \(B\) if all of the following sets are non-empty:
\[
A \cap B, A \setminus B, B \setminus A, \overline{A \cup B}.
\]
We write \(G = (V, E, x)\) to denote an (undirected) graph \(G\) together with special vertices \(u_0, v_0\) and a weight function \(x : E \to \mathbb{R}_{\geq 0}\). Similarly, let \(G_0 = (V, E_0, x^0)\) and let \(G_{/e_0} = G_0 / \{e_0\}\), i.e. \(G_{/e_0}\) is the graph \(G_0\) with the edge \(e_0\) contracted.

C. Polyhedral background

For any graph \(G = (V, E)\), Edmonds [34] gave the following description for the convex hull of spanning trees of a graph \(G = (V, E)\), known as the spanning tree polytope.
\[
z(E) = |V| - 1
\]
\[
z(E(S)) \leq |S| - 1 \quad \forall S \subseteq V
\]
\[
z_e \geq 0 \quad \forall e \in E.
\]

Edmonds [34] proved that the extreme point solutions of this polytope are the characteristic vectors of the spanning trees of \(G\).

Fact II.2. Let \(x^0\) be a feasible solution of (1) such that \(x^0_{e_0} = 1\) with support \(E_0 = E \cup \{e_0\}\). Let \(x\) be \(x^0\) restricted to \(E\); then \(x\) is in the spanning tree polytope of \(G = (V, E)\).

Since \(c(e_0) = 0\), the following fact is immediate.

Fact II.3. Let \(G = (V, E, x)\) where \(x\) is in the spanning tree polytope. If \(\mu\) is any distribution of spanning trees with marginals \(x\) then \(\mathbb{E}_{T \sim \mu}[c(T \cup e_0)] = c(x)\).

To bound the cost of the min-cost matching on the set \(O\) of odd degree vertices of the tree \(T\), we use the following characterization of the \(O\)-join polyhedron\(^3\) due to Edmonds and Johnson [35].

Proposition II.4. For any graph \(G = (V, E)\), cost function \(c : E \to \mathbb{R}_{+}\), and a set \(O \subseteq V\) with an even number of vertices, the minimum weight of an \(O\)-join equals the optimum value of the following integral linear program.
\[
\begin{align*}
\text{min} & \quad c(y) \\
\text{s.t.} & \quad y(\delta(S)) \geq 1 \quad \forall S \subseteq V, |S \cap O| \text{ odd} \quad (3) \\
& \quad y_e \geq 0 \quad \forall e \in E
\end{align*}
\]

Definition II.5 \((\text{Satisfied cuts})\). For a set \(S \subseteq V\) such that \(u_0, v_0 \notin S\) and a spanning tree \(T \subseteq E\) we say a vector \(y : E \to \mathbb{R}_{\geq 0}\) satisfies \(S\) if one of the following holds:
\[
\begin{itemize}
\item \(\delta(S)_T\) is even, or
\item \(y(\delta(S)) \geq 1\).
\end{itemize}
\]

To analyze this class of algorithms, the main challenge is to construct a (random) vector \(y\) that satisfies all cuts (with probability 1) and for which \(\mathbb{E}[c(y)] \leq (1/2 - \varepsilon)c(x)\).

D. Near Min Cuts

Definition II.6. For \(G = (V, E, x)\), we say a cut \(S \subseteq V\) is an \(\eta\)-near min cut if \(x(\delta(S)) < 2 + \eta\).\(^4\)

We omit the proofs of the following simple lemmas in this version:

\(^2\)We put this notation in a box because it is so important and ubiquitous in this paper.

\(^3\)The standard name for this is the \(T\)-join polyhedron. Because we reserve \(T\) to represent our tree, we call this the \(O\)-join polyhedron, where \(\bar{O}\) represents the set of odd vertices in the tree.

\(^4\)Note this differs slightly from the notation in [28], [29] and Section I-B in which an \(\eta\) near min cut is said to be within a \(1 + \eta\) factor of the edge connectivity of the graph.
Lemma II.7. For $G = (V, E, x)$, let $A, B \subseteq V$ be two crossing $\epsilon_A, \epsilon_B$ near min cuts respectively. Then, $A \cap B, A \cup B, A \setminus B, B \setminus A$ are $\epsilon_A + \epsilon_B$ near min cuts.

Lemma II.8. If $A, B \subseteq V$ are disjoint and $C = A \cup B$ is an $\epsilon$ near min cut then $x(E(A, B)) \geq 1 - \epsilon / 2$.

The following lemma is proved in [36]:

Lemma II.9 ([36, Lem 5.3.5]). For $G = (V, E, x)$, let $A, B \subseteq V$ be two crossing $\epsilon$-near minimum cuts. Then, $x(E(A \cap B, A - B)), x(E(A \cap B, B - A)), x(E(\overline{A} \cup B, A - B))$ and $x(E(\overline{A} \cup B, B - A))$ are all at least $(1 - \epsilon / 2)$.

