2018 IEEE 59th Annual Symposium on Foundations of Computer Science

Beating the integrality ratio for s-t-tours in graphs

Vera Traub and Jens Vygen
Research Institute for Discrete Mathematics, University of Bonn
{traub,vygen}@or.uni-bonn.de

***Abstract*—Among various variants of the traveling salesman problem, the s-t-path graph TSP has the special feature that we know the exact integrality ratio, 3/2, and an approximation algorithm matching this ratio. In this paper, we go below this threshold: we devise a polynomial-time algorithm for the s-t-path graph TSP with approximation ratio 1.497. Our algorithm can be viewed as a refinement of the 3/2-approximation algorithm by Sebő and Vygen [16], but we introduce several completely new techniques. These include a new type of ear-decomposition, an enhanced ear induction that reveals a novel connection to matroid union, a stronger lower bound, and a reduction of general instances to instances in which s and t have small distance (which works for general metrics).**

***Index Terms*—traveling salesman problem; approximation algorithms; integrality ratio; ear-decomposition**

I. Introduction

Since 2010, there have been many interesting results on approximation algorithms for variants of the traveling salesman problem. This includes in particular better approximation algorithms for the graph TSP [16], the asymmetric TSP [18], and the s-t-path TSP [21].

Almost all the algorithms work with the classical linear programming relaxations, with an LP solution as starting point of the algorithm or at least for the analysis. Although these LPs have been studied intensively for decades, both for the symmetric and the asymmetric TSP, we still do not know their integrality ratios. For the (symmetric) s-t-path TSP we are quite close: the integrality ratio is between 1.5 and 1.53 [17], and in the s-t-path graph TSP (a well-studied special case), it is indeed $\frac{3}{2}$, and an approximation guarantee matching the integrality ratio is known [16].

In this paper, we go below this threshold: we show that the s-t-path graph TSP has a 1.497-approximation algorithm.

The s-t-path graph TSP is defined as follows. Given a connected undirected graph G and two vertices s and t, find a shortest tour from s to t that visits all vertices. Such a tour can be described as a sequence $v_0, v_1, \ldots, v_k$ of vertices such that $v_0 = s$, $v_k = t$, every vertex appears at least once, and $\{v_{i-1}, v_i\} \in E(G)$ for all i (and we minimize k). Equivalently (via Euler's theorem), it can be described as a subset F of $2E(G)$ (the multi-set that contains two copies of every edge) such that $(V(G), F)$ is a connected multi-graph in which all vertices have even degree except s and t (which have odd degree unless $s = t$). Such a set F is called an s-t-*tour*; the goal is to minimize $|F|$.

The s-t-path graph TSP is a special case of the general s-t-path TSP (in which the edges have arbitrary nonnegative costs). It has been studied mainly because the well-known examples that yield the lower bounds for the integrality ratio ($\frac{4}{3}$ for $s = t$ and $\frac{3}{2}$ for $s \neq t$) are in fact graph instances. Moreover, some ideas developed for the graph case triggered progress on general weights later on; an example is Gao's [8] new proof of the integrality ratio $\frac{3}{2}$ for the s-t-path graph TSP and its use by [9] for the general s-t-path TSP.

For $s \neq t$, Christofides' algorithm yields only a $\frac{5}{3}$-approximation, even for the s-t-path graph TSP, as Hoogeveen [11] showed. Within nine months in 2011–2012, the approximation ratio for the s-t-path graph TSP was improved four times: first to

2575-8454/18/$31.00 ©2018 IEEE
DOI 10.1109/FOCS.2018.00078

1.586 by Mömke and Svensson [13], then to 1.584 by Mucha [14], then to 1.578 by An, Kleinberg and Shmoys [1], and finally to 1.5 by Sebő and Vygen [16]. This approximation ratio $\frac{3}{2}$ matches the integrality ratio lower bound. However, in this paper, we improve on this.

We present a polynomial-time algorithm that guarantees to find, for every instance of the s-t-path graph TSP, an s-t-tour with at most $1.497\,\text{OPT}$ edges, where OPT is the minimum number of edges of such a tour.

A. Preliminaries on T-tours

Given a vertex set V and a set $T \subseteq V$ of even cardinality, a T*-join* is a (multi)set J of edges such that T is exactly the set of odd-degree vertices in (V, J). A T*-tour* in a graph G is a T-join $J \subseteq 2E(G)$ such that $(V(G), J)$ is connected. We allow taking two copies of an edge, but more than two are never useful. Henceforth we speak of edge sets and graphs even if they contain parallel edges. It is obvious that a T-tour exists if and only if G is connected and $|T|$ is even. The s-t-path graph TSP asks for an $(\{s\}\triangle\{t\})$-tour of minimum cardinality in a given connected graph G. Instead of $(\{s\}\triangle\{t\})$-tour we will use the shorter name s-t-*tour*. The s-t-path graph TSP is NP-hard, and unless P=NP there is no polynomial-time algorithm with better approximation ratio than $\frac{685}{684}$ [12].

The more general T-tour problem (in graphs or in general) has been introduced by Sebő and Vygen [16]; they gave a $\frac{3}{2}$-approximation algorithm for finding a smallest T-tour in a graph, and a $\frac{7}{5}$-approximation algorithm if $T = \emptyset$. Sebő [15] showed an approximation ratio of $\frac{8}{5}$ for finding a minimum-weight T-tour in a weighted graph. These approximation ratios have not been improved since then. Section II of our paper works for general T-tours, but later parts do not seem to extend beyond constant $|T|$.

For a graph G and a set $T \subseteq V(G)$ with $|T|$ even and a set $W \subseteq V(G)$, let $(G, T)/W$ be the instance of the T-tour problem arising by *contraction* of W. More precisely, we define $(G, T)/W$ to be the instance of the T-tour problem where we are looking for a T'-tour in the graph G/W and T' contains all elements of $T \setminus W$ and contains in addition the vertex arising from the contraction if $|T \cap W|$ is odd.

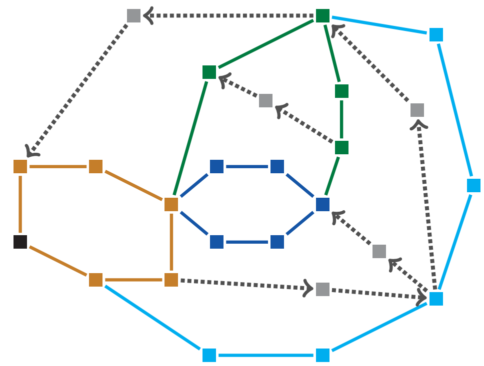

Fig. 1. An example of a well-oriented ear-decomposition. The oriented short ears are shown in gray and dotted.

Without loss of generality one may assume that the input graph is 2-vertex-connected because if $G = G[W_1] \cup G[W_2]$, where W_1 and W_2 share a single vertex, then it suffices to solve the instances $(G, T)/W_1$ and $(G, T)/W_2$ (cf. [16]).

We denote $n := |V(G)|$ throughout this paper. As an obvious lower bound, note that every T-tour has at least $n-1$ edges (for any T). Another (often better) lower bound is given by the classical linear program, which, for $T = \{s, t\}$, is

$$\begin{aligned}\min\{\ & x(E(G)) : \\ & x(\delta(U)) \geq 2 \ (\emptyset \subset U \subseteq V(G) \setminus \{s,t\}), \\ & x(\delta(U)) \geq 1 \ (\{s\} \subseteq U \subseteq V(G) \setminus \{t\}), \\ & x_e \geq 0 \ (e \in E(G))\},\end{aligned}$$

where $\delta(U)$ denotes the set of edges with exactly one endpoint in U, and $x(F) := \sum_{e\in F} x_e$ for $x \in \mathbb{R}^{E(G)}$ and $F \subseteq E(G)$. Let LP denote the value of this linear program. The *integrality ratio* is the supremum of $\frac{\text{OPT}}{\text{LP}}$ over all instances. If G is a circuit and s and t have distance $\frac{n}{2}$ in G, then the value of this LP is n, but every s-t-tour has at least $\frac{3}{2}n - 2$ edges. This classical example shows that the integrality ratio is at least $\frac{3}{2}$.

B. Preliminaries on ear-decompositions

For a finite sequence $P_0, P_1, \ldots, P_l$ of graphs, let $V_i = V(P_0) \cup V(P_1) \cup \cdots \cup V(P_i)$ and $G_i = (V_i, E(P_1) \cup \cdots \cup E(P_i))$. If P_0 has a single vertex, and each P_i is either a circuit with $|V(P_i) \cap V_{i-1}| = 1$ or a path such that exactly its endpoints belong to V_{i-1} ($i = 1, \ldots, l$), then $P_0, P_1, \ldots, P_l$ is called an *ear-decomposition* of G_l.

The graphs $P_1, \ldots, P_l$ are called *ears*. The vertices of $V(P_i) \cap V_{i-1}$ are called *endpoints* of P_i, the other vertices of P_i are its *internal vertices*. We denote the set of internal vertices of P_i by $\text{in}(P_i)$. We always have $|\text{in}(P_i)| = |E(P_i)| - 1$. An ear is called *open* if it is P_1 or it is a path. Other ears are called *closed*. If all ears are open, the ear-decomposition is called *open*. An ear is called an r-*ear* if it has exactly r edges. We denote the number of r-ears of a fixed ear-decomposition by k_r. The 1-ears are also called *trivial* ears; they have no internal vertices. A *short* ear is a 2-ear or 3-ear. Ears with more than three edges are called *long*. An ear is *odd* if the number of its edges is odd, otherwise *even*.

Figure 1 shows an ear-decomposition with P_0 colored black, a 6-ear P_1 colored brown, a closed 6-ear P_2 colored blue, an open 5-ear P_3 colored green, an open 6-ear colored cyan, and five 2-ears (gray, dotted, ignore the orientation). Every vertex is an internal vertex of exactly one ear (except for the vertex of P_0) and is colored accordingly in this figure.

We say that an ear P is *attached to* an ear Q (at v) if v is an internal vertex of Q and an endpoint of P. A vertex is *pendant* if it is not an endpoint of any nontrivial ear, and an ear is *pendant* if it is nontrivial and all its internal vertices are pendant. Having a fixed vertex set T, we call an ear P *clean* if it is short and $|T \cap \text{in}(P)| = \emptyset$, i.e. none of its internal vertices is contained in T.

C. Simple ear induction

The following lemma tells how to construct a T-tour by considering the nontrivial ears in reverse order. For any nontrivial ear P_i, note that G_i/V_{i-1} is a circuit with $|E(G_i/V_{i-1})| = |E(P_i)|$.

Lemma 1 ([16]): Let P be a circuit and $T_P \subseteq V(P)$ with $|T_P|$ even. Then there exists a T_P-join $F \subseteq 2E(P)$ such that the graph $(V(P), F)$ is connected and

$$|F| \;\leq\; \tfrac{3}{2}(|E(P)| - 1) - \tfrac{1}{2} + \gamma, \qquad (1)$$

where $\gamma = 1$ if $|E(P)| \leq 3$ and $T_P = \emptyset$, and $\gamma = 0$ otherwise.

Proof: If $T_P = \emptyset$, then $F = E(P)$ does the job because $|E(P)| - 1 \geq 3$ or $\gamma = 1$.

Let now $T_P \neq \emptyset$. The vertices of T_P subdivide P into subpaths. Color these paths alternatingly red and blue. Let E_R and E_B denote the set of edges of red and blue subpaths, respectively. Without loss of generality $|E_R| \leq |E_B|$. Then we take two copies of each edge in E_R and one copy of each edge in E_B. Note that $E_R \neq \emptyset$, and remove one pair of parallel edges. This yields $F \subseteq 2E(P)$ with

$$\begin{aligned} |F| \;&= |E_B| + 2|E_R| - 2 \;\leq\; \tfrac{3}{2}|E(P)| - 2 \\ &= \tfrac{3}{2}(|E(P)| - 1) - \tfrac{1}{2}. \end{aligned}$$

□

This can be used to construct a T-tour as follows. Let $P_1, \ldots, P_l$ be the nontrivial ears of an ear-decomposition of G (trivial ears can be deleted beforehand). Starting with $T_l := T$ and $F := \emptyset$, we do the following for $i = l, \ldots, 1$. Apply Lemma 1 to $(G_i, T_i)/V_{i-1}$ and obtain a set $F_i \subseteq 2E(P_i)$. Set $F := F \cup F_i$ and $T_{i-1} := T_i \triangle \{v \in V(P_i) : |\delta_{F_i}(v)| \text{ odd}\}$. Then the union of F_i and any T_{i-1}-tour in G_{i-1} is a T_i-tour in G_i. By induction, $F_1 \cup \cdots \cup F_l$ is a T-tour in G. Since $|F_i| \leq \frac{3}{2}(|E(P_i)| - 1) - \frac{1}{2} + \gamma_i = \frac{3}{2}|\text{in}(P_i)| - \frac{1}{2} + \gamma_i$, this T-tour has at most $\frac{3}{2}(n-1)$ edges if $\gamma_i = 0$ for at least half of the nontrivial ears, in particular if most ears are long.

D. Outline of the Sebő–Vygen algorithm

The previously best approximation algorithm for the graph s-t-path TSP, due to [16], is the basis of our work. Let us briefly review this algorithm before we explain how to improve on it. The previous section shows already why short ears (of length 2 and 3) need special attention.