Lemma II.10. For $G = (V, E, x)$, let $A, B \subseteq V$ be two $\epsilon$ near min cuts such that $A \subseteq B$. Then
\[
x(\delta(A) \cap \delta(B)) = x(E(A, B)) \leq 1 + \epsilon, \text{ and}
\]
\[
x(E(A, B \setminus A)) \geq 1 - \epsilon / 2.
\]

E. Random spanning trees

The following simple lemmas appear in e.g. [1], thus we omit the proofs:

Lemma II.11. Let $G = (V, E, x)$, and let $\mu$ be any distribution over spanning trees with marginals $x$. For any $\epsilon$-near min cut $S \subseteq V$ (such that none of the endpoints of $e_0 = (u_0, v_0)$ are in $S$), we have $\mathbb{P}_{T \sim \mu} [S \text{ is a subtree of } T]$, or equivalently:
\[
= \mathbb{P}_{T \sim \mu} [\lvert T \cap E(S) \rvert = |S| - 1] \geq 1 - \epsilon / 2.
\]

Corollary II.12. Let $A, B \subseteq V$ be disjoint sets such that $A, B, A \cup B$ are $\epsilon_A, \epsilon_B, \epsilon_{A \cup B}$-near minimum cuts w.r.t., $x$ respectively, where none of them contain endpoints of $e_0$. Then for any distribution $\mu$ of spanning trees on $E$ with marginals $x$,
\[
\mathbb{P}_{T \sim \mu} [E(A, B)_T = 1] \geq 1 - (\epsilon_A + \epsilon_B + \epsilon_{A \cup B}) / 2.
\]

The following simple fact also holds by the union bound.

Fact II.13. Let $G = (V, E, x)$ and let $\mu$ be a distribution over spanning trees with marginals $x$. For any set $A \subseteq E$, we have
\[
\mathbb{P}_{T \sim \mu} [T \cap A = \emptyset] \geq 1 - x(A).
\]

III. Proof Overview

Algorithm 1 consists of two steps: sampling a tree whose marginals match $x$ (and hence has expected cost equal to $c(x)$), and then augmenting this with a minimum cost matching on the odd degree vertices of the tree. The goal of the current paper is to show that the expected cost of the minimum cost matching on the odd degree vertices of the sampled tree is at most $(1/2 - \epsilon)c(x)$. This is done by showing the existence of a cheap feasible O-join solution to (3).

First, note that if we only wanted to get an O-join solution of value at most $c(x)/2$, to satisfy all cuts, it is enough to set $y_c := 0.5x_e$ for each edge $\eta$ [17]. Now notice that if all of the near min cuts of $x$ containing $\epsilon$ are even, then we can reduce $y_c$ strictly below $0.5x_e$. The difficulty in implementing this approach comes from the fact that a high cost edge can be on many near min cuts and it may be exceedingly unlikely that all of these cuts will be even simultaneously. The idea in [1] is to initialize $y_c := 0.5x_e$ and then modify it by adding to it a random slack vector $s : E \to \mathbb{R}$: For each edge $e$, when certain special (few) $\eta$-near-min-cuts that $\epsilon$ is on are even in the tree, $s_e$ is set to $-x_e\beta$ where $\beta \approx \eta / 4$ chosen in the proof of Theorem II.1; for other cuts that contain $\epsilon$, whenever they are odd, the slack of other edges on that cut is increased to satisfy them (i.e., maintain feasibility of $y$ for that cut). The bulk of the effort was to show that this can be done while guaranteeing that $\mathbb{E}[s_e] < -c_xe$ for some $e > 0$, and therefore $\mathbb{E}[y_c] = 0.5x_e + \mathbb{E}[s_e] < (0.5 - \epsilon)x_e$.

To help the reader understand both the big picture as well as the ideas and contribution of the current paper, it is useful to first review in a bit more detail the approach taken in [1]. Let $N_\eta$ be the set of all $\eta$-near min cuts of $x$. A key idea there was to partition $N_\eta$ into three types: a set of near min cuts $\mathcal{H}$ that form a hierarchy (which is a laminar family of cuts), a set of cuts $N_{\eta,1}$ that are "crossed on one side" and a set of cuts $N_{\eta,2}$ that are "crossed on both sides". [1] showed that if we only need to satisfy the O-join constraints coming from $\mathcal{H}$, then we can find such a vector $s$.

However, this vector $s$ (which is negative in expectation) might "break" O-join constraints on cuts that are not in the hierarchy (i.e., cuts in $N_{\eta,1}$ and $N_{\eta,2}$). To resolve this, [1] showed how a negligible increase in the slack of certain edges

\footnote{This is because $x$ satisfies $x(\delta(S)) \geq 2$ for all $S$, whereas $y$ must satisfy $y(\delta(S)) \geq 1$ just for those cuts that have odd intersection with the tree $T$.}

\footnote{where the randomness comes from the random sampling of the tree}

\footnote{We will explain the terms in quotes shortly.}
(a slack component they called $s^*$) can be used to restore the feasibility of the O-join solution on all cuts, including those that are not in the hierarchy.