The first step is to compute a *nice* ear-decomposition: one with minimum number of even ears, in which all short ears are pendant, and internal vertices of distinct short ears are non-adjacent. We will present a strengthening of this in Section III.

The second step is to re-design the short ears so that as many of them as possible are part of a forest (i.e., help connecting vertices that are not internal vertices of short ears). Re-designing a short ear means changing its endpoints by replacing its first and/or last edge by an edge of a trivial ear. This can be reduced to a matroid intersection problem (with a graphic matroid and a partition matroid). For every short ear that is not part of this forest, we can raise the lower bound. We will present a refinement of this step in Section IV.

Finally, two simple algorithms are applied to the resulting ear-decomposition. If at least half of the nontrivial ears are long, simple ear induction (Lemma 1) yields a short tour. Otherwise one can complete the forest of short ears to a spanning tree, so that internal vertices of short ears in the forest (which are pendant) keep their even degree, and then do parity correction like Christofides. The catch is that this parity correction can be done in the ear-decomposition without the short ears in the forest, and hence significantly cheaper if there are many of these.

Here we will combine the two algorithms in the last step to a single new step, which will be described in Section II.

The critical case, when this algorithm has no better approximation ratio than $\frac{3}{2}$, is when (essentially) all ears are even (note that we save $\frac{1}{2}$ more for odd ears in (1) by rounding down the right-hand side), half of the ears are 2-ears, and the 2-ears form a forest.

E. Well-oriented ear-decompositions

Given an ear-decomposition, let $\mathcal{F}$ be a subset of the pendant ears that form a forest. Let $ear(v)$ denote the index of the ear that contains v as an internal vertex. A *rooted orientation* of $\mathcal{F}$ is an orientation of the edges of the ears in $\mathcal{F}$ such that each connected component is an arborescence whose root is a vertex v with $ear(v)$ minimum. Then every ear of $\mathcal{F}$ is a directed path. A *well-oriented ear-decomposition* consists of an ear-decomposition and a rooted orientation of a subset of pendant ears that form a forest. See Figure 1 for an example; the dotted, gray 2-ears are pendant and have a rooted orientation.

We denote by $r(w)$ the root of the connected component of the branching of oriented edges that contains w. We say that an ear Q *enters* another ear P if $Q \in \mathcal{F}$ and there is an oriented edge (v, w) of Q such that $w \in \text{in}(P)$. If any ear enters P we call P *entered*; other nontrivial ears are called *non-entered*. In particular, all oriented ears are pendant and hence non-entered.

F. Summary of new techniques and structure of the paper

Although our proof can be viewed as a refined version of [16], we need many new ideas, some of which may be of independent interest or have further applications.

In Section II, we describe a more sophisticated ear induction, and we assume that we have a well-oriented ear-decomposition in which the oriented ears are precisely the short ears. In particular, we assume the short ears to form a forest. First we will directly use the connectivity service of the short ears, exploiting their orientation and revealing a novel connection to matroid union (Section II-B). This saves in many cases but does not always help. Therefore, we also propose a second new way to benefit from the 2-ears: instead of taking a 2-ear as it is, one can also double one edge and discard the other, changing the parity at the endpoints. Combining those two different possibilities of exploiting the short ears, either for connectivity or for parity, we obtain the main result of Section II. Our ear induction algorithm saves at least $\frac{1}{26}$ for every non-entered ear, compared to $\frac{3}{2}(n-1)$, unless most of the long ears are 4-ears (Theorem 4).

Therefore, in addition to the properties of short ears, we need an ear-decomposition with extra properties of 4-ears. In Section III we show that one can always obtain such an ear-decomposition in polynomial time. In particular there will be only four types of 4-ears: pendant, blocked (with a closed ear attached to it), horizontal, or vertical, and at most one third of the long ears can be blocked 4-ears.

Then, in Section IV we re-design short ears but also vertical 4-ears. Again we can use matroid intersection, one matroid is again graphic, but the other one is now a laminar matroid (instead of a partition matroid as in [16]). We can raise the lower bound not only for short ears that are not part of the forest, but also for horizontal and vertical 4-ears.

By this we remove the assumptions that the short ears form a forest and there are not too many 4-ears. See Section V.

We are done unless there are only few non-entered ears. Then there are few nontrivial ears at all because every entered ear is entered by a non-entered (short) ear. But then it is quite easy to obtain a better approximation ratio than $\frac{3}{2}$ if in addition there is a short s-t-path P in G. To see this, let G' result from G by deleting the trivial ears (note that $|E(G')|$ is $n-1$ plus the number of nontrivial ears), and let the vector $x \in \mathbb{R}^{E(G)}$ be the sum of the incidence vectors of P and G', both multiplied by $\frac{1}{3}$. Then one can easily show that x is in the convex hull of T-joins for $T = \{s\}\triangle\{t\}\triangle\{v : v \text{ has odd degree in } G'\}$, and hence adding a minimum T-join results in an s-t-tour with $\frac{4}{3}|E(G')| + \frac{1}{3}|E(P)|$ edges. One can do a bit better by applying the removable-pairing technique of Mömke and Svensson [13] in a slightly novel way; see Section VI. We prove a variant of their lemma for well-oriented ear-decompositions that works without the 2-vertex-connectivity assumption.

The only remaining case is when the distance from s to t in G is large (close to $\frac{n}{2}$), but then one can apply recursive dynamic programming similar to [2] and [19] (Section VII). This recursive dynamic programming algorithm yields a general statement about approximation algorithms for the s-t-path TSP, not only applying to the graph case: we show that for finding a polynomial-time α-approximation for some constant $\alpha > 1$, it is sufficient to consider the special case where the distance of the vertices s and t is at most $\frac{1}{3}+\varepsilon$ times the cost of an optimum solution for some arbitrary constant $\varepsilon > 0$.

The case in which our ear-decomposition has only very few non-entered ears is the only case in which we do not compare our solution to the optimum LP value but to the optimum s-t-tour. Here, the dynamic programming algorithm allows us to bound the number of edges of our s-t-tour with respect to OPT (rather than the LP value), which we need to obtain an approximation ratio below the integrality ratio of the LP.

Let us review the overall algorithm: first use the reduction to the case where s and t have small distance (Theorem 12). To solve this case, first compute an initial ear-decomposition as in Theorem 6. Based on this, compute an optimized and well-oriented ear-decomposition (Section IV). If there are many non-entered ears, we get a short tour by enhanced ear induction (Theorem 4). If there are few non-entered ears, we obtain a short tour by the removable-pairing technique; see Lemma 10. Our presentation follows a different order because each section is motivated by the previous ones.

For complete proofs see the full paper [20].