Concretely, because the cuts in $N_{\eta,2}$ have a rather complex structure, to simplify their handling, [1] changed the plan: Instead of starting with $y_e = 0.5x_e$, they started with $y_e = (x_e + \text{OPT}_e)/2$, where $\text{OPT}_e$ is an indicator for edge $e$ being in the optimal integral TSP solution. They then constructed slack vectors relative to the near min cuts of $(\text{OPT} + x)/2$. The advantage of doing so is that it guarantees that all near min cuts correspond to intervals of vertices along the optimal cycle, greatly simplifying the structure of the family of near min cuts under consideration. Slack on the edges in the optimal cycle was then used to handle the cuts in $N_{\eta,1}$ and $N_{\eta,2}$.

Unfortunately, this meant that the bound on the expected cost of the minimum cost matching from [1] is at most $(1/2 - \epsilon)(c(x) + c(\text{OPT}))/2$, which is insufficient to prove that the integrality gap of the LP is strictly below $3/2$.

In the present paper, we return to the plan of initializing $y_e := 0.5x_e$ and then construct a slack vector for each edge with the desired properties. Our starting point is the polygon decomposition $D$ of the $\eta$-near min cuts of $x$ [29].8 As stated previously, a polygon9 is a connected component of crossing $2 + \eta$ near minimum cuts, where two cuts are connected if they cross each other. It turns out that the way the polygon representation $D$ is constructed, each cut in $N_{\eta,2}$ is in exactly one polygon, and each edge on such a cut will have its slack increase in at most one polygon. Thus, cuts in $N_{\eta,2}$ can be handled independently for each polygon.

The main result of this paper is to show how to handle the cuts in $N_{\eta,2}$ (polygon by polygon) without resorting to the use of the OPT vector. Specifically, we prove the following:

**Theorem III.1** (Informal main theorem). For any connected component $C$ of $N_\eta$ (i.e. a polygon), let $C_2$ be the cuts in $C$ that are crossed on both sides. For any $\alpha > 0$, there is a vector $s^* : E \to \mathbb{R}$ depending on $T$ s.t.

(i) $\forall e \in E$, $s^*_e \geq 0$;

(ii) $\mathbb{E}[s^*_e] = O(\eta x_e)$, where the expectation is over the choice of tree $T$.

(iii) If $S \in C_2$ is a cut such that $\delta(S)$ is odd, then $s^*_e(\delta(S)) := \sum_{e \in \delta(S)} s^*_e \geq \alpha(1 - \eta)$.

Once cuts in $N_{\eta,2}$ are handled, the remaining cut structure becomes significantly simpler in that the polygons start to look very much like cycles: they contain only “outside atoms” and the fractional mass $x(a_i, a_{i+1})$ between adjacent atoms is $1 \pm \Theta(\eta)$ [1]. This enables us, with minor modification to the way in which cuts crossed on one side are handled to adapt one of the main results in [1].

**Theorem III.2** (Informal). Given a family $N_{\eta,\leq 1}$ of near-min cuts containing no cuts crossed on both sides, for any $\beta > 0$, there is a vector $s : E \to \mathbb{R}$ depending on $T$ such that

(i) $\forall e \in E$, $s_e \geq -\beta x_e$;

(ii) $\mathbb{E}[s_e] < -\epsilon x_e$ for some absolute constant $\epsilon > 0$, independent of $\eta$, where the expectations are over the choice of $T$.

(iii) If $S \in N_{\eta,\leq 1}$ is a cut such that $\delta(S)$ is odd, then $s_T(\delta(S)) = \sum_{e \in \delta(S)} s_e \geq 0$.

Note that this theorem crucially relies on the fact that the tree is sampled from a max-entropy distribution, whereas Theorem III.1 does not.

Before we explain the ideas underlying the proof of Theorem III.1, we quickly show how by setting

$$y_e(T) := 0.5x_e + s^*_e + s_e \quad \forall e,$$

these two theorems together imply the main result of this paper.

First, we show that $\mathbb{E}[c(y)] \leq c(x)(0.5 - \epsilon)$. To see this, observe that Theorem III.1(ii) together with Theorem III.2(ii) imply that for every edge $e \in E$,

$$\mathbb{E}[y_e] = 0.5x_e + \mathbb{E}[s_e] + \mathbb{E}[s^*_e] \leq x_e (0.5 + O(\eta a) - \epsilon\beta) \leq x_e (0.5 - \eta \epsilon'),$$

for $\alpha, \beta$ as chosen below and $\eta$ sufficiently smaller than $\epsilon$. Summing over all edges, this gives

$$\mathbb{E}[c(y)] \leq \left(\frac{1}{2} - \epsilon'\right)c(x).$$

Note that since $s^*_e$ is always nonnegative, it does not help us in our quest to reduce $y_e$ strictly below $0.5x_e$. That reduction comes only from $s_e$ being negative. Indeed, the raison d’être of the slack vector $s^*$ is to repair the feasibility of cuts which are odd in the tree but which have $s_e$ negative on some edges in
δ(S). This is why it is crucial that \( \mathbb{E}[s_e] \) is much smaller than \( -\mathbb{E}[s_e^2] \).