II. Enhanced ear induction

A. Outline of our ear induction algorithm

In this section we describe an ear induction algorithm that computes a T-tour, where $T \subseteq V(G)$ is a given even-cardinality set. Our goal is to obtain an upper bound on the number of edges where we gain some constant amount per non-entered ear, compared to $\frac{3}{2}(n-1)$.

For the entire Section II we will assume that we are given a well-oriented ear-decomposition in which all short ears are clean and the oriented ears are precisely the clean ears. In particular, the clean ears are all pendant and form a forest. Later (in Section IV) we consider the general case.

Let $G_\gamma = (V(G), E_\gamma)$ be the spanning subgraph of G that contains only the edges of clean ears. Due to the rooted orientation this is a branching. Every connected component of G_γ will be used either for connectivity or for parity correction. If we use a connected component of G_γ for connectivity, we add all edges of the component to our T-tour. However, we can instead use a component C of G_γ consisting of only 2-ears for parity correction as follows: Let T' be a set of vertices that are contained in the component C, but are not internal

Fig. 2. Using 2-ears for parity correction: The filled squares denote internal vertices of long ears, the circles internal vertices of 2-ears. Edges of long ears and the orientation of short ears are not shown here. The vertex set T' is shown in green. The blue edges (in the left picture) show a T'-join in this component of G_γ. The right picture shows the same component after "flipping" the T'-join. Compared to the left picture, precisely the parity of the degree of the vertices in T' are changed.

vertices of short ears (see Figure 2). If $|T'|$ is even, we can change the parity of exactly the vertices in T' by "flipping" the 2-ears that are part of a T'-join in C, i.e. we take two copies of one edge instead of one copy of each of the edges of the "flipped" 2-ears (see Figure 2). As a consequence, we can choose the parity of all vertices that have an entering clean ear in C and then fix the parity at the root of the component C such that the set T' of vertices where we need to change the parity of the degree has even cardinality. We can also flip 3-ears, but then we need four instead of three edges.

If we could bound the number of edges that we need during ear induction (as in Section I-C) for every ear P by $\frac{3}{2}|\text{in}(P)|$, we would obtain a T-tour with at most $\frac{3}{2}(n-1)$ edges. Lemma 1 yields an even better bound for long ears (of length at least four); so we can gain some constant amount per long (non-entered) ear. However, for (clean) 2-ears we need two edges, which is $\frac{1}{2}$ more than $\frac{3}{2}|\text{in}(P)|$, and also for (clean) 3-ears we can not improve over $\frac{3}{2}|\text{in}(P)| = 3$. To make up for this, we would like to improve over $\frac{3}{2}|\text{in}(P)|$ for long ears by some constant amount for each short (oriented) ear entering the ear P. In order to gain from a short ear entering P at some vertex w, we then either exploit that the clean ears connect w to the root $r(w)$ (if the component of G_γ containing w is used for connectivity) or make use of the fact that we can choose the parity of the vertex w (by possibly changing the parity at $r(w)$).

In Section II-B we prove our main lemma for enhanced ear induction, which allows us to benefit from the connectivity service of the clean ears in many cases. In Section II-C we describe our ear induction algorithm that makes use of the short ears for connectivity. However, some connected components of G_γ will not be used for connectivity and we instead use them for parity correction in a post-processing step improving the T-tour found by ear induction. We then compute a second T-tour, again by ear-induction, now using clean ears only for parity correction. Taking the better of the two constructed T-tours, we can improve upon $\frac{3}{2}(n-1)$ by $\frac{1}{26}$ per non-entered ear, unless a large fraction of the long ears are 4-ears. This is our main result of Section II, summarized in Theorem 4.

B. Using clean ears for connectivity via matroid union

In this section we prove our main lemma for enhanced ear induction. It yields a better bound than Lemma 1 in many cases by making use of the contribution of the clean ears to connectivity. In the proof we will need a lemma that can be proved using the matroid union theorem (cf. Figure 3 (a),(b)).

Lemma 2: Let (V, E) be a graph (possibly with parallel edges) and a partition of E into nonempty sets R, B, U such that (V, U) is a forest and $(V, R \cup B)$ is a circuit. Then there is a partition of U into sets U_R, U_B, and Z such that $(V, R \cup U_R)$ is a forest, $(V, B \cup U_B)$ is a forest, and Z contains at most one element.

An *antipodal pair* in a circuit P is a set of two vertices of P that have distance $\frac{|E(P)|}{2}$ in P. Obviously, only even circuits have antipodal pairs. We use the following lemma to exploit the connectivity service of clean ears during ear induction. The clean ears entering P are represented by the edge set U in this lemma. A subset C of U will be used for connectivity.

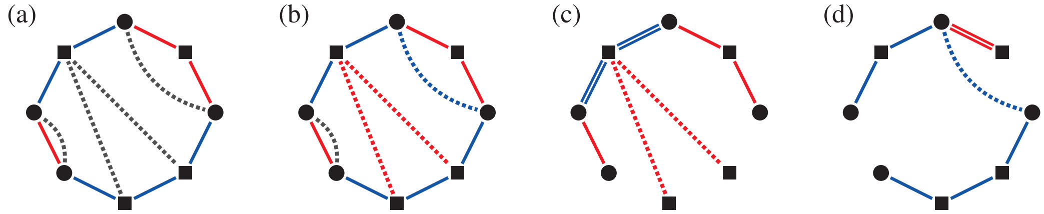

Fig. 3. Enhanced ear induction in Lemma 3: (a) ear P is drawn with solid lines, dotted lines indicate the edges in U, black squares are elements of $V(P)\setminus T_P$; black circles are elements of $V(P)\cap T_P$; (b) coloring the edges of U red and blue; (c) the red solution F_R; (d) the blue solution F_B.

Lemma 3: Let P be a circuit with at least four edges and $T_P \subseteq V(P)$ with $|T_P|$ even. Let $(V(P), U)$ be a forest. Then one of the following holds:

(i) $|E(P)| = 4$ and $|U| = 1$ and $T_P = \emptyset$;

(ii) T_P is an antipodal pair, and U consists of a single edge with endpoints T_P;

(iii) There exists a T_P-join $F \subseteq 2E(P)$ and a subset $C \subseteq U$ such that the graph $(V(P), F \cup C)$ is connected,

$$|F| \le \tfrac{3}{2}(|E(P)|-1) - \tfrac{1}{2}|U| - \tfrac{1}{2}\max\{1, |U|-1\},$$

and $|C| \le 2\left(\frac{3}{2}(|E(P)|-1) - \frac{1}{2}|U| - |F|\right)$.

Proof: Similarly to the proof of Lemma 1, we distinguish two cases.

Case 1: $T_P \neq \emptyset$.