Next, we show that \( y(T) \) is feasible for every tree \( T \). For this, we need to consider three types of cuts:

a) Case 1:: \( \delta(S)_T \) is odd and \( x(\delta(S)) > 2 + \eta \).

Since \( s_T^+(\delta(S)) \geq 0 \) and \( s_T(\delta(S)) \geq -\beta x(\delta(S)) \), we have

\[
y_T(\delta(S)) = 0.5x(\delta(S)) + s_T^+(\delta(S)) + s_T(\delta(S)) \\
\geq (0.5 - \beta)x(\delta(S)) \geq (0.5 - \beta)(2 + \eta) \geq 1,
\]

for \( \beta \approx \eta/2 \).

b) Case 2:: \( \delta(S)_T \) is odd, \( S \in \mathbb{N}_{\eta/2} \). In this case \( s_T(\delta(S)), s_T^+(\delta(S)) \geq 0 \) so \( y_T(\delta(S)) \geq 0.5x(\delta(S)) \geq 1 \).

c) Case 3:: \( \delta(S)_T \) is odd, \( x(\delta(S)) \leq 2 + \eta \), and \( S \in \mathbb{N}_{\eta/2} \). In this case, \( s_T^+(\delta(S)) \geq \alpha(1 - \eta) \) and \( s_T(\delta(S)) \geq -\beta x(\delta(S)) \), so we have

\[
y_T(\delta(S)) = 0.5x(\delta(S)) + s_T^+(\delta(S)) + s_T(\delta(S)) \\
\geq (0.5 - \beta)x(\delta(S)) + \alpha(1 - \eta) \geq 1,
\]

for \( \alpha \approx 2\beta \) using \( x(\delta(S)) \geq 2 \).

A. Overview of proof of Theorem III.1 – no inside atoms

Given a connected component \( C \) of cuts in \( \mathbb{N}_{\eta/2} \), we can partition vertices of \( G \) into sets \( a_0, \ldots, a_{m-1} \) (called atoms); this is the coarsest partition such that for each \( a_i \), and each \( (S, S) \subseteq C \), we have \( a_i \subseteq S \) or \( a_i \subseteq \overline{S} \). One of these atoms, \( a_0 \) is the atom that contains \( u_0, v_0 \). We call \( a_0 \) the root. In the following, we will often identify an atom with the set of vertices that it represents.10

If \( \eta = 0 \), then [23]) shows that the structure of cuts in \( C \) can be represented by a cycle; namely we can arrange these atoms around a cycle such that, perhaps after renaming, for any \( 0 \leq i \leq m - 1 \),

\[
x(E(a_i, a_{i+1} \text{mod } m)) = 1 \text{ and cuts of } C \text{ are just the mincuts of this cycle.}
\]

As mentioned, [28], [29] studied the case when \( 0 < \eta \leq 2/5 \) and introduced the notion of polygon representation, in which case atoms can be placed on the sides of an equilateral polygon \( P \) and some atoms placed inside the polygon, such that every cut in \( C \) can be represented by a diagonal of this polygon.

In the rest of this section, we fix \( C \) and we outline the ideas behind the proof of Theorem III.1 in the special case that the polygon \( P \) representing the connected component of cuts \( C \) contains no inside atoms. This latter assumption simplifies the argument but still illustrates many of the main ideas.

We assume that the atoms of \( P \) are labelled counterclockwise from \( a_0 \) to \( a_{m-1} \). We associate to each diagonal (defining a cut) the side which does

---

10For example, it will be convenient to write cuts as subsets of atoms. In this case the cut is the union of the vertices in those atoms.
Since $E$ also crosses $S_r$, which contains the fewest number of atoms in $S$ (green atoms), is the same as $L(p)$. The edges in $E^-(L(p))$ are those that go between green atoms and brown atoms. Note also that any edge in $\delta(S)$ with one endpoint to the right of $p$ that is not in $E^+(L(p))$ is in $E^-(L(p))$. (It can't be in $E^+(L(p))$ since those edges have one endpoint to the left of $l$.) Note also that since the green + yellow region as well as the brown region are each the difference of two crossing $\eta$ near min-cut, each is a $2\eta$ near min cut. So by Lemma II.8, the fraction of edges with one endpoint in each of these regions is $1 - O(\eta)$. (To extend this to the case where polygon $P$ may have inside atoms, in the full version we show that there are no atoms in the yellow region.)