The vertices of T_P subdivide P into subpaths, alternatingly colored red and blue. Let E_R and E_B denote the set of edges of red and blue subpaths, respectively. Let T_R and T_B be the set of vertices having odd degree in $(V(P_i), E_R)$ and $(V(P_i), E_B)$, respectively. Note that $\{E_R, E_B\}$ is a partition of $E(P)$, both sets are nonempty, and $T_R = T_B = T_P$. Color the edges in U red and blue according to Lemma 2; one edge may remain uncolored. Let C be the set of colored edges in U.

We consider two solutions (see Figure 3): To construct F_R, we take E_R plus two copies of some edges of E_B. Since E_R plus the red elements of U form a forest, the number of blue edges needed for connectivity (with two copies each) is at most $|E_B| - 1$ minus the number of red elements of U. To construct F_B, we exchange the roles of red and blue.

The smaller of the two has at most $\frac{1}{2}(3|E(P)| - 4 - 2|C|) \le \frac{1}{2}(3|E(P)| - 4 - (|U|-1) - |C|) = \frac{3}{2}(|E(P)|-1) - \frac{1}{2}|U| - \frac{1}{2}|C|$ edges. Since $|C| \ge |U| - 1$, we are done if $|U| > 1$ or $|U| = |C| = 1$.

Now let $|U| \le 1$. If $U = \emptyset$, then the smaller of the sets F_B and F_R has size at most $\frac{1}{2}(3|E(P)| - 4) = \frac{3}{2}(|E(P)|-1) - \frac{1}{2}$.

Now consider the remaining case that $|U| = 1$ and $C = \emptyset$, i.e., the only edge in U cannot be colored. Then the endpoints of this edge are connected by a path in E_R and a path in E_B, and hence T_P consists of exactly these two elements. If (ii) does not hold, T_P is not an antipodal pair, and hence $|E_R| \neq |E_B|$. Then the smaller of the sets F_B and F_R has size at most $|E(P)| + \min\{|E_R|, |E_B|\} - 2 \le \frac{3}{2}|E(P)| - \frac{1}{2} - 2 = \frac{3}{2}(|E(P)|-1) - \frac{1}{2}|U| - \frac{1}{2}$.

Case 2: $T_P = \emptyset$.

We can set $F = E(P)$ and $C = \emptyset$, but instead we can also set $C = U$ and take all but $|U|+1$ edges, each with two copies, making $2(|E(P)| - |U| - 1)$ edges. The smaller of the two choices for F has at most $\frac{1}{2}(3|E(P)| - 2|U| - 2) = \frac{3}{2}(|E(P)| - 1) - \frac{1}{2}|U| - \frac{1}{2}(|U|-1)$ edges. If $|U| > 1$, this implies the claimed upper bound on $|F|$. If $|U| > 1$ and both solutions have the same number of edges, $F = E(P)$ and $C = \emptyset$ fulfills (iii). If $|U| > 1$ and one of the two solutions for F has fewer edges, the smaller of the two choices for F has at most $\frac{1}{2}(3|E(P)| - 2|U| - 3) = \frac{3}{2}(|E(P)|-1) - \frac{1}{2}|U| - \frac{1}{2}|U|$ edges. Then we also have $|C| \le |U| = 2\left(\frac{3}{2}(|E(P)|-1) - \frac{1}{2}|U| - |F|\right)$.

If $|U| \le 1$, we have (i) or $|E(P)| - 1 \ge 3 + |U|$. In the latter case, $|E(P)| \le \frac{3}{2}(|E(P)|-1) - \frac{1}{2}|U| - \frac{1}{2}$. Hence, we can set $F = E(P)$ and $C = \emptyset$. □

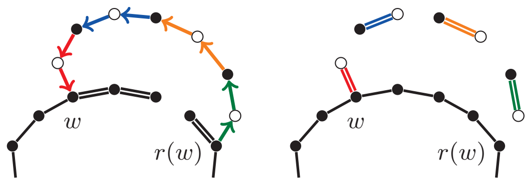

Fig. 4. Post-processing a T-tour, “flipping” clean ears that have not been used for connectivity. The edges shown in black are edges in $E(P_i)$, the colored edges are edges of (pendant) 2-ears that are part of the $r(w)$-w-path. The edges of different 2-ears are shown in different colors. The filled vertices are internal vertices of long ears, i.e. these vertices are contained in V_l. The non-filled vertices are internal vertices of (pendant) 2-ears. The left picture shows the edges of $E(P)$ and the $r(w)$-w-path used in the T-tour before modifying it, and the right picture shows the used edges after the modification.

C. Ear induction algorithm

We now explain how to construct a T-tour by enhanced ear induction. We first take the set F of all edges of short ears. Then we proceed with the long ears $P_1, \ldots, P_l$ in reverse order. We set $T_l := T \triangle \{v : |\delta_F(v)| \text{ odd}\}$. Now we use ear-induction to construct a T_l-join in the graph G_l. In contrast to the simple ear-induction in Section I-C, this T_l-join will not necessarily be connected, since we will exploit the connectivity service of the short ears. Let again (P, T_P) be $(G_i, T_i)/V_{i-1}$. We now define the set U that models the connectivity provided by the clean ears. For each ear entering P_i at w, the set U contains the edge that results from $\{r(w), w\}$ by contracting V_{i-1}. Whenever possible, we apply Lemma 3 (iii). Otherwise, we apply Lemma 1. In any case, we obtain a set $F_i \subseteq 2E(P_i)$. We set $F := F \cup F_i$ and $T_{i-1} := T_i \triangle \{v \in V(P_i) : |\delta_{F_i}(v)| \text{ odd}\}$ and continue with the next long ear (in reverse order). This algorithm computes a T-tour F in G.

We now apply a post-processing step to improve our T-tour. Recall that in Lemma 3 (iii) we do not use all edges of U for connectivity, but only those in C. If there is a connected component of the branching of short ears that is not used for connectivity at all, we may use it for parity changes. This helps in case (ii) of Lemma 3 if the entering ear enters at w and $r(w) \in \text{in}(P_i)$; see Figure 4.

Then we compute a second T-tour, again by ear induction. However, now we will use all connected components of clean ears for parity correction. If a clean ear Q enters a long ear P_i at w and $r(w) \notin \text{in}(P_i)$, the root $r(w)$ must be contained in V_{i-1} by our choice of the orientation of the clean ears. Thus, by “flipping” clean ears as in Figure 2, we can choose the parity of the vertex w (which implies a change of the parity at $r(w)$). For details see the full paper.

Finally, we take the better of the two computed T-tours. The result of this ear induction algorithm is summarized in the following theorem.

Theorem 4: Let G be a graph and $T \subseteq V(G)$ with $|T|$ even. Given a well-oriented ear-decomposition of G where all short ears are clean and the oriented ears are precisely the clean ears, we can compute a T-tour in G with at most

$$\tfrac{3}{2}(n-1) - \tfrac{1}{26}\pi + \tfrac{1}{26}(k_4 - 2k_{\geq 5})$$

edges, where π is the number of non-entered ears.

III. Computing the initial ear-decomposition

Previous papers that used ear-decompositions for approximation algorithms include [3], [16], and [10]. They all exploit a theorem of Frank [6]: one can compute an ear-decomposition with minimum number of even ears in polynomial time. Our ear-decompositions will also have this property, although we exploit it only during the construction of the ear-decomposition. The ear-decompositions in the above papers also have certain properties of 2-ears and 3-ears; we will additionally deal with 4-ears. As in [16] we will compute a nice ear-decomposition. In particular, we make all short ears pendant.

We have seen in Theorem 4 that non-entered ears are cheap but 4-ears are expensive. For pendant 4-ears we can apply Lemma 1 beforehand and apply Theorem 4 to the rest. Ideally, we would like to make all 4-ears pendant, but this is not always possible. We distinguish four kinds of 4-ears: pendant, blocked, vertical, and horizontal (we will compute an ear decomposition in which every ear is of exactly one of these kinds); see Figure 5.

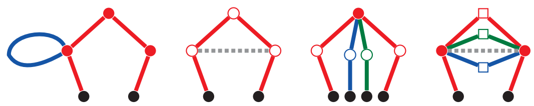

Fig. 5. (a) blocked 4-ear, (b) pendant 4-ear (c) vertical 4-ear, (d) horizontal 4-ear. Filled circles denote arbitrary vertices, unfilled circles denote pendant vertices, unfilled squares denote degree-2 vertices. The endpoints of the 4-ears and endpoints of 2-ears that are not internal vertices of these 4-ears are shown at the bottom (black filled circles); some of these can be identical. Curves denote closed ears. Dotted edges are possible trivial ears connecting colored vertices.

Definition 5: A 4-ear is called

- *blocked* if a closed ear is attached to it.
- *vertical* if it is nonpendant, its internal vertices are v_1, v_2, v_3 in this order, v_1 and v_3 are pendant and not adjacent, and the only nontrivial ears attached to v_2 are 2-ears whose middle vertex is not adjacent to v_1 or v_3.
- *horizontal* if it is nonpendant, its internal vertices are v_1, v_2, v_3 in this order, v_2 has degree 2, and all nontrivial ears attached to it are 2-ears with vertices v_1, x, v_3, where x is a degree-2 vertex.

Call an ear *outer* if it is a 2-ear, 3-ear, pendant 4-ear, vertical 4-ear, or horizontal 4-ear. Call an ear *inner* if it is a blocked 4-ear or has length at least 5.

Later we will show that there are only few blocked 4-ears (Lemma 8). Moreover, in Section IV we make as many vertical 4-ears pendant as possible and raise the lower bound for every 2-ear that is still attached to a vertical or horizontal 4-ear.

The main result of this section is the following.

Theorem 6: For any 2-vertex-connected graph G we can construct an ear-decomposition in polynomial time that satisfies the following conditions:

(a) All short ears are open and pendant.
(b) Each 4-ear is blocked or pendant or vertical or horizontal; no closed 4-ear is attached to any closed 4-ear.
(c) If there are internal vertices v of an outer ear P and w of another outer ear Q that are adjacent, then P is attached to Q at w, Q is attached to P at v, or both v and w are middle vertices of 4-ears. No 2-ear is attached to two outer 4-ears.

To prove this theorem we start from an open ear-decomposition with minimum number of even ears. Then we apply a set of operations that change the ear decomposition locally as long as possible. If none of the operations can be applied anymore, the ear-decomposition fullfills the properties as stated in Theorem 6. In order to bound the number of operations that we need to apply, we consider a potential function that increases with every operation we apply, and that can attain only polynomially many values. For the details of the proof, we refer to the full version of this paper.

IV. Optimizing outer ears and improving the lower bound

The ear-decomposition from Theorem 6 is the starting point for optimizing the outer ears. While we do not touch pendant or horizontal 4-ears, we will change short ears and vertical 4-ears. Like in [16], our goal is that as many short ears as possible form a forest. We will in addition try to make the vertical 4-ears pendant, by re-designing the 2-ears attached to them. The two subpaths of a 4-ear from the middle vertex to an endpoint will be part of this optimization, and might be replaced by attached 2-ears.

For every 2-ear that will not be part of the forest or remains attached to an outer 4-ear, we will raise the lower bound. This includes in particular 2-ears attached to horizontal ears, which cannot be optimized.

A. Matroid intersection

Given an ear-decomposition as in Theorem 6, let M be the union of

$$\big\{\text{in}(P) : P \text{ short ear not attached to a horizontal 4-ear}\big\}$$

and

$$\big\{\{v\} : v \text{ non-middle internal vertex of a vertical 4-ear}\big\}.$$

See Figure 6. Moreover, let A denote the set of middle vertices of vertical 4-ears, and for $a \in A$ let

$$M(a) \ := \ \big\{\{v\} \in M : \{v, a\} \in E(G)\big\}.$$

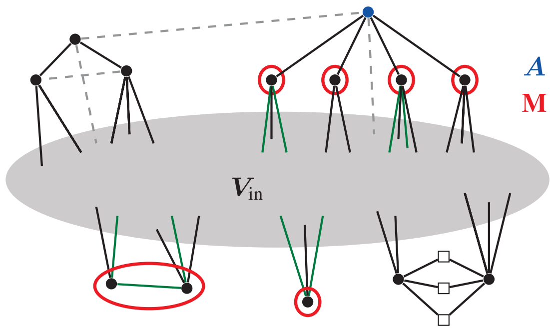

Fig. 6. The outer ears in an ear-decomposition as in Theorem 6, and the sets M and A. Green edges constitute a possible set of paths that is independent in both matroids. Dashed edges show possible trivial ears.

Note that $|M(a)| - 2$ is the number of 2-ears attached to a. Let

$$V_{\text{in}} := \{v \in V(P) : P \text{ an inner ear}\}.$$

The sets A, V_{in} and the elements of M are pairwise disjoint. Moreover, the vertices of G not contained in any of these sets are precisely the internal vertices of pendant or horizontal 4-ears and the internal vertices of 2-ears attached to horizontal 4-ears.

Now let $\mathcal{P}_f$ for $f \in M$ denote the set of paths in G with the following two properties:

- The set of internal vertices of the path is f.
- The endpoints of the path are both contained in V_{in}.

We now define two matroids on the ground set $\bigcup_{f \in M} \mathcal{P}_f$.