Fig. 3: Since $S$ is crossed on the right and any cut that crosses $S$ on the right also crosses $L(p)$ on the right, the cut $S_r$, which contains the fewest number of atoms in $S$ (green atoms), is the same as $L(p)$. The edges in $E^-(L(p)) = E^+(S)$ are those that go between green atoms and brown atoms. Note also that any edge in $\delta(S)$ with one endpoint to the right of $p$ that is not in $E^+(L(p))$ is in $E^-(L(p))$. (It can't be in $E^+(L(p))$ since those edges have one endpoint to the left of $l$.) Note also that since the green + yellow region as well as the brown region are each the difference of two crossing $\eta$ near min-cut, each is a $2\eta$ near min cut. So by Lemma II.8, the fraction of edges with one endpoint in each of these regions is $1 - O(\eta)$. (To extend this to the case where polygon $P$ may have inside atoms, in the full version we show that there are no atoms in the yellow region.)

Fig. 4: Edge $e = \{a_i, a_k\}$ is in $E^-(L(p_i))$ for all $i$.

The reason we do this is that it is crucial for subsequent arguments to be able to condition on near min cuts being trees using Lemma II.11, i.e., that for $S$ a near min cut, $E(S) \cap T$ is very likely to be a tree. However, this lemma can only be used on sets which do not contain $a_0, a_0$.

Thus, we will refer to a cut by the set of outside atoms it contains, say $[a_i, a_j], i < j$. (This denotes the side of the diagonal containing the atoms $a_i, a_{i+1}, \ldots, a_j$.) We equivalently refer to this cut by giving the left and right polygon points of its diagonal $[p_{i-1}, p_i]$.

As mentioned above, the reason for the slack vector $s^*$ that we construct here is to restore the feasibility of cuts $S$ in $C_2$ which are odd in the tree but which have $s_e$ negative on some edges in $\delta(S)$. The high level approach in the proof is the following. Initialize $s^*_e := 0$ for all $e$. Now define a set of bad events whose occurrence signifies that some of these near min cuts are potentially in need of such a repair. These bad events should satisfy the follow desiderata:

(a) Each bad event occurs with probability $O(\eta)$, where the probability is taken over the choice of tree $T$.

(b) The occurrence of a bad event $B$ in a tree $T$ triggers a slack increase on an associated set of edges $E(B)$. Specifically, when $B$ occurs, each edge $e \in E(B)$ has its slack $s^*_e$ increased by $\alpha x_e$.

(c) Each edge $e$ is in $E(B)$ for a constant number of bad events $B$. Combining (a) and (b), this implies that $\mathbb{E}[s^*_e] = O(\eta \alpha x_e)$ (condition 2 of...
Theorem III.1).

(d) Each $\eta$-near-min cut $S$ is associated with a constant number of bad events $B(S)$, such that when $\delta(S)_T$ is odd, at least one of the bad events $B \in B(S)$ occurs. We will ensure that the edges in $E(B)$ (on which slack increases are triggered) are a subset of $\delta(S)$ of fractional value at least $\Omega(1)$. Therefore, if $S$ is odd in the tree, $s^* \eta (\delta(S)) \geq ax(E(B)) \geq \Omega(ax)$ implying condition (iii) of Theorem III.1 (once the constant are chosen appropriately).

B. Satisfying the above desiderata

Consider any near min cut $S$ in $P$, which is crossed on the left and on the right. Let $S_L$ and $S_R$ be the cuts crossing $S$ on the left and right with minimum sized intersection with $S$. See Figure 1.

One of the very nice things about cuts crossed on both sides is the following:

Claim III.3. For any near min cut $S \in \mathcal{C}_2$, $\mathbb{P} [\delta(S)_T = 2] \geq 1 - O(\eta)$.

Proof sketch. To see this, for a set $S$ crossed on both sides, let

\[
E^\leftarrow(S) := E(S \cap S_L, S_L \setminus S) \\
E^\to(S) := E(S \cap S_R, S_R \setminus S) \\
E^0(S) := \delta(S) \setminus E^\leftarrow(S) \setminus E^\to(S)
\]

and consider the bad events

\[
B^\leftarrow(S) := \mathbb{1}\{E^\leftarrow(S)_T \neq 1\} \\
B^\to(S) := \mathbb{1}\{E^\to(S)_T \neq 1\} \\
B^0(S) := \mathbb{1}\{E^0(S)_T \neq 0\}.
\]

See Figure 1.

Clearly if none of these bad events occur, then $S$ is even in the tree (i.e., $\delta(S)_T = 2$). Now, note that $S_L, S_R, S_T$ are all $\eta$-near min cuts and so by Lemma II.9 and Corollary II.12, we have

\[
\lambda(S^\leftarrow(S)) \geq 1 - \frac{\eta}{2}, \lambda(S^\to(S)) \geq 1 - \frac{\eta}{2}, \lambda(S^0(S)) = \lambda(\delta(S) \setminus E^\leftarrow(S) \setminus E^\to(S)) = O(\eta)
\]

and $\mathbb{P}[B^\leftarrow(S)], \mathbb{P}[B^\to(S)], \mathbb{P}[B^0(S)] = O(\eta)$. □

The next step in our plan is to decide what slack increases are triggered by these bad events. The first thing one might think of is to have the above bad events (4) trigger a slack increase on $E^\leftarrow(S) \cup E^\to(S)$. Namely, for each set $S$ crossed on both sides, $\forall e \in E^\leftarrow(S) \cup E^\to(S)$ we set:

\[
s^e_\eta := a x_e \cdot \mathbb{1}\{B^\leftarrow(S), B^\to(S) \text{ or } B^0(S) \text{ occurs}\}.
\]

In addition to desiderata (a) and (b), this approach satisfies (d) since $\lambda(S^\leftarrow(S) \cup E^\to(S)) \geq 2 - \eta$.