The independent sets of the first matroid $\mathcal{M}_1$ are given by all sets $\mathcal{I} \subseteq \bigcup_{f \in M} \mathcal{P}_f$ such that $\big(V(G), \bigcup_{I \in \mathcal{I}} E(I)\big)$ is a forest. It is clear that $\mathcal{M}_1$ is a graphic matroid (every path can be represented by an edge connecting its endpoints).

In the second matroid $\mathcal{M}_2$ a set $\mathcal{I} \subseteq \bigcup_{f \in M} \mathcal{P}_f$ is independent if and only if it fulfills the following two conditions:

(i) $|\mathcal{I} \cap \mathcal{P}_f| \leq 1$ for all $f \in M$

(ii) For every $a \in A$ we have

$$\left| \bigcup_{f \in M(a)} \mathcal{P}_f \cap \mathcal{I} \right| \leq |M(a)| - 2.$$

Note that (i) and (ii) characterize the independent sets of a laminar matroid (see e.g. [7]).

We compute a maximum cardinality set $\mathcal{I} \subseteq \bigcup_{f \in M} \mathcal{P}_f$ such that $\mathcal{I}$ is an independent set in both matroids defined above.

The idea is that we want to choose as many short ears as possible to form a forest but leave two elements of each $M(a)$ unused; they will form the 4-ear with middle vertex a. Thus the vertical 4-ears can also change.

Using the matroid intersection theorem we can show:

Theorem 7: If $\mathcal{I}$ is a maximum cardinality set that is independent in both matroids, then

$$\text{LP} \geq n - 3 + \tfrac{1}{2}(k_2 + k_3 - |\mathcal{I}|).$$

V. Ear-decompositions with many non-entered ears

After optimizing the outer ears via the matroid intersection approach described above, we will distinguish between primary ears (those that were inner ears, plus clean ears that form a forest) and secondary ears. For secondary ears we will apply Lemma 1, and for secondary short ears we raise the lower bound using Theorem 7. For primary ears we will apply Theorem 4. Hence we need to bound the number of primary 4-ears, which correspond exactly to the blocked 4-ears before optimization. Their number can be bounded easily as follows.

Lemma 8: Given an ear-decomposition as in Theorem 6, denote by $k_{4,\text{ blocked}}$ and $k_{4,\text{ non-blocked}}$ the number of blocked and non-blocked 4-ears, respectively. Then

$$k_{4,\text{ blocked}} \leq 2k_{\geq 5} + k_{4,\text{ non-blocked}}.$$

Proof: Recall that a 4-ear is blocked if a closed ear is attached to it. Since all short ears are open and no closed 4-ear is attached to any closed 4-ear, the only closed ears attached to a 4-ear can be ears of length at least five, non-blocked 4-ears, and 4-ears to which a closed ear of length at least five is attached. Using that every closed ear is attached to exactly one ear, we get the result. □

With this bound we now easily obtain the following theorem.

Theorem 9: Given a graph G and $s, t \in V(G)$, we can compute a well-oriented ear-decomposition of G with π non-entered ears, and an s-t-tour with at most

$$\left(\frac{3}{2} - \frac{1}{26} \cdot \frac{\pi}{n-1}\right) \text{LP} + 3$$

edges in polynomial time.

VI. Ear-decompositions with few non-entered ears

In this section we show how to compute a cheap s-t-tour if our well-oriented ear-decomposition has only few non-entered ears and s and t have small distance. The following lemma shows a new way to apply the removable-pairing technique of Mömke and Svensson [13] to ear-decompositions. In contrast to [16], we do not require the graph after deleting the trivial ears to be 2-vertex-connected. This requires a slight modification of their proof.

Lemma 10: Let G be a graph with a well-oriented ear-decomposition, and $s, t \in V(G)$. Let π be the number of non-entered ears. Then we can compute an s-t-tour with at most

$$\frac{4}{3}(n-1) + \frac{2}{3}\pi + \frac{1}{3}\text{dist}(s,t)$$

edges in polynomial time, where $\text{dist}(s,t)$ denotes the distance of s and t in G.

Theorem 11: Given an instance of the s-t-path graph TSP with a 2-vertex-connected graph in which s and t have distance at most $0.3334\,\text{OPT}$, where OPT denotes the number of edges in the optimum solution, we can compute an s-t-tour with at most $1.497\,\text{OPT}$ edges in polynomial time.

Proof: We apply Theorem 9 and obtain a well-oriented ear-decomposition and an s-t-tour. If the number of non-entered ears is at least $\frac{13}{165}(n-1)$, then this s-t-tour has at most $(\frac{3}{2} - \frac{1}{330})\text{LP} + 3$ edges. If $n > 99000$, this is at most $1.497\,\text{LP} \leq 1.497\,\text{OPT}$ since $\text{LP} \geq n-1$. For $n \leq 99000$ we can solve the instance by complete enumeration.

If the number of non-entered ears is at most $\frac{13}{165}(n-1)$, then Lemma 10 yields an s-t-tour with at most $(\frac{4}{3} + \frac{26}{495})(n-1) + \frac{0.3334}{3}\,\text{OPT} < 1.497\,\text{OPT}$ edges. □

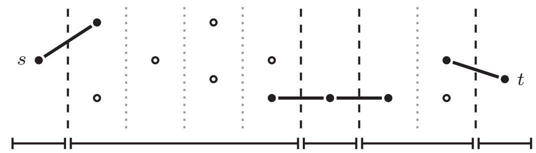

Fig. 7. Partitioning the instance by level cuts. The vertical lines show the level cuts; the dashed vertical lines are those that contain only one edge of the optimum s-t-tour F^*. (This edge is shown.) Here, we compose a tour of these four edges and tours for the five subinstances indicated at the bottom, some of which are trivial.

VII. Solving instances with large distance of s and t

We show that for finding a polynomial time α-approximation for the s-t-path graph TSP for some constant $\alpha > 1$, it is sufficient to consider the special case where the distance of the vertices s and t is at most $\frac{1}{3} + \varepsilon$ times the length of an optimum solution for some arbitrary constant $\varepsilon > 0$.

Theorem 12: Let $\varepsilon > 0$ and $\alpha > 1$ be constants. Assume there exists a polynomial time algorithm that computes a solution with at most $\alpha \cdot \text{OPT}$ edges for instances of the s-t-path graph TSP with a 2-vertex-connected graph G and $\text{dist}(s,t) \leq (\frac{1}{3} + \varepsilon) \cdot \text{OPT}$. Then there exists a polynomial time α-approximation algorithm for the s-t-path graph TSP.

Our proof of Theorem 12 also applies to weighted graphs and hence yields an analagous statement for the general s-t-path TSP (see the full version of this paper).

Assume we have a β-approximation algorithm $\mathcal{A}$ for the s-t-path graph TSP which achieves an approximation ratio of α in the special case where $\text{dist}(s,t) \leq (\frac{1}{3} + \varepsilon) \cdot \text{OPT}$, where $1 < \alpha < \beta$. Then we show how to obtain such an algorithm for α and β', where $\beta' = \max\left\{\alpha,\ \beta - \frac{3}{2}(\beta - 1) \cdot \varepsilon\right\}$. After a constant number of iterations, this yields an α-approximation algorithm for general instances.

Let F^* be an optimum s-t-tour. If $\text{dist}(s,t) > (\frac{1}{3} + \varepsilon) \cdot \text{OPT}$, then a constant fraction of the level cuts $C_i := \delta(\{v : \text{dist}(s,v) < i\})$ $(i = 1, \ldots, \text{dist}(s,t))$ contains only one edge of F^*.

For each pair (C_i, C_j) of level cuts $(i < j)$ and vertices s', t' with $\text{dist}(s,s') = i$ and $\text{dist}(s,t') = j - 1$, we apply the algorithm $\mathcal{A}$ to find an s'-t'-

tour visiting the vertices between the level cuts, i.e. $\{v : i \leq \text{dist}(s, v) < j\}$. We then use dynamic programming to construct an s-t-tour by combining a selection of these tours; see Figure 7. A similar dynamic program was used in [2] to obtain the first constant-factor approximation algorithm for the orienteering problem, and also in [19] and [21] for the s-t-path TSP.

VIII. Conclusion

Theorem 12 and Theorem 11 directly imply or main theorem:

Theorem 13: There is a polynomial-time 1.497-approximation algorithm for the s-t-path graph TSP.

Our main goal was to show that one can go below the integrality ratio lower bound, even while using the LP in the analysis. We did not attempt to optimize the running time or the approximation ratio of our algorithm, and improvements are certainly possible (but would probably make the proof more complicated). A more interesting question is whether our new techniques have further applications.

Given that one can achieve a better approximation ratio for the s-t-path TSP than $\frac{3}{2}$, the integrality ratio, it is natural to ask what is the best approximation ratio that can be obtained. The only lower bound known today (unless P=NP) is $\frac{685}{684}$ [12].

Another interesting question is how the integrality ratio depends on the distance of s and t. Let $\rho(d)$ denote the integrality ratio for 2-vertex-connected instances with $\text{dist}(s,t) = d \cdot \text{LP}$. The proof of Theorem 11 shows (for n large enough) that $\rho(d) < \frac{3}{2}$ for all $d < \frac{1}{2}$ and the bound improves as d decreases.

References

[1] An, H.-C., Kleinberg, R., and Shmoys, D.B. [2015]: Improving Christofides' algorithm for the s-t path TSP. Journal of the ACM 62 (2015), Article 34

[2] Blum, A., Chawla, S., Karger, D., Lane, T., Meyerson, A., and Minkoff, M. [2007]: Approximation algorithms for orienteering and discounted-reward TSP. SIAM Journal on Computing 37 (2007), 653–670

[3] Cheriyan, J., Sebő, A., and Szigeti, Z. [2001]: Improving on the 1.5-approximation of a smallest 2-edge connected spanning subgraph. SIAM Journal on Discrete Mathematics 14 (2001), 170–180

[4] Edmonds, J. [1968]: Matroid partition. In: Mathematics of the Decision Sciences, Part 1 (G.B. Dantzig, A.F. Veinott, eds.). Americal Mathematical Society, Providence 1968, pp. 335–345

[5] Edmonds, J. [1970]: Submodular functions, matroids and certain polyhedra. In: Combinatorial Structures and Their Applications; Proceedings of the Calgary International Conference on Combinatorial Structures and Their Applications 1969 (R. Guy, H. Hanani, N. Sauer, J. Schönheim, eds.), Gordon and Breach, New York 1970, pp. 69–87

[6] Frank, A. [1993]: Conservative weightings and ear-decompositions of graphs. Combinatorica 13 (1993), 65–81

[7] Frank, A. [2011]: Connections in Combinatorial Optimization. Oxford University Press 2011

[8] Gao, Z. [2013]: An LP-based $\frac{3}{2}$-approximation algorithm for the s-t path graph traveling salesman problem. Operations Research Letters 41 (2013), 615–617

[9] Gottschalk, C., and Vygen, J. [2016]: Better s-t-tours by Gao trees. In: Integer Programming and Combinatorial Optimization; Proceedings of the 18th IPCO Conference; LNCS 9682 (Q. Louveaux, M. Skutella, eds.), Springer, Cham 2016, pp. 126–137. Mathematical Programming B, to appear

[10] Heeger, K., and Vygen, J. [2017]: Two-connected spanning subgraphs with at most $\frac{10}{7}$ OPT edges. SIAM Journal on Discrete Mathematics 31 (2017), 1820–1835

[11] Hoogeveen, J.A. [1991]: Analysis of Christofides' heuristic: some paths are more difficult than cycles. Operations Research Letters 10 (1991), 291–295

[12] Karpinski, M., and Schmied, R. [2015]: Approximation hardness of graphic TSP on cubic graphs. RAIRO Operations Research 49 (2015) 651–668

[13] Mömke, T., and Svensson, O. [2016]: Removing and adding edges for the traveling salesman problem. Journal of the ACM 63 (2016), 2:1-2:28

[14] Mucha, M. [2014]: $\frac{13}{9}$-approximation for graphic TSP. Theory of Computing Systems 55 (2014), 640–657

[15] Sebő, A. [2013]: Eight fifth approximation for TSP paths. In: Integer Programming and Combinatorial Optimization; Proceedings of the 16th IPCO Conference; LNCS 7801 (J. Correa, M.X. Goemans, eds.), Springer 2013, pp. 362–374

[16] Sebő, A., and Vygen, J. [2014]: Shorter tours by nicer ears: 7/5-approximation for graph-TSP, 3/2 for the path version, and 4/3 for two-edge-connected subgraphs. Combinatorica 34 (2014), 597–629

[17] Sebő, A., and van Zuylen, A. [2016]: The salesman's improved paths: 3/2+1/34 integrality gap and approximation ratio. Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2016), 118–127

[18] Svensson, O., Tarnawski, J., and Végh, L. [2018]: A constant-factor approximation algorithm for the asymmetric traveling salesman problem. arXiv:1708.04215; Proceedings of the 50th Annual ACM Symposium on the Theory of Computing (2018), 204–213

[19] Traub, V., and Vygen, J. [2018]: Approaching $\frac{3}{2}$ for the s-t-path TSP. Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (2018), 1854–1864

[20] Traub, V., and Vygen, J. [2018]: Beating the integrality ratio for s-t-tours in graphs. arXiv:1804.03112

[21] Zenklusen, R. [2018]: A 1.5-approximation for path TSP arXiv:1805.04131