Unfortunately though, this does not satisfy desiderata (c), since if $e \in E(\delta, \eta)$, it could be that $e \in (E^\leftarrow(S) \cup E^\to(S)$ for many near min cuts $S$ in which case $E[s^e_\eta]$ is

\[
ax_e \cdot \mathbb{P}[\exists S \text{ odd in } T \text{ s.t. } e \in (E^\leftarrow(S) \cup E^\to(S))].
\]

This could be way too large (say, around $ax_e$).

So, rather than defining a bad event for every cut $S$ crossed on both sides individually (i.e., up to $O(n^2)$ events), we instead define a constant number of bad events for each polygon point $p$, hence at most $O(m)$ events.

1) Defining bad events for each polygon point: For a fixed polygon point $p$, let $L := L(p)$ be the set crossed on both sides that extends farthest clockwise from $p$ and as above, let $L_R$ be the cut that crosses it on the right with the minimum number of outside atoms in the intersection. Analogously define $R := R(p)$ and $R_L$. See Figure 2.
Now we consider two bad events:
\[
B^+(p) = 1\{E^+(L(p))_T \neq 1 \text{ or } E^0(L(p))_T \neq 0\}, \\
B^-(p) = 1\{E^-(R(p))_T \neq 1 \text{ or } E^0(R(p))_T \neq 0\}.
\] (5)

For these events, we have the following two claims:

**Claim III.4.** For any near min cut \( S = [p, q] \), \( E^-(L(q)) = E^-(S) \) and \( E^+(R(p)) = E^-(S) \). Moreover, \( E^-(S) \subseteq E^0(L(p)) \cup E^0(R(p)) \). See Figure 3. Therefore, if neither \( B^+(q) \) or \( B^-(p) \) occur, then \( \delta(S)_T \) is even.

In addition we have

**Claim III.5.** For any polygon point \( p \), \( P[B^+(p)] \), \( P[B^-(p)] = O(\eta) \).

This follows arguments similar to those used in Claim III.3, using that \( x(E^-(L(p))) \geq 1 - \eta/2 \), and \( x(E^0(L(p))) = O(\eta) \) (and similarly for \( R(p) \)).

These bad events satisfy the desiderata (a) and (d) (assuming we define \( E(B) \) such that \( x(E(B)) \in O(1) \)).

2) Defining the slack increase sets for bad events: It remains to determine the sets \( E(B^+(p)), E(B^-(p)) \) for which slack increases are triggered when the bad events occur. In particular, we will let \( E(B^+(p)) \subseteq E^+(L(p)) \) and \( E(B^-(p)) \subseteq E^-(R(p)) \) such that:

\* \( x(E(B^+(p))) \geq \Omega(1) \) and \( x(E(B^-(p))) \geq \Omega(1) \) (to guarantee (d)),

\*\* All edges \( e \) are in at most a constant number of sets \( E(B) \) (to guarantee (c)).

Assuming we can satisfy \( \ast \) and \( \ast\ast \), we can set \( s^*_e = ax_e \) for all \( e \in E(B) \) when \( B \) occurs to satisfy all four desiderata.

a) First try: The most natural choice is to simply let \( E(B^+(p)) = E^+(L(p)) \). Here, \( \ast \) obviously holds but unfortunately \( \ast\ast \) fails. Indeed, there are examples (see Figure 4) for which there exist edges \( e \in E(a_i, a_j) \) with \( |j - i| = \Omega(m) \) that belong to \( E^+(L(p_k)) \) for \( \Omega(m) \) many values of \( i \leq k \leq j \).

b) Second try: Let \( a_i \) be the atom immediately to the left of \( p \) and \( a_{i+1} \) the atom immediately to the right of \( p \) (i.e. \( p = p_i \)). Note that all edges with one endpoint in \( a_i \) and one in \( a_{i+1} \) are in \( E^+(L(p)) \).

Now if it was always the case that \( x(E(a_i, a_{i+1})) \geq \gamma \) for some universal constant \( \gamma > 0 \), then, when one of these bad events occurs, say \( B^+(p) \), we could simply increase the slack of every edge \( e \) in \( E(a_i, a_{i+1}) \) by \( ax_e \). This approach is analogous to the method employed in [1] where slack was increased on OPT edges. One might have some hope that this is true since it holds with \( \gamma = 1 \) for the cactus representation of min cuts (i.e. when \( \eta = 0 \)).

Unfortunately, as observed in [6] there is a family of near minimum cuts such that the polygon representation has no inside atoms, yet \( E(a_i, a_{i+1}) = \emptyset \) for some consecutive pairs of (outside) atoms (see Figure 5) (even though there are cuts whose diagonals end between those atoms). So, this method is doomed even if inside atoms are not present.

**c) Our method:** The first try works if there are no “long” edges. So, to rectify that attempt we essentially “ignore” long edges (edges between distant atoms) in our charging argument and argue that they only contribute minimally to \( E^-(L(p)) \) and \( E^+(R(p)) \).

To this end, define \( L(p)^{\cap R} := L(p) \cap L(p)^{\cap R} \), and let \( L^+(p) \) be the cut crossing \( L(p)^{\cap R} \) on the left that maximizes the number of outside atoms in the intersection of \( L^+(p) \) and \( L(p)^{\cap R} \) (and similarly \( R^+(p) \) to maximize the intersection with \( R(p)^{\cap L} \) on the right). If \( L^+(p) \) does not exist, i.e. no cut crosses \( L(p)^{\cap R} \) on the left, set \( L^+(p) = \emptyset \), and similarly for \( R^+(p) \). See Figure 6. We let:

\[
E(B^-(p)) := E(L(p)^{\cap R} \cup L^+(p), L(p) \cap L^+(p)^{\cap R}) \\
E(B^+(p)) := E(R(p)^{\cap L} \cup R^+(p), R(p) \cap R^+(p)^{\cap L}).
\] (6)

The following claim establishes \( \ast \) for Eq. (6). It can be proved using methods similar to Claim III.3; see Figure 3.

**Claim III.6.** For all polygon points \( p \), \( x(E(B^+(p))) \geq 1 - O(\eta) \).

And finally, the following claim establishes \( \ast\ast \):

**Claim III.7.** For any edge \( e \), we have \( e \in E(B^+(p)) \) for at most one polygon point \( p \) and similarly \( e \in E(B^-(q)) \) for at most one polygon point \( q \).

The proof of Claim III.7 is more involved, thus we defer its discussion to the full version. We have also until now assumed there are no inside atoms; to deal with this possibility we need to prove new structural properties of polygons with inside atoms. These are presented in the full version of the paper.

**References**

[1] A. R. Karlin, N. Klein, and S. Oveis Gharan, “A (slightly) improved approximation algorithm for metric tsp,” in STOC. ACM, 2021.
[2] D. L. Applegate, R. E. Bixby, V. Chvatal, and W. J. Cook, The Traveling Salesman Problem: A Computational Study (Princeton Series in Applied Mathematics). Princeton, NJ, USA: Princeton University Press, 2007.

[3] M. Karpinski, M. Lampis, and R. Schmied, “New inapproximability bounds for tsp,” Journal of Computer and System Sciences, vol. 81, no. 8, pp. 1665 – 1677, 2015.

[4] N. Christofides, “Worst case analysis of a new heuristic for the traveling salesman problem,” Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA, Report 388, 1976.

[5] A. I. Serdyukov, “O nekotorykh ekstremal’nykh obkhodakh v grafakh,” Upravlyayushchaya sistemy, vol. 17, pp. 76–79, 1978. [Online]. Available: http://rusl.math.nsc.ru/aim/journals/us/us17/us17_007.pdf

[6] S. Oveis Gharan, A. Saberi, and M. Singh, “A randomized rounding approach to the traveling salesman problem,” in FOCS. IEEE Computer Society, 2011, pp. 550–559.

[7] T. Moemke and O. Svensson, “Approximating graphic tsp by matchings,” in FOCS, 2011, pp. 560–569.

[8] M. Mucha, “1.33-approximation for graphic tsp,” in STACS, 2012, pp. 30–41.

[9] A. Sebő and J. Vygen, “Shorter tours by nicer ears,” 2012, coRR abs/1201.1870.

[10] A. Haddadan, A. Newman, and R. Ravi, “Shorter tours and longer detours: uniform covers and a bit beyond,” Math. Program., vol. 185, no. 1-2, pp. 245–273, 2021. [Online]. Available: https://doi.org/10.1007/s10107-019-01426-8

[11] A. R. Karlin, N. Klein, and S. Oveis Gharan, “An improved approximation algorithm for tsp in the half integral case,” in STOC, K. Makarychev, Y. Makarychev, M. Tulsiani, G. Kamath, and J. Chuzhoy, Eds. ACM, 2020, pp. 28–39.

[12] A. Haddadan and A. Newman, “Towards improving christofides algorithm for half-integer tsp,” in ESA, ser. LIPIcs, M. A. Bender, O. Svensson, and G. Herman, Eds., vol. 144. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, pp. 56:1–56:12.

[13] A. Gupta, E. Lee, J. Li, M. Mucha, H. Newman, and S. Sarkar, “Matroid-based tsp rounding for half-integral solutions,” CoRR, vol. abs/2111.09290, 2021. [Online]. Available: https://arxiv.org/abs/2111.09290

[14] G. Dantzig, D. Fulkerson, and S. Johnson, “On a linear programming combinatorial approach to the traveling salesman problem,” OR, vol. 7, pp. 58–66, 1959.

[15] M. Held and R. Karp, “The traveling salesman problem and minimum spanning trees,” Operations Research, vol. 18, pp. 1138–1162, 1970.

[16] M. Goemans and D. Bertsimas, “Survivable network, linear programming relaxations and the parsimonious property,” Math Program, vol. 60, 06 1993.

[17] L. A. Wolsey, “Heuristic analysis, linear programming and branch and bound,” in Combinatorial Optimization II, ser. Mathematical Programming Studies. Springer Berlin Heidelberg, 1980, vol. 13, pp. 121–134.

[18] V. Traub, J. Vygen, and R. Zenklusen, “Reducing path tsp to tsp,” in STOC, K. Makarychev, Y. Makarychev, M. Tulsiani, G. Kamath, and J. Chuzhoy, Eds. ACM, 2020, pp. 14–27.

[19] R. Carr and P. Rego, “A new bound for the 2-edge connected subgraph problem,” in IPCO, 1998, pp. 112–125.

[20] S. Boyd, Y. Fu, and Y. Sun, “A 5/4-approximation for subcubic 2ec using circulations and obliged edges,” Discrete Applied Mathematics, vol. 209, pp. 48–58, 2016.

[21] A. Sebő and J. Vygen, “Shorter tours by nicer ears: 7/5-approximation for the graph-tsp, 5/2 for the path version, and 4/3 for two-edge-connected subgraphs,” Combinatorica, vol. 34, no. 5, pp. 597–629, 2014.

[22] S. Boyd, J. Cheriyan, R. Cummings, L. Grout, S. Ibrahimpur, Z. Szegedy, and L. Wang, “A 4/3-Approximation Algorithm for the Minimum 2-Edge Connected Multisubgraph Problem in the Half-Integral Case,” in APPROX/RANDOM, J. Byrka and R. Meka, Eds., vol. 176. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020, pp. 61:1–61:12.

[23] E. Dinitis, A. Karzanov, and M. Lomonosov, “On the structure of a family of minimal weighted cuts in graphs,” Studies in Discrete Mathematics (in Russian), ed. A.A. Fridman, 290-306, Nauka (Moscow), 1976.

[24] D. R. Karger, “Minimum cuts in near-linear time,” J. ACM, vol. 47, no. 1, pp. 46–76, Jan. 2000.

[25] A Near-Linear Time Algorithm for Constructing a Cactus Representation of Minimum Cuts, 2009.

[26] J. Byrka, F. Grandoni, and A. J. Ameli, “Breaching the 2-approximation barrier for connectivity augmentation: A reduction to steiner tree,” in STOC. Association for Computing Machinery, 2020, p. 815–825.

[27] F. Cecchetto, V. Traub, and R. Zenklusen, Bridging the Gap between Tree and Connectivity Augmentation: Unified and Stronger Approaches. New York, NY, USA: Association for Computing Machinery, 2021, p. 370–383. [Online]. Available: https://doi.org/10.1145/3406325.3451086

[28] A. A. Benczúr, “A representation of cuts within 6/5 times the edge connectivity with applications,” in FOCS, 1999, pp. 92–102.

[29] A. A. Benczúr and M. X. Goemans, “Deformable polygon representation and near-mincuts,” Building Bridges: Between Mathematics and Computer Science, M. Groetschel and G.O.H. Katona, Eds., Bolyai Society Mathematical Studies, vol. 19, pp. 103–135, 2008.

[30] A. R. Karlin, N. Klein, S. O. Gharan, and X. Zhang, “An improved approximation algorithm for the minimum k-edge connected multi-subgraph problem,” 2021. [Online]. Available: https://arxiv.org/abs/2101.05921

[31] L.-C. Lau, R. Ravi, and M. Singh, Iterative Methods in Combinatorial Optimization, 1st ed. New Y ork, NY, USA: Cambridge University Press, 2011.

[32] K. Jain, “A factor 2 approximation algorithm for the generalized steiner network problem,” Combinatorica, vol. 21, pp. 39–60, 2001.

[33] A. Asadpour, M. X. Goemans, A. Madry, S. Oveis Gharan, and A. Saberi, “An o(log n / log log n)-approximation algorithm for the asymmetric traveling salesman problem,” in SODA, 2010, pp. 379–389.

[34] J. Edmonds, “Submodular functions, matroids and certain polyhedra,” in Combinatorial Structures and Their Applications. New York, NY, USA: Gordon and Breach, 1970, pp. 69–87.

[35] J. Edmonds and E. L. Johnson, “Matching, euler tours and the chinese postman,” Mathematical Programming, vol. 5, no. 1, pp. 88–124, 1973.

[36] A. A. Benczúr, “Cut structures and randomized algorithms in edge-connectivity problems,” Ph.D. dissertation, MIT, 1997